

PSYCHOPHYSICS

A PRACTICAL INTRODUCTION

SECOND EDITION

FREDERICK A.A. KINGDOM
McGill University, Montreal, Quebec, Canada

NICOLAAS PRINS
University of Mississippi, Oxford, MS, USA



AMSTERDAM • BOSTON • HEIDELBERG • LONDON
NEW YORK • OXFORD • PARIS • SAN DIEGO
SAN FRANCISCO • SINGAPORE • SYDNEY • TOKYO
Academic Press is an imprint of Elsevier



Academic Press is an imprint of Elsevier
125 London Wall, London EC2Y 5AS, UK
525 B Street, Suite 1800, San Diego, CA 92101-4495, USA
225 Wyman Street, Waltham, MA 02451, USA
The Boulevard, Langford Lane, Kidlington, Oxford OX5 1GB, UK

Copyright © 2016, 2010 Elsevier Ltd. All rights reserved.

Cover image: This item is reproduced by permission of The Huntington Library, San Marino, California.

No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or any information storage and retrieval system, without permission in writing from the publisher. Details on how to seek permission, further information about the Publisher's permissions policies and our arrangements with organizations such as the Copyright Clearance Center and the Copyright Licensing Agency, can be found at our website: www.elsevier.com/permissions.

This book and the individual contributions contained in it are protected under copyright by the Publisher (other than as may be noted herein).

Notices

Knowledge and best practice in this field are constantly changing. As new research and experience broaden our understanding, changes in research methods, professional practices, or medical treatment may become necessary.

Practitioners and researchers must always rely on their own experience and knowledge in evaluating and using any information, methods, compounds, or experiments described herein. In using such information or methods they should be mindful of their own safety and the safety of others, including parties for whom they have a professional responsibility.

To the fullest extent of the law, neither the Publisher nor the authors, contributors, or editors, assume any liability for any injury and/or damage to persons or property as a matter of products liability, negligence or otherwise, or from any use or operation of any methods, products, instructions, or ideas contained in the material herein.

ISBN: 978-0-12-407156-8

British Library Cataloguing-in-Publication Data

A catalogue record for this book is available from the British Library

Library of Congress Cataloging-in-Publication Data

A catalog record for this book is available from the Library of Congress

For information on all Academic Press publications
visit our website at <http://store.elsevier.com/>



Working together
to grow libraries in
developing countries

www.elsevier.com • www.bookaid.org

Publisher: Mica Haley

Acquisition Editor: Melanie Tucker

Editorial Project Manager: Kristi Anderson

Production Project Manager: Caroline Johnson

Designer: Matt Limbert

Typeset by TNQ Books and Journals

www.tnq.co.in

Printed and bound in the United States of America

Dedication

FK would like to dedicate this book to his late parents Tony and Joan, and present family Beverley and Leina. NP would like to dedicate this book to his mother Nel and late father Arie.

About the Authors

Frederick A.A. Kingdom is a Professor at McGill University conducting research into various aspects of visual perception, including color vision, brightness perception, stereopsis, texture perception, contour-shape coding, the perception of transparency, and visual illusions. He also has an interest in models of summation for the detection of multiple stimuli.

Nicolaas Prins is an Associate Professor at the University of Mississippi specializing in visual texture perception, motion perception, contour-shape coding, and the use of statistical methods in the collection and analysis of psychophysical data.

Preface to the Second Edition

The impetus for this book was a recurring question: “Is there a book that explains how to do psychophysics?” Evidently, a book was needed that not only explained the theory behind psychophysical procedures but also provided the practical tools necessary for their implementation. What seemed to be missing was a detailed and accessible exposition of how raw psychophysical responses are turned into meaningful measurements of sensory function; in other words, a book that dealt with the nuts and bolts of psychophysics data analysis.

The need for a practical book on psychophysics inevitably led to a second need: a comprehensive package of software for analyzing psychophysical data. The result was Palamedes. Initially developed in conjunction with the first edition of the book, Palamedes has since taken on a life of its own, and one purpose of the second edition is to catch up with its latest developments! Palamedes will of course continue to be developed so readers are encouraged to keep an eye on the regular updates.

The first few chapters of the book are intended to introduce the basic concepts and terminology of psychophysics as well as familiarize readers with a range of psychophysical procedures. The remaining chapters focus on specialist topics: psychometric functions, adaptive procedures, signal detection theory, summation measures, scaling methods, and statistical model

comparisons. We have also provided an updated quick reference guide to the terms, concepts, and many of the equations described in the book.

In writing the second edition we have endeavored to improve each chapter and have extended all the technical chapters to include new procedures and analyses. Chapter 7 is the book’s one new chapter. It deals with an old but vexing question of how multiple stimuli combine to reach threshold. The chapter attempts to derive from first principles and make accessible to the reader the mathematical basis of the myriads of summation models, scenarios, and metrics that are scattered throughout the literature.

Writing both editions of this book has been a considerable challenge for its authors. Much effort has been expended in trying to make accessible the theory behind different types of psychophysical data analysis. For those psychophysical terms that to us did not appear to have a clear definition we have improvised our own (e.g., the definition of “appearance” given in Chapter 2), and for other terms where we felt there was a lack of clarity we have challenged existing convention (e.g., by referring to a class of forced-choice tasks as 1AFC). Where we have challenged convention we have explained our reasoning and hope that even if readers do not agree with us, they will still find our ideas on the matter thought-provoking.

Acknowledgments

We are indebted to the following persons for kindly reviewing and providing insightful comments on individual chapters: Neil Macmillan and Douglas Creelman for helping one of the authors (FK) get to grips with the calculation of d' for same-different tasks (Chapter 6); Mark Georgeson for providing the derivation of the equation for the criterion measure $\ln\beta$ for a 2AFC task (Chapter 6); Alex Baldwin for the idea of incorporating a stimulus scaling factor g for converting stimulus intensity to d' when modeling psychometric functions within a Signal Detection Theory framework (Chapters 6 and 7); Mark McCourt for providing the figures illustrating grating-induction (Chapter 3); Laurence Maloney for permission to develop and describe the routines for Maximum Likelihood Difference Scaling (Chapter 8); Stanley Klein for encouraging us to include a section on the Chi-squared test (Chapter 9); and Ben Jennings for carefully checking the equations in the summation chapter (Chapter 7).

Thanks also to the many persons—too many to mention individually—who have over the years expressed their appreciation for the book as well as the Palamedes toolbox and provided useful suggestions for improvements to both.

Introduction and Aims

Frederick A.A. Kingdom¹, Nicolaas Prins²

¹McGill University, Montreal, Quebec, Canada; ²University of Mississippi, Oxford, MS, USA

OUTLINE

1.1 What is Psychophysics?	1	1.4 What's New in the Second Edition?	5
1.2 Aims of the Book	1	References	9
1.3 Organization of the Book	2		

1.1 WHAT IS PSYCHOPHYSICS?

According to the online encyclopedia *Wikipedia*, psychophysics "... quantitatively investigates the relationship between physical stimuli and the sensations and perceptions they affect." The term was first coined by Gustav Theodor Fechner, who in his *Elements of Psychophysics* (1860/1966) set out the principles of psychophysical measurement, describing the various procedures used by experimentalists to map out the relationship between matter and mind. Although psychophysics refers to a methodology, it is also a research area in its own right, and much effort continues to be devoted to developing new psychophysical techniques and new methods for analyzing psychophysical data.

Psychophysics can be applied to any sensory system, whether vision, hearing, touch, taste, or smell. This book primarily draws on the visual system to illustrate the principles of psychophysics, but the principles are applicable to all sensory domains.

1.2 AIMS OF THE BOOK

Broadly speaking, the book has three aims. The first is to provide newcomers to psychophysics with an overview of different psychophysical procedures in order to help them

select the appropriate designs and analyses for their experiments. The second aim is to direct readers to the software tools, in the form of Palamedes, for analyzing psychophysical data. This is intended for both newcomers and experienced researchers alike. The third aim is to explain the theory behind the analyses. Again both newcomers and experienced researchers should benefit from the detailed expositions of the bulk of the underlying theory. To this end we have made every effort to make accessible the theory behind a wide range of psychophysical procedures, analytical principles, and mathematical computations, such as Bayesian curve fitting; the calculation of d-primes (d'); summation theory; maximum likelihood difference scaling; goodness-of-fit measurement; bootstrap analysis; and likelihood-ratio testing, to name but a few. In short, the book is intended to be both practical and pedagogical.

The inclusion of the description of the Palamedes tools, placed in this edition in separate boxes alongside the main text, will hopefully offer the reader something more than is provided by traditional textbooks, such as the excellent *Psychophysics: The Fundamentals* by [Gescheider \(1997\)](#). If there is a downside, however, it is that we do not always delve as deeply into the relationship between psychophysical measurement and sensory function as *The Fundamentals* does, except when necessary to explain a particular psychophysical procedure or set of procedures. In this regard *A Practical Introduction* is not intended as a replacement for other textbooks on psychophysics but as a complement to them, and readers are encouraged to read other relevant texts alongside our own. Two noteworthy recent additions to the literature on psychophysics are [Knoblauch and Maloney's \(2012\) Modeling Psychophysical Data in R](#) and [Lu and Dosher's \(2013\) Visual Psychophysics](#).

Our approach of combining the practical and the pedagogical into a single book may not be to everyone's taste. Doubtless some would prefer to have the description of the software routines put elsewhere. However, we believe that by describing the software alongside the theory, newcomers will be able to get a quick handle on the nuts and bolts of psychophysics methods, the better to then delve into the underlying theory if and when they choose.

1.3 ORGANIZATION OF THE BOOK

The book can be roughly divided into two parts. Chapters 2 and 3 provide an overall framework and detailed breakdown of the variety of psychophysical procedures available to the researcher. Chapters 4–9 are the technical chapters. They describe the theory and implementation for six specialist topics: psychometric functions; adaptive methods; signal detection measures; summation measures; scaling methods; and model comparisons (Box 1.1).

In Chapter 2 we provide an overview of some of the major varieties of psychophysical procedures and offer a framework for classifying psychophysics experiments. The approach taken here is an unusual one. Psychophysical procedures are discussed in the context of a critical examination of the various dichotomies commonly used to differentiate psychophysics experiments: Class A versus Class B; Type 1 versus Type 2; performance versus appearance; forced-choice versus nonforced-choice; criterion-dependent versus criterion-free; objective

BOX 1.1**P A L A M E D E S**

According to Wikipedia, the Greek mythological figure Palamedes (“pal-uh-MEE-deez”) is said to have invented “... counting, currency, weights and measures, jokes, dice and a fore-runner of chess called *pessoi*, as well as military ranks.” The story goes that Palamedes also uncovered a ruse by Odysseus. Odysseus had promised Agamemnon that he would defend the marriage of Helen and Menelaus but pretended to be insane to avoid having to honor his commitment. Unfortunately, Palamedes’s unmasking of Odysseus led to a gruesome end; he was stoned to death for being a traitor after Odysseus forged false evidence against him. Palamedes was chosen as the name for the toolbox because of the legendary figure’s (presumed) contributions to the art of measurement, interest in stochastic processes (he did invent dice!), numerical skills, humor, and wisdom. The Palamedes Swallowtail butterfly (*Papilio palamedes*) on the front cover also provides the toolbox with an attractive icon.

Palamedes is a set of routines and demonstration programs written in MATLAB® for analyzing psychophysical data (Prins and Kingdom, 2009). The routines can be downloaded from www.palamedestoolbox.org. We recommend that you check the website periodically, because new and improved versions of the toolbox will be posted there for download. Chapters 4–9 explain how to use the routines and describe the theory behind them. The descriptions of Palamedes do not assume any knowledge of MATLAB, although a basic knowledge will certainly help. Moreover, Palamedes requires only basic MATLAB; the specialist toolboxes such as the Statistics toolbox are not required. We have also tried to make the routines compatible with earlier versions of MATLAB, where necessary including alternative functions that are called when later versions are undetected. Palamedes is also compatible with the free software package GNU Octave (<http://www.octave.org>).

It is important to bear in mind what Palamedes is not. It is not a package for generating stimuli or for running experiments. In other words it is not a package for dealing with the “front-end” of a psychophysics experiment. The two exceptions to this rule are the Palamedes routines for adaptive methods, which are designed to be incorporated into an actual experimental program, and the routines for generating stimulus lists for use in scaling experiments. But by and large, Palamedes is a different category of toolbox from the stimulus-generating toolboxes such as VideoToolbox (<http://vision.nyu.edu/VideoToolbox/>), PsychToolbox (<http://psychtoolbox.org>), PsychoPy (<http://www.psychopy.org>; see also Peirce, 2007, 2009), and Psykinematix (<http://psykinematix.kybervision.net/>) (for a comprehensive list of such toolboxes see <http://visionscience.com/documents/strasburger/strasburger.html>). Although some of these toolboxes contain routines that perform similar functions to some of the routines in Palamedes, for example fitting psychometric functions (PFs), they are in general complementary to, rather than in competition with, Palamedes.

A few software packages deal primarily with the analysis of psychophysical data. Most of these are aimed at fitting and analyzing psychometric functions. psignifit (<http://psignifit.sourceforge.net/>; see also Fründ et al., 2011) is perhaps the best known of these. Another option is quickpsy, written for R by Daniel Linares and Joan López-Moliner (<http://dlinares.org/quickpsy.html>; see also Linares & López-Moliner, in preparation). Each of the packages

BOX 1.1 (*cont'd*)

will have their own strengths and weaknesses and readers are encouraged to find the software that best fits their needs. A major advantage of Palamedes is that it can fit PFs to multiple conditions simultaneously, while providing the user considerable flexibility in defining a model to fit. Just to give one simple example, one might assume that the lapse rate and slope of the PF are equal between several conditions but that thresholds are not. Palamedes allows one to specify and implement such assumptions and fit the conditions accordingly. Users can also provide their own custom-defined relationships among the parameters from different conditions. For example, users can specify a model in which threshold estimates in different conditions adhere to an exponential decay function (or any other user-specified parametric curve). Palamedes can also determine standard errors for the parameters estimated in such multiple condition fits and perform goodness-of-fit tests for such fits.

The flexibility in model specification provided by Palamedes can also be used to perform statistical model comparisons that target very specific research questions that a researcher might have. Examples are to test whether thresholds differ significantly between two or more conditions, to test whether it is reasonable to assume that slopes are equal between the conditions, to test whether the lapse rate differs significantly from zero (or any other specific value), to test whether the exponential decay function describes the pattern of thresholds well, etc.

Palamedes also does much more than fit PFs; it has routines for calculating signal detection measures and summation measures, implementing adaptive procedures, and analyzing scaling data.

versus subjective; detection versus discrimination; and threshold versus suprathreshold. We consider whether any of these dichotomies could usefully form the basis of a fully-fledged classification scheme for psychophysics experiments and conclude that one, the performance versus appearance distinction, is the best candidate.

Chapter 3 takes as its starting point the classification scheme outlined in Chapter 2 and expands on it by incorporating a further level of categorization based on the number of stimuli presented per trial. The expanded scheme serves as the framework for detailing a much wider range of psychophysical procedures than described in Chapter 2.

Four of the technical chapters, Chapters 4, 6, 8, and 9, are divided into two sections. In these chapters Section A introduces basic concepts and takes the reader through the Palamedes routines that perform the relevant data analyses. Section B provides more detail as well as the theory behind the analyses. The idea behind the Section A versus Section B distinction is that readers can learn about the basic concepts and their implementation without necessarily having to grasp the underlying theory, yet have the theory available to delve into if they want. For example, Section A of Chapter 4 describes how to fit psychometric functions and derive estimates of their critical parameters such as threshold and slope, while Section B describes the theory behind the various fitting procedures. Similarly, Section A

in Chapter 6 outlines why d' measures are useful in psychophysics and how they can be calculated using Palamedes, while Section B describes the theory behind the calculations.

Here and there, we present specific topics in some detail in separate boxes. The idea behind this is that the reader can easily skip these boxes without loss of continuity, while readers specifically interested in the topics discussed will be able to find detailed information there. Just to give one example, Box 4.6 in Chapter 4 explains in much detail the procedure that is used to fit a psychometric function to some data, gives information as to how some fits might fail, and provides tips on how to avoid failed fits.

1.4 WHAT'S NEW IN THE SECOND EDITION?

A major change from the first edition is the addition of the chapter on summation measures (Chapter 7). This chapter provides a detailed exposition of the theory and practice behind experiments that measure detection thresholds for multiple stimuli. Besides the new chapter, all the other chapters have been rewritten to a greater or lesser degree, mainly to include new procedures and additional examples.

Another important change from the first edition is that the description of the Palamedes routines has been put into boxes placed alongside the relevant text. This gives readers greater flexibility in terms of whether, when, and where they choose to learn about Palamedes. The boxes in this chapter ([Box 1 through Box 3](#)) are designed to introduce the reader to Palamedes and its implementation in MATLAB.

BOX 1.2

ORGANIZATION OF PALAMEDES

All the Palamedes routines are prefixed by an identifier `PAL`, to avoid confusion with the routines used by MATLAB. After `PAL`, many routine names contain an acronym for the class of procedure they implement. Box 1.3 lists the acronyms currently in the toolbox, what they stand for, and the book chapter where they are described. In addition to the routines with specialist acronyms, there are a number of general-purpose routines.

Functions

In MATLAB there is a distinction between a function and a script. A function accepts one or more input arguments, performs a set of operations, and returns one or more output arguments. Typically, Palamedes functions are called as follows:

```
>>[x y z] = PAL_FunctionName(a,b,c);
```

where `a`, `b`, and `c` are the input arguments, and `x`, `y`, and `z` the output arguments. In general, the input arguments are “arrays.” Arrays are simply listings of numbers. A scalar is a single number, e.g., 10, 1.5, 1.0e–15. A vector is a one-dimensional array of numbers. A matrix is a two-dimensional array of numbers. It will help you to think of all as being arrays. As a matter of fact, MATLAB represents all as two-dimensional arrays. That is, a scalar is represented as a

Continued

BOX 1.2 (*cont'd*)

1×1 (1 row \times 1 column) array, vectors either as an $m \times 1$ array or a $1 \times n$ array, and a matrix as an $m \times n$ array. Arrays can also have more than two dimensions.

In order to demonstrate the general usage of functions in MATLAB, Palamedes includes a function named `PAL_ExampleFunction`, which takes two arrays of any dimensionality as input arguments and returns the sum, the difference, the product, and the ratio of the numbers in the arrays corresponding to the input arguments. For any function in Palamedes you can get some information as to its usage by typing `help` followed by the name of the function:

```
>>help PAL_ExampleFunction
```

MATLAB returns

```
PAL_ExampleFunction calculates the sum, difference, product, and
ratio of two scalars, vectors or matrices.
```

```
syntax: [sum difference product ratio] = ...
PAL_ExampleFunction(array1, array2)
```

```
This function serves no purpose other than to demonstrate the
general usage of Matlab functions.
```

For example, if we type and execute

```
[sum difference product ratio] = PAL_ExampleFunction(10, 5);
```

MATLAB will assign the arithmetic sum of the input arguments to a variable labeled `sum`, the difference to `difference`, etc. In case the variable `sum` did not previously exist, it will have been created when the function was called. In case it did exist, its previous value will be overwritten (and thus lost). We can inquire about the value of a variable by typing its name, followed by <return>:

```
>>sum
```

MATLAB returns

```
sum = 15
```

We can use any name for the returned arguments. For example, typing

```
>>[s d p r] = PAL_ExampleFunction(10,5)
```

creates a variable `s` to store the sum, etc.

Instead of passing values directly to the function, we can assign the values to variables and pass the name of the variables instead. For example the series of commands

```
>>a = 10;
>>b = 5;
>>[sum difference product ratio] = PAL_ExampleFunction(a, b);
```

BOX 1.2 (*cont'd*)

generates the same result as before. You can also assign a single alphanumeric name to vectors and matrices. For example, to create a vector called `vect1` with values 1, -2, 4, and 105 one can simply type and follow with a <return>:

```
>> vect1 = [1 -2 4 105]
```

Note the square, not round brackets. `vect1` can then be entered as an argument to a routine, provided the routine is set up to accept a 1×4 vector. To create a matrix called `matrix1` containing two columns and three rows of numbers, type and follow with a <return>, for example

```
>> matrix1 = [0.01 0.02; 0.04 0.05; 0.06 0.09]
```

where the semicolon separates the values for different rows. Again, `matrix1` can now be entered as an argument, provided the routine accepts a 3×2 (rows by columns) matrix.

Whenever a function returns more than one argument, we do not need to assign them all to a variable. Let's say we are interested in the sum and the difference of two matrices only. We can type:

```
>>[sum difference] = PAL_ExampleFunction([1 2; 3 4], [5 6; ...  
7 8]);
```

Demonstration Programs

A separate set of Palamedes routines are suffixed by `_Demo`. These are located in the folder `PalamedesDemos` separate from the other Palamedes routines. The files in the `PalamedesDemos` folder are demonstration scripts that in general combine a number of Palamedes function routines into a sequence to demonstrate some aspect of their combined operation. They produce a variety of types of output to the screen, such as numbers with headings, graphs, etc. While these programs do not take arguments when they are called, the user might be prompted to enter something when the program is run, e.g.,

```
>>PAL_Example_Demo  
Enter a vector of stimulus levels
```

Then the user might enter something like `[.1 .2 .3]`. After pressing return there will be some form of output, for example data with headings, a graph, or both.

Error Messages

The Palamedes toolbox is not particularly resistant to user error. Incorrect usage will more often result in a termination of execution accompanied by an abstract error message than it will in a gentle warning or a suggestion for proper usage. As an example, let us pass some

Continued

BOX 1.2 (*cont'd*)

inappropriate arguments to our example function and see what happens. We will pass two arrays to it of unequal size:

```
>>a = [1 2 3];
>>b = [4 5];
>>sum = PAL_ExampleFunction(a, b);
```

MATLAB returns

```
??? Error using ==> unknown
Matrix dimensions must agree.
Error in ==> PAL_ExampleFunction at 15
sum = array1 + array2;
```

This is actually an error message generated by a resident MATLAB function, not a Palamedes function. Palamedes routines rely on many resident MATLAB functions and operators (such as “+”), and error messages you see will typically be generated by these resident MATLAB routines. In this case, the problem arose when `PAL_ExampleFunction` attempted to use the “+” operator of MATLAB to add two arrays that are not of equal size.

BOX 1.3**ACRONYMS USED IN PALAMEDES**

Acronyms used in names for Palamedes routines, their meaning, and the chapters in which they are described

Acronym	Meaning	Chapter
AMPM	Adaptive methods: psi method	5
AMRF	Adaptive methods: running fit	5
AMUD	Adaptive methods: up/down	5
MLDS	Maximum likelihood difference scaling	7
PF	Psychometric function	4
PFBA	Psychometric function: Bayesian	4
PFLR	Psychometric function: likelihood ratio	8
PFML	Psychometric function: maximum likelihood	4, 8
SDT	Signal detection theory	6

References

- Fechner, G., 1860/1966. Elements of Psychophysics. Hilt, Rinehart & Winston, Inc.
- Fründ, I., Haenel, N.V., Wichmann, F.A., 2011. Inference for psychometric functions in the presence of nonstationary behavior. *J. Vis.* 11 (6), 16.
- Gescheider, G.A., 1997. Psychophysics: The Fundamentals. Lawrence Erlbaum Associates, Mahwah, New Jersey.
- Knoblauch, K., Maloney, L.T., 2012. Modeling Psychophysical Data in R. Springer.
- Linares, D., López-Moliner, J., in preparation. Quickpsy: An R Package to Analyse Psychophysical Data.
- Lu, Z.-L., Dosher, B., 2013. Visual Psychophysics. MIT Press, Cambridge, MA.
- Peirce, J.W., 2007. PsychoPy – psychophysics software in Python. *J. Neurosci. Methods* 162 (1–2), 8–13.
- Peirce, J.W., 2009. Generating stimuli for neuroscience using PsychoPy. *Front. Neuroinform.* 2, 10. <http://dx.doi.org/10.3389/neuro.11.010.2008>.
- Prins, N., Kingdom, F.A.A., 2009. Palamedes: MATLAB Routines for Analyzing Psychophysical Data. <http://www.palamedestoolbox.org>.

Classifying Psychophysical Experiments*

Frederick A.A. Kingdom¹, Nicolaas Prins²

¹McGill University, Montreal, Quebec, Canada; ²University of Mississippi, Oxford, MS, USA

OUTLINE

2.1 Introduction	11	2.3.6 “Objective” versus “Subjective”	28
2.2 Tasks, Methods, and Measures	12	2.3.7 “Detection” versus “Discrimination”	29
2.3 Dichotomies	14	2.3.8 “Threshold” versus “Suprathreshold”	31
2.3.1 “Class A” versus “Class B” Observations	14	2.4 Classification Scheme	32
2.3.2 “Type 1” versus “Type 2”	19	Further Reading	33
2.3.3 “Performance” versus “Appearance”	20	Exercises	33
2.3.4 “Forced-Choice” versus “Nonforced-Choice”	24	References	34
2.3.5 “Criterion-Free” versus “Criterion-Dependent”	27		

2.1 INTRODUCTION

This chapter describes various classes of psychophysical procedure and proposes a scheme for classifying them. The aim is not so much to judge the pros and cons of different procedures—this will be dealt with in the next chapter—but to examine how they differ and how they interrelate. The proposed classification scheme is arrived at through a critical

*This chapter was primarily written by Frederick Kingdom.

examination of the familiar “dichotomies” that make up the vernacular of psychophysics, e.g., “Class A” versus “Class B” observations, “Type 1” versus “Type 2” tasks, “forced-choice” versus “nonforced-choice” tasks, etc. These dichotomies do not always mean the same thing to all people, so one of the aims of the chapter is to clarify what each dichotomy means and consider how useful each might be as a category in a classification scheme.

Why a classification scheme? After all, the seasoned practitioner designs his or her psychophysics experiment based on knowledge accumulated over years of research experience, including knowledge as to what is available, what is appropriate, and what is valid given the question about visual function being asked. And that is how it should be. However, a framework that captures both the critical differences as well as intimate relationships between different psychophysical procedures could be useful to newcomers in the field, helping them to select the appropriate experimental design from what might seem a bewildering array of possibilities. Thinking about a classification scheme is also a useful intellectual exercise, not only for those of us who like to categorize things, put them into boxes, and attach labels to them, but for anyone interested in gaining a deeper understanding of psychophysics. But before discussing the dichotomies, consider the components that make up a psychophysics experiment.

2.2 TASKS, METHODS, AND MEASURES

Although the outcome of a psychophysics experiment—typically a set of measurements—reflects more than anything else the particular question about sensory function being asked, other components of the experiment, in particular the stimulus and the observer’s task, must be carefully tailored to achieve the experimental goal. A psychophysics experiment consists of a number of components, and we have opted for the following breakdown: stimulus; task; method; analysis; and measure (Figure 2.1). To illustrate our use of these terms, consider one of the most basic experiments in the study of vision: the measurement of a “contrast detection threshold.” A contrast detection threshold is defined as the minimum amount of contrast necessary for a stimulus to be just detectable. Figure 2.2 illustrates the idea for a stimulus consisting of a patch on a uniform background. The precise form of the stimulus must, of course, be tailored to the specific question about sensory function being asked, so we assume that the patch is the appropriate stimulus. The contrast of the patch can be measured in terms of Weber contrast, defined as the difference between the luminance of the patch and its background, ΔL , divided by the luminance of the background L_b , i.e., $\Delta L/L_b$. The contrast detection threshold is therefore the smallest value of Weber contrast needed to detect the patch. Many procedures exist for measuring a contrast detection threshold, each involving a different task for the observer. Before the advent of digital computers, a common

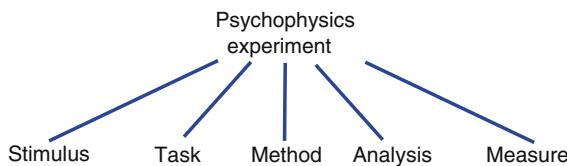


FIGURE 2.1 Components of a psychophysics experiment.

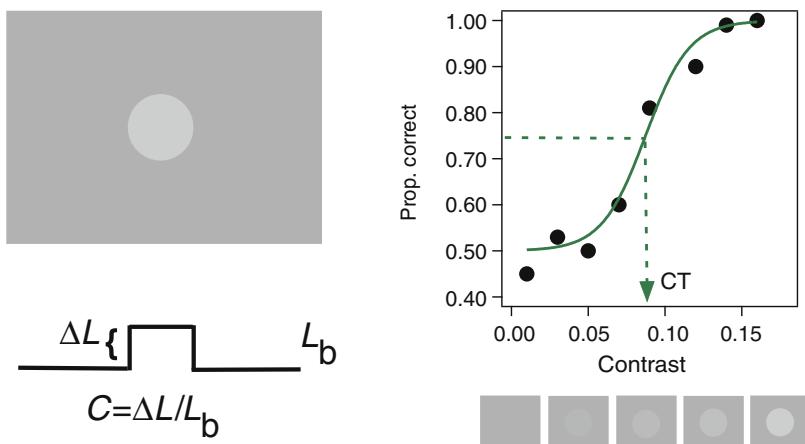


FIGURE 2.2 Top left: circular test patch on a uniform background. Bottom left: luminance profile of the patch and the definition of Weber contrast. Right: results of a standard two-interval-forced-choice (2IFC) experiment. The various stimulus contrasts are illustrated on the abscissa. Black circles are the proportion of correct responses for each contrast. The green curve is the best fit of a psychometric function, and the calculated contrast detection threshold (CT) is indicated by the arrow. See text for further details. L = luminance; L_b = luminance of background; ΔL = difference in luminance between patch and background; C = Weber contrast.

method was to display the stimulus on an oscilloscope and ask observers to adjust the contrast with a dial until the stimulus was just visible. The just-visible contrast would then be recorded as the contrast detection threshold. This method is typically termed the “method of adjustment”, or MOA.

Nowadays the preferred approach is to present stimuli on a computer display and use a “two-interval forced-choice,” or 2IFC, task. Using this procedure, two stimuli are presented briefly on each trial, one of which is a blank screen, the other the test patch. The order of stimulus presentation—blank screen followed by test patch or test patch followed by blank screen—is unknown to the observer (although of course “known” to the computer) and is typically random or quasi-random. The two stimuli are presented consecutively, and the observer chooses the interval containing the test patch, indicating his or her choice by pressing a key. The computer keeps a record of the contrast of the patch for each trial, along with the observer’s response, which is scored as either “correct” or “incorrect.” A given experimental session might consist of, say, 100 trials, and a number of different patch contrasts would be presented in random order.

With the standard 2IFC task, different methods are available for selecting the contrasts presented on each trial. On the one hand, they can be preselected before the experiment—for example, 10 contrasts ranging from 0.01 to 0.1 at 0.01 intervals. If preselected in this way, the 10 stimuli at each contrast would be presented in random order during the session, making 100 trials in total. This is known as the “method of constants.” At the end of each session the computer calculates the number of correct responses for each contrast. Typically, there would be a number of sessions and the overall proportion correct across sessions for each patch contrast calculated, then plotted on a graph as shown for the hypothetical data in Figure 2.2. On the other hand, one could use an “adaptive” (or “staircase”) method, in which the contrast selected on each trial is determined by

the observer's responses on previous trials. The idea behind the adaptive method is that the computer "homes in" on the contrasts that are close to the observer's contrast detection threshold, thus not wasting too many trials on stimuli that are either too easy or too hard to see. Adaptive methods are the subject of Chapter 5.

The term "analysis" refers to how the data collected during an experiment are converted into measures. For example, with the method of adjustment the observer's settings might be averaged to obtain the threshold. On the other hand, using the 2IFC procedure in conjunction with the method of constants, the proportion correct data may be fitted with a function whose shape is chosen to match the data. The fitting procedure can be used to estimate the contrast detection threshold defined as the proportion correct, say 0.75 or 75%, as shown in [Figure 2.2](#). Procedures for fitting psychometric functions are discussed in Chapter 4.

To summarize, using the example of an experiment aimed at measuring a contrast detection threshold for a patch on a uniform background, the components of a psychophysical experiment are as follows. The "stimulus" is a uniform patch of given spatial dimensions and of various contrasts. Example "tasks" include adjustment and 2IFC. For the adjustment task, the "method" is the method of adjustment, while for the 2IFC task one could employ the method of constants or an adaptive method. In the case of the method of adjustment, the "analysis" might consist of averaging the set of adjustments, whereas for the 2IFC task it might consist of fitting a psychometric function to the proportion correct responses as a function of contrast. For the 2IFC task in conjunction with an adaptive method, the analysis might involve averaging contrasts, or it might involve fitting a psychometric function. The "measure" in all cases is a contrast detection threshold, although other measures may also be extracted, such as an estimate of the variability or "error" on the threshold and the slope of the psychometric function.

The term "procedure" is used ubiquitously in psychophysics and can refer variously to the task, method, analysis, or some combination thereof. Similarly, the term "method" has broad usage. The other terms in our component breakdown are also often used interchangeably. For example, the task in the contrast detection threshold experiment, whether adjustment or 2IFC, is sometimes termed a "detection" task and sometimes a "threshold" task, while in our taxonomy the terms "detection threshold" refer to the output measure. The lesson here is that one needs to be flexible in the use of psychophysics terminology and not overly constrained by any predefined scheme.

Next we consider some of the common dichotomies used to characterize different psychophysical procedures and experiments. The aim here is to introduce some common terminology, illustrate other varieties of psychophysical experiment besides contrast detection, and to examine which, if any, of the dichotomies might be candidates for a psychophysics classification scheme.

2.3 DICHOTOMIES

2.3.1 "Class A" versus "Class B" Observations

An influential dichotomy introduced some years ago by [Brindley \(1970\)](#) is that between "Class A" and "Class B" psychophysical observations. Although one rarely hears these terms today, they are important to our understanding of the relationship between psychophysical measurement and sensory function. Brindley used the term "observation" to describe the

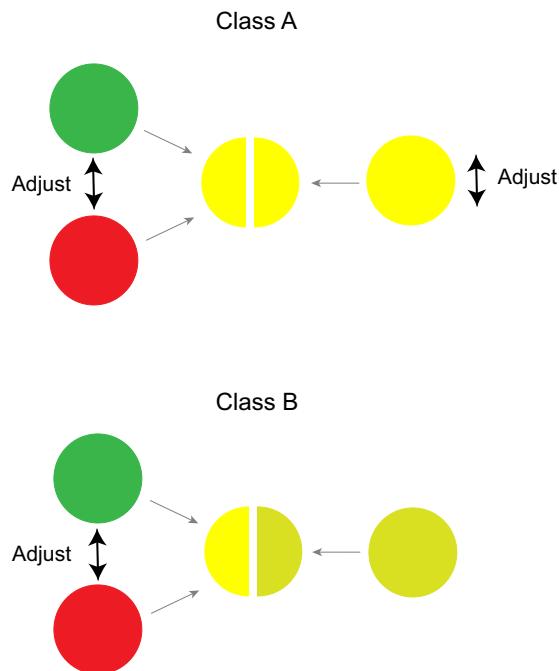


FIGURE 2.3 The Rayleigh match illustrates the difference between a Class A and Class B psychophysical observation. For Class A, the observer adjusts both the intensity of the yellow light in the right half of the bipartite field as well as the relative intensities of the red and green lights in the mixture in the left half of the bipartite field until the two halves appear identical. For Class B, the observer adjusts only the relative intensities of the red and green lights in the left half to match the hue of a yellow light in the right half that in this example is different in brightness.

perceptual state of an observer while executing a psychophysical task. The distinction between Class A and Class B attempted to identify how directly a psychophysical observation related to the underlying mental processes. Brindley framed the distinction in terms of a comparison of sensations: a Class A observation refers to the situation in which two physically different stimuli are perceptually indistinguishable, whereas a Class B observation refers to all other situations.

The best way to understand the difference between Class A and Class B is with an example, and for this we have adopted Gescheider's (1997) example of the Rayleigh match (Rayleigh, 1881; Thomas and Mollon, 2004). Rayleigh matches are used to identify and study certain types of color vision deficiency (e.g., Shevell et al., 2008), but for the present purposes the aim of a Rayleigh match is less important than the nature of the measurement itself. Figure 2.3 shows a bipartite circular stimulus, one half consisting of a mixture of red and green monochromatic lights, the other half a yellow monochromatic light.¹ During the

¹Because the lights are monochromatic, i.e., narrow band in wavelength, this experiment cannot be conducted on a CRT (cathode ray tube) monitor, because CRT phosphors are relatively broadband in wavelength. Instead an apparatus is required that can produce monochromatic lights, such as a Nagel Anomaloscope or a Maxwellian view system.

measurement procedure the observer is given free reign to adjust both the intensity of the yellow light as well as the relative intensities of the red and green lights. The task is to adjust the lights until the two halves of the stimulus appear identical, as illustrated in the top of the figure. In color vision, two stimuli with different spectral (i.e., wavelength) compositions but that appear identical are termed “metamers.” According to Brindley, metamerically matches such as the Rayleigh match are Class A observations. The identification of an observation as Class A accords with the idea that when two stimuli appear identical to the eye they elicit identical neural responses in the brain. Since the neural responses are identical, Brindley argues, it is relatively straightforward to map the physical characteristics of the stimuli onto their internal neural representations.

An example of a Class B observation is shown at the bottom of [Figure 2.3](#). This time the observer has no control over the intensity of the yellow light, only control over the relative intensities of the red and green lights. The task is to match the hue (or perceived chromaticity) of the two halves of the stimulus but with the constraint that the intensity (or brightness) of the two halves remains different. Thus, the two halves will never appear identical and therefore, according to Brindley, neither will the neural responses they elicit. Brindley was keen to point out that one must not conclude that Class B observations are inferior to Class A observations: our example Class B observation is not a necessary evil due to defective equipment! On the contrary, we may wish to determine the spectral combinations that produce hue matches for stimuli that differ in brightness, precisely to understand how hue and brightness interact in the brain. In any case, the aim here is not to judge the relative merits of Class A and Class B observations (for a discussion of this see [Brindley, 1970](#)) but rather to illustrate what the terms mean.

What other types of psychophysical experiment are Class A and Class B? According to Brindley, experiments that measure thresholds, such as the contrast detection threshold experiment discussed in the previous section, are Class A. This might not be intuitively obvious, but the argument goes something like this. There are two states: stimulus present and stimulus absent. As the stimulus contrast is decreased to a point where it is below threshold, the observation passes from one in which the two states are discriminable to one in which they are indiscriminable. The fact that the two states may not be discriminable even though they are physically different (the stimulus is still present even though below threshold) makes the observation Class A. Two other examples of Class A observations that accord to the same criterion are shown in [Figure 2.4](#).

Class B observations characterize many types of psychophysical procedure. Following our example Class B observation in [Figure 2.3](#), any experiment that involves matching two stimuli that are perceptibly different on completion of the match is Class B. Consider, for example, the brightness-matching experiment illustrated in [Figure 2.5](#). The aim of this experiment is to determine how the brightness, i.e., perceived luminance, of a test disk is influenced by the luminance of its surround. As a rule, increasing the luminance of a surround annulus causes the disk inside to decrease in brightness, i.e., become dimmer. One way to measure the amount of dimming is to adjust the luminance of a second, matching disk until it appears equal in brightness to the test disk. The matching disk can be thought of as a psychophysical “ruler.” When the matching disk is set to be equal in brightness to the test disk, the two disks are said to be at the “point of subjective equality,” or PSE. The luminances of the test and match disks at the PSE will not necessarily be the same; indeed it is precisely because they are as a rule different that is of interest. The difference in

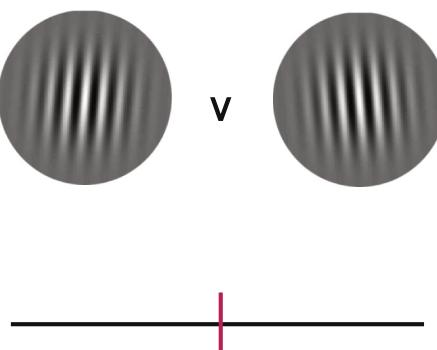


FIGURE 2.4 Two other examples of Class A observations. Top: orientation discrimination task. The observer is required to discriminate between two gratings that differ in orientation, and a threshold orientation difference is measured. Bottom: line bisection task. The observer is required to position the vertical red line midway along the horizontal black line. The precision or variability in the observer's settings is a measure of his or her line-bisection acuity.

luminance between the test and match disks at the PSE tells us something about the effect of context on brightness, the “context” in this example being the annulus. This type of experiment is sometimes referred to as “asymmetric brightness matching,” because the test and match disks are situated in different contexts (e.g., [Blakeslee and McCourt, 1997](#); [Hong and Shevell, 2004](#)).

It might be tempting to think of an asymmetric brightness match as a Class A observation, on the grounds that it is quite different from the Class B version of the Rayleigh match described above. In the Class B version of the Rayleigh match, the stimulus region that

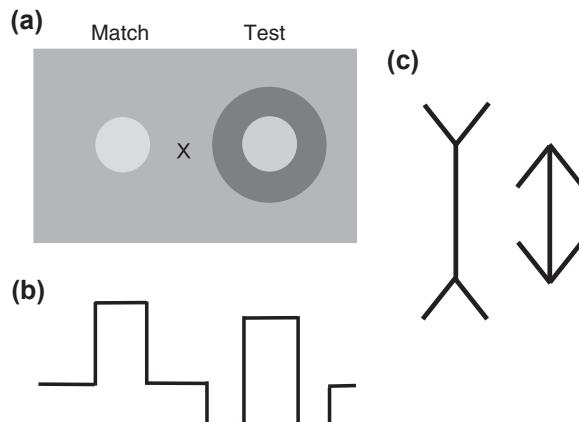


FIGURE 2.5 Two examples of Class B observations. In (a) the goal of the experiment is to find the point of subjective equality (PSE) in brightness between the fixed test and variable match patch as a function of the luminance (and hence contrast) of the surround annulus; (b) shows the approximate luminance profile of the stimulus; (c) is the Muller–Lyer illusion. The two center lines are physically identical but appear different in length. The experiment described in the text measures the relative lengths of the two vertical axes at which they appear equal in length.

observers match in hue is also the region that differs along the other dimension—brightness. In an asymmetric brightness-matching experiment on the other hand, the stimulus region that observers match, brightness, is not the region that differs between the test and match stimuli - in this instance it is the annulus. However, one cannot “ignore” the annulus when deciding whether the observation is Class A or Class B simply because it is not the part of the stimulus to which the observation is directed. Asymmetric brightness matches are Class B because, even when the stimuli are matched, they are recognizably different by virtue of the fact that one stimulus has an annulus and the other does not.

Another example of a Class B observation is the Muller–Lyer illusion shown in [Figure 2.5\(c\)](#), a geometric illusion that has received considerable attention (e.g., [Morgan et al., 1990](#)). The lengths of the axes in the two figures are the same, yet they appear different due to the arrangement of the fins at either end. One of the methods for measuring the size of the illusion is to require observers to adjust the length of the axis, say of the fins-inward stimulus, until it matches the perceived length of the axis of the other, say fins-outward stimulus. The physical difference in length at the PSE, which could be expressed as a raw, proportional, or percentage difference, is a measure of the size of the illusion. The misperception of relative line length in the Muller–Lyer figures is a Class B observation, because even when the lengths of the axes are adjusted to make them perceptually equal, the figures remain perceptibly different as a result of their different fin arrangements.

Another example of a Class B observation is magnitude estimation. This is the procedure whereby observers provide a numerical estimate of the perceived magnitude of a stimulus, for example along the dimension of contrast, speed, depth, size, etc. Magnitude estimation is Class B because our perception of the stimulus and our judgment of its magnitude utilize different mental modalities.

An interesting case that at first defies classification into Class A or Class B is illustrated in [Figure 2.6](#). The observer’s task is to discriminate the mean orientation of two random arrays of line elements, whose mean orientations are right- and left-of-vertical (e.g., [Dakin, 2001](#)). Below threshold, the mean orientations of the two arrays are indiscriminable, yet the two arrays are still perceptibly different by virtue of their different element arrangements. In the previously mentioned Class B examples, the “other” dimension—brightness in the case of the Rayleigh match, annulus luminance in the case of the brightness-matching experiment—was relevant to the task. However in the mean-orientation-discrimination experiment the “other” dimension—element position—is irrelevant. Does the fact that

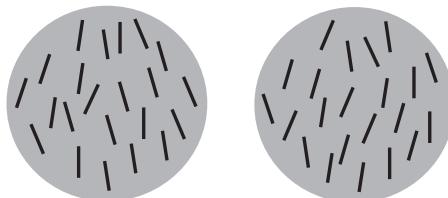


FIGURE 2.6 Class A or Class B? The observer’s task is to decide which of the two stimuli contains elements that are on average left-oblique. When the difference in mean element orientation is below threshold, the stimuli are identical in terms of their perceived mean orientation, yet are discriminable on the basis of the arrangement of their elements.

element arrangement is irrelevant make it Class A, or does the fact that the stimuli are discriminable below threshold on the basis of element arrangement make it Class B? Readers can decide.

In summary, the Class A versus Class B distinction is important for understanding the relationship between psychophysical measurement and sensory function. However, we choose not to use this dichotomy as a basis for classifying psychophysics experiments, in part because there are cases that seem hard to classify in terms of Class A or Class B, and in part because other dichotomies for us better capture the critical differences between psychophysical experiments.

2.3.2 “Type 1” versus “Type 2”

An important consideration in sensory measurement concerns whether or not an observer’s responses can be designated as “correct” or “incorrect”. If they can be so designated, the procedure is termed Type 1 and if not Type 2 (Sperling, 2008; see also Sperling et al., 1990). The term Type 2 has sometimes been used to refer to an observer’s judgments about their own Type 1 decisions (Galvin et al., 2003); in this case, the Type 2 judgment might be a rating of, say, 1–5, or a binary judgment such as “confident” or “not confident,” in reference to their Type 1 decision².

The forced-choice version of the contrast threshold experiment described earlier is a prototypical Type 1 experiment, whereas the brightness-matching and Muller–Lyer illusion experiments, irrespective of whether or not they employ a forced-choice procedure, are prototypical Type 2 experiments. There is sometimes confusion, however, as to why some forced-choice experiments are Type 2. Consider again the Muller–Lyer illusion experiment. As with the contrast detection threshold experiment, there is more than one way to measure the size of the illusion. We have already described the adjustment procedure. Consider how the Muller–Lyer might be measured using a forced-choice procedure. One method would be to present the two fin arrangements as a forced-choice pair on each trial, with the axis of one fixed in length and the axis of the other variable in length. Observers would be required on each trial to indicate the fin arrangement that appeared to have the longer axis. Figure 2.7 shows hypothetical results from such an experiment. Each data point represents the proportion of times the variable-length axis is perceived as longer than the fixed-length axis, as a function of the length of the latter. At a relative length of 1, meaning that the axes are physically the same, the observer perceives the variable axis as longer almost 100% of the time. However, at a relative axis length of about 0.88, the observer chooses the variable axis as longer only 50% of the time. Thus, the PSE is 0.88. However, even though the Muller–Lyer experiment, like the contrast threshold experiment, can be measured using a forced-choice procedure, there is an important difference between the two experiments. Whereas in the contrast detection threshold experiment there is a correct and an incorrect response on every trial, there is no correct or incorrect response for the Muller–Lyer trials. Whatever response the observer makes on a Muller–Lyer trial, it is meaningless to score it as correct or incorrect, at least given the goal of the experiment, which is to measure a PSE. Observers unused to doing psychophysics often have difficulty grasping this idea and even when told repeatedly

²Note that the dichotomy is not the same as Type I and Type II errors in statistical inference testing.

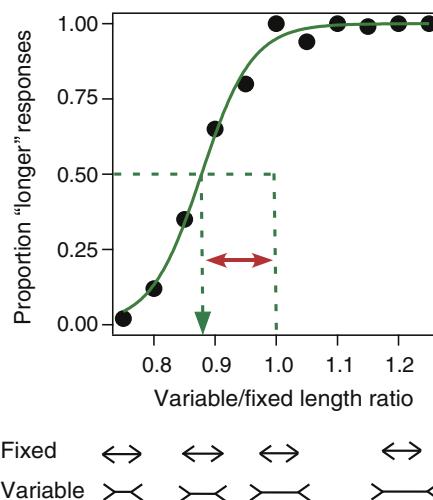


FIGURE 2.7 Results of a hypothetical experiment aimed at measuring the size of the Muller–Lyer illusion using a forced-choice procedure and the method of constant stimuli. The critical measurement is the PSE between the lengths of the axes in the fixed test and variable comparison stimuli. The graph plots the proportion of times subjects perceive the variable axis as “longer.” The continuous line through the data is the best-fitting logistic function (see Chapter 4). The value of 1.0 on the abscissa indicates the point where the fixed and variable axes are physically equal in length. The PSE is calculated as the variable axis length at which the fixed and variable axis lengths appear equal, indicated by the vertical green arrow. The horizontal red-arrowed line is a measure of the size of the illusion.

that there are no correct and incorrect answers, insist on asking at the end of the experiment how many trials they scored correct!

The Type 1 versus Type 2 dichotomy is not synonymous with Class A versus Class B, though there is some overlap. For example, the Rayleigh match experiment described above is Class A but Type 2 because no “correct” match exists. On the other hand, the two-alternative forced-choice (2AFC) contrast threshold experiment is both Class A and Type I.

The Type 1 versus Type 2 dichotomy is an important one in psychophysics. It dictates, for example, whether observers can be provided with feedback during an experiment, such as a tone for an incorrect response. However, one should not conclude that Type 1 is “better” than Type 2. The importance of Rayleigh matches (Class A but Type 2) for understanding color deficiency is an obvious case in point.

2.3.3 “Performance” versus “Appearance”

A dichotomy related to Type 1 versus Type 2, but differing from it in important ways, is that between “performance” and “appearance.” Performance-based tasks measure aptitude, i.e., “how good” an observer is at a particular task. For example, suppose one measures contrast detection thresholds for two sizes of patch, call them “small” and “big.” If thresholds for the big patch are found to be lower than those for the small patch, one can conclude that observers are better at detecting big patches than small ones. By the same token, if orientation discrimination thresholds are found to be lower in central than in peripheral

vision, one can conclude that orientation discrimination is better in central vision than in the periphery. Both of the above tasks aim to establish the limits of our perception. On the other hand, suppose we measure the size of the Muller–Lyer illusion for two different fin angles, say 45° and 60° (relative to the axis), and find that the illusion is bigger for the 45° fins. It would be meaningless to conclude that we are “better” at the Muller–Lyer task when it has 45° compared to 60° fins. PSEs are not aptitudes. For this reason the Muller–Lyer experiment is best considered as measuring stimulus appearance. A simple heuristic can be used to decide whether a psychophysical procedure measures performance or appearance. If the end measurement can be meaningfully considered as showing greater aptitude for one condition than another, then it is measuring performance, and if not, appearance. This still leaves open the question of a precise definition of appearance, other than “not performance.” The term appearance, however, is not easy to define, but for most of the situations described in this book appearance can be defined as the apparent magnitude of a stimulus dimension.

Sometimes the same psychophysical procedure can be used to measure both performance and appearance. Consider the Vernier alignment task illustrated in Figure 2.8, applied to two stimulus arrangements, labelled A and B. The goal of the experiment using stimulus A is to measure Vernier acuity, which is defined as the smallest misalignment that can be detected. This is a threshold measure and hence a performance measure. The goal of the experiment using stimulus B is to measure the effect of the flanking white lines on the perceived position of the black lines. The white lines in B tend to have a small repulsive effect, causing the black lines to appear slightly shifted from their normal perceived position, in a direction away from that of the white lines (e.g., Badcock and Westheimer, 1985). For both experiments, however, the task is the same: decide on each trial whether the upper black line lies to the left (or to the right) of the lower black line.

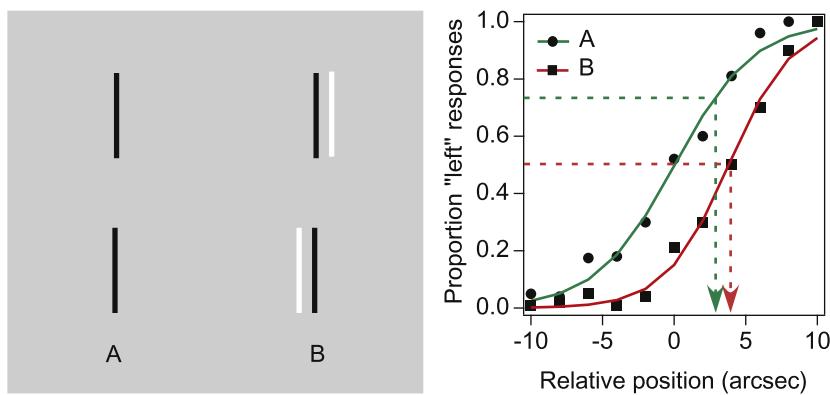


FIGURE 2.8 Left: stimulus arrangements A and B for two Vernier alignment experiments. Right: hypothetical data from each experiment. The abscissa plots the horizontal physical separation between the black lines, with positive values indicating that the top line is physically to the left of the bottom line and negative values indicating that the top line is physically to its right. The ordinate gives the proportion of times the observer responds that the top line is “left.” The continuous curves are best-fitting logistic functions. The green arrow indicates for stimulus A the Vernier threshold and the red arrow indicates for stimulus B the point-of-subjective-alignment.

Hypothetical data for A and B are shown in the graph on the right. The data points have been fitted with logistic functions, as described in Chapter 4. For the A data, Vernier acuity can be calculated as the line separation producing a proportion of 0.75 “left” responses, indicated by the green arrow. Sometimes, however, the observer will perceive the two lines as aligned even when they are physically misaligned. In other words, the point-of-subjective-alignment, or PSA will not be zero. A nonzero PSA may result from optical aberration in the eye or because the observer’s internal representation of space is nonveridical or because the monitor display is physically distorted. It therefore makes more sense to measure Vernier acuity as the separation (or half the separation) between the points on the abscissa corresponding to the 0.25 and 0.75 response levels, as this takes into account any nonzero PSA. Alternatively, the measure of Vernier acuity can be the steepness, or slope, of the psychometric function. As mentioned earlier, the slope of the psychometric function is inversely related to the standard deviation of the function used to fit the data, so the standard deviation is therefore also a measure of (the inverse of) Vernier acuity (e.g., [Watt and Morgan, 1983](#); [McGraw et al., 2004](#)). Recall also that the standard deviation is a measure of precision, with a smaller standard deviation indicating a higher precision. Whether the threshold or slope is used as the measure of Vernier acuity, however, both are performance measures since the “better than” heuristic applies. Note, however, that because the PSA might be nonzero, it is best not to regard the experiment using stimulus A as Type 1, i.e., as having a correct and an incorrect response on each trial. Why? Suppose that when physically aligned, an observer perceives the upper line as slightly to the left of the lower line. On trials where the upper line is presented slightly to the right, the observer will tend to respond “left” and if the experiment is treated as Type I, scored “incorrect.” If correct-versus-incorrect feedback is provided to the observer this will inevitably cause confusion—after all, the observer really did see those lines as “left”—and the confusion could be detrimental to performance.

The fact that a performance measure such as Vernier acuity is best measured without feedback exemplifies how the distinction between performance and appearance is not synonymous with Type 1 and Type 2. Moreover, precision, which we have argued is a performance measure, can be obtained from any Type 2 experiment measuring a PSE. Other examples of performance measures not necessarily derived from Type 1 experiments are contrast detection thresholds obtained using the method of adjustment, measures of accuracy (see next paragraph), and measures of reaction time. Thus, although all Type 1 experiments measure performance, not all performance measures are obtained from Type 1 experiments. On the other hand, all experiments that measure appearance are Type 2.

Not only the precision but also the bias in the Vernier alignment experiment using stimulus A can be considered as a measure of performance. The bias is measured in relation to the true physical alignment, and so one can define the accuracy of the measure as its closeness to the true alignment. Accuracy is important to vision, for example when estimating distances and other spatial relationships as one navigates the visual world. For the Vernier experiment, the bigger the bias the lower the accuracy. A similar argument holds for the line bisection task illustrated in [Figure 2.4](#). In this case, accuracy is how close the observer’s mean setting is to the physical midpoint of the line. Since one can legitimately argue that one observer is more accurate than another in either Vernier alignment or line bisection, the accuracy measured in

these tasks is a performance measure. However, as we shall now see, measures of bias in many circumstances are better considered to be measures of appearance.

Consider the Vernier alignment task using stimulus B. As with the Muller–Lyer and brightness-matching experiments, it is the bias that we are primarily interested in. We want to know by how much the PSA is shifted by the presence of the white lines. The shift in the PSA is measured as the separation between the PSAs for stimuli A and B, with each PSA calculated as the point on the abscissa corresponding to 50% “left” responses. Assuming that the PSA with stimulus A is at zero, the shift in PSA caused by the white lines is indicated by the green arrow on the graph associated with stimulus B. This shift is a measure of appearance.

Innumerable aspects of stimulus appearance avail themselves to psychophysical measurement, for example choosing the computer sketch of a stimulus that best matches its appearance (e.g., [Georges, 1992](#)); indicating when a simulated three-dimensional random-dot rotating cylinder appears to reverse direction (e.g., [Li and Kingdom, 1999](#)); adjusting the colors of a moving chromatic grating until the grating appears to almost stop ([Cavanagh et al., 1984](#)); and labeling contour-defined regions in images of natural scenes as being either “figure” or “ground” (e.g., [Fowlkes et al., 2007](#)). Are there any broad classes of procedure that measure appearance? Matching and scaling experiments are arguably example classes. Matching experiments measure PSEs between two physically different stimuli, as in the Rayleigh match, brightness-matching, Muller–Lyer, and Vernier task B experiments described above. Scaling experiments, the topic of Chapter 8, determine the relationship between the perceived and physical dimensions of a stimulus. Example perceptual scales are the relations between perceived and physical contrast, hue (or perceived chromaticity) and wavelength, perceived and physical velocity, and perceived depth and retinal disparity. Although not all perceptual scales are appearance-based, most of them are.

Example data from a scaling experiment are shown in [Figure 2.9](#). Unlike the hypothetical data used so far to illustrate generic experimental results, every perceptual scale has a unique shape, so for [Figure 2.9](#) we have reproduced a specific case from an experiment conducted by [Whittle \(1992\)](#). Whittle was interested in the relationship between the brightness (or perceived luminance) and the physical luminance of discs on a gray background. Observers were presented with a display consisting of 25 discs arranged in a spiral, with the first and last fixed in luminance at respectively the lowest and highest available on the monitor, corresponding to “black” and “white.” Observers adjusted the luminances of the remaining 23 discs until they appeared to be at equal intervals in brightness. [Figure 2.9](#) plots the disc number (1–25) against the resulting luminance settings. If brightness (the perceptual dimension) was linearly related to luminance (the physical dimension) then the function would be a straight line. Instead it has a complex shape. The reason for this particular shape is another story (see [Kingdom and Whittle, 1996](#)); our aim here is merely to illustrate a type of perceptual scale. There are many different procedures for deriving perceptual scales, and these are summarized in Chapter 3, with further details in Chapter 8.

Both performance-based and appearance-based experiments are important to our understanding of vision. Measures from both types of experiment are necessary to characterize the system. The relationship between performance and appearance, and the question as to what each tells us about visual function, is an important but complex issue that is beyond the remit of this book (e.g., in some instances they appear to measure

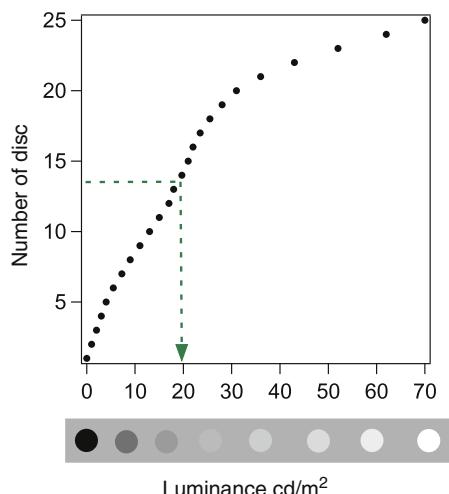


FIGURE 2.9 Data from a brightness scaling experiment. The graph plots the number of the disc against its luminance, after the luminances of all the discs have been adjusted to make them appear at equal brightness intervals. The green arrow indicates the point where the discs change from being decrements to increments. *Data based on Whittle (1992).*

closely related sensory processes, such as the luminance-discrimination threshold and brightness scaling results compared in Whittle (1992), while in other instances they deal with different processes, as argued by Gheorghiu and Kingdom (2008) in relation to curvature perception). However, we argue that the performance versus appearance dichotomy more than any other dichotomy is the principle dividing line in psychophysics. For this reason we propose it as the candidate for the superordinate division in our classification scheme. In the next section, we discuss a possible second level of categorization in the scheme.

2.3.4 “Forced-Choice” versus “Nonforced-Choice”

By now the reader should be familiar with the concept of the forced-choice procedure, but as with many of the terms in psychophysics, the devil lies in the details. In particular, there are different conventions as to when one should and when one should not use the term “forced-choice” and different conventions for the number of alternatives/intervals that prefix the term. In Signal Detection Theory (Wickens, 2002; McNicol, 2004; Macmillan and Creelman, 2005), the subject of Chapters 6 and 7, “forced-choice” is mainly used to characterize experiments in which two or more stimulus alternatives are presented during a trial, one of which is the “target.” Example forced-choice tasks that accord with this usage are: deciding which of two stimuli, a blank field or a patch, contains the patch; deciding which of two patches is brighter; and deciding which of three lines, two oriented -5° and one oriented $+5^\circ$, is the -5° line. In these examples, the observer is required to select a stimulus from two or more stimuli during each trial. Typically, at the end of the experiment the proportion of trials in which the target alternative was selected is calculated for each stimulus

magnitude. Recall that the measure derived from these proportions may be a performance measure, such as a threshold, or an appearance measure, such as a PSE.

In the signal detection literature, most other types of discrimination task are not explicitly referred to as forced-choice, perhaps to avoid the term becoming redundant. Take the procedure termed “yes/no,” in which only one stimulus is presented per trial. [Figure 2.10](#) illustrates the procedure when applied to a contrast detection threshold experiment, along with the two-stimulus-per-trial version (2AFC), explicitly referred to as forced-choice. In the yes/no experiment, the target is normally presented on half the trials and the observer responds “yes” or “no” on each trial, depending on whether they see the target. Although yes/no experiments figure prominently in the signal detection literature, they are not widely employed today in visual psychophysics; the 2AFC procedure is generally preferred for reasons discussed later and in Chapters 3 and 6. The more popular type of single-stimulus-per-trial experiment is the variety we term here “symmetric,” meaning that the stimulus alternatives are akin to mirror images, that is are “equal and opposite.” Example symmetric one-stimulus-per-trial experiments include the orientation discrimination task illustrated in [Figure 2.4](#) (grating left-oblique versus grating right-oblique) and the Vernier task A in [Figure 2.8](#) (upper line to the left versus upper line to the right). Although in the Vernier alignment experiment two lines are presented to the observer on each trial, one must think of the experiment as an example of “single stimulus alternative.” As with the yes/no task, Signal Detection Theory does not generally refer to symmetric single-stimulus-per-trial experiments as forced-choice.

We argue here that it is important to distinguish between procedures that require forced-choice responses and those that do not. Therefore, in this book, we have adopted the convention of referring to any procedure as forced-choice if the observer has two or more prespecified response options to choose from. According to this convention, a yes/no experiment is forced-choice because there are two response options: “yes” and “no”, and the single-alternative-per-trial orientation-discrimination and Vernier acuity experiments described above are also forced-choice. We refer to this convention as the response-based definition of forced-choice. Readers may prefer to think of the response-based definition of forced-choice in terms of choices between “stimulus states,” for example in the yes/no

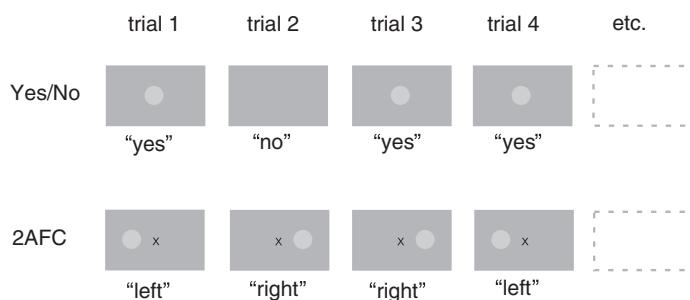


FIGURE 2.10 Yes/no versus 2AFC (two-alternative forced-choice) procedures. In the yes/no task the two alternatives—“stimulus present” and “stimulus absent”—are presented on separate trials, whereas in the 2AFC task they are presented within the same trial. Correct responses are indicated below the stimuli. In this book, both types of task are referred to as “forced-choice.”

experiment between “stimulus present” and “stimulus absent”. As it turns out, the response-based definition of forced-choice is widely used in both the literature and in common parlance, as exemplified by the many single-stimulus-per-trial experiments that are routinely termed “forced-choice” (e.g., [Dakin et al., 1999](#)).

Are there drawbacks to a response-based definition of forced-choice? Consider the method of limits, used mainly to obtain thresholds. Observers are presented with a series of stimuli that are systematically increased (or decreased) in intensity and are prompted to indicate whether or not they can see the stimulus. The stimulus intensity at which the observer switches response from “no” to “yes” (or vice versa) is then taken as the threshold. With a response-based definition of forced-choice, the procedure is arguably forced-choice. Suppose, however, the observer “takes control” of the stimulus presentation and adjusts the stimulus himself/herself. This is normally regarded as the method of adjustment and not forced-choice. But are the two procedures really so different? In both experimenter-controlled and observer-controlled procedures there is no correct and incorrect answer on each stimulus presentation, because the stimulus is always present, albeit with different intensities, so both procedures are Type 2. Moreover, with the observer-controlled adjustment procedure the observer is constantly updating their decision as to whether or not the stimulus is visible, so is this not forced-choice, according to our definition? The example of the method of limits highlights a conundrum for the response-based definition of forced-choice: where does forced-choice end and method of adjustment begin? The resolution of the conundrum lies in a caveat to our definition of forced-choice, namely that the experiment must involve clearly demarcated trials.

Forced-choice tasks are invariably denoted by the abbreviations AFC (alternative forced-choice) or IFC (interval forced-choice). AFC is the generic term, while IFC is reserved for procedures in which the stimulus alternatives are presented in temporal order. Both acronyms are invariably prefixed by a number. In this book, the prefix is the number of stimulus alternatives presented on each trial, denoted by M . The value of M is important for the signal detection analyses described in Chapters 6 and 7, since it relates to the degree of uncertainty as to the target interval/location as well as to the amount of information present during a trial. Because we have adopted the convention of characterizing all tasks that require forced-choice responses as AFC or IFC, we characterize single-stimulus-per-trial procedures such as the yes/no and symmetric single-interval tasks as 1AFC. To spell out our usage, 1AFC means “... a forced-choice task in which only one stimulus alternative is presented per trial.” Readers should be aware, however, that other investigators use the number of response choices as the prefix, at least when referring to single-stimulus-per-trial experiments, where the number of choices is usually 2 (e.g., [Dakin et al., 1999](#)).

An interesting paradox relevant to the choice of M was brought to our attention by Andrew Schofield. Suppose on each trial the observer is presented with two patches a distance apart either side of fixation, one dark the other bright, the task being to select the dark patch. Clearly this is 2AFC. Now bring the two patches together so that they abut, but keep the task the same. Two abutting patches arguably form a single stimulus—a dark-bright edge—implying that the task might now be 1AFC. Yet the only thing that has changed is the distance between the patches. Should the two arrangements of patches be denoted with the same or different M ? Readers may wish to ponder “Schofield’s paradox.”

The other important parameter in forced-choice tasks that we have already discussed is the number of response choices, denoted in this book by m . In most cases M (the number of stimulus alternatives) and m (number of response choices) are the same. For example, in tasks where one interval contains the target and the other a blank field there are two alternatives per trial—blank field and target—and two response choices per trial—“1” (first interval) and “2” (second interval). So M and m are both 2. However, with single-interval tasks there are typically two response choices, i.e., $M = 1$ and $m = 2$. As we have noted above, sometimes m is used as the prefix for a forced-choice task, leading to single-interval tasks being denoted 2AFC (e.g., [Dakin et al., 1999](#)), rather than 1AFC as here.

Our choice of M rather than m as the prefix for a forced-choice task is a concession to Signal Detection Theory, where the distinction between single-interval/alternative and two-interval/alternative tasks needs to be explicit. Nevertheless, m is an important parameter, as it determines the guessing rate in a forced-choice task. The guessing rate is the proportion of times an observer is expected to be correct if simply guessing and is hence calculated as $1/m$. For example, the guessing rate in both a yes/no and 2AFC task is 0.5, assuming the proportion of target-present trials is 0.5. The guessing rate is a critical parameter when fitting psychometric functions, as we shall see in Chapter 4.

A third important parameter in forced-choice tasks is the number of stimuli presented per trial, denoted here by N . Again, in most procedures N is the same as M (and hence m). However, in some forced-choice tasks, such as the “same-different” task that will be discussed in more detail in Chapters 3 and 6, the values of N and M are not the same. Same-different tasks in vision research typically use either two or four stimuli per trial. In the $N = 2$ version, the two stimuli on each trial are either the same or are different, and the observer is required to respond “same” or “different.” In the $N = 4$ version, a same pair and a different pair are presented on each trial, usually in temporal order, and the observer responds “1” or “2,” depending on the interval perceived to contain the same (or different) pair. In both the $N = 2$ and $N = 4$ same-different tasks, the number of response alternatives, m , is 2, and the number of stimulus alternatives, M , is respectively 1 and 2. Values of N , m , and M for a variety of different psychophysical tasks are given in Table 6.1 in Chapter 6.

2.3.5 “Criterion-Free” versus “Criterion-Dependent”

It is often said that the yes/no task described above is “criterion-dependent,” whereas the 2AFC/2IFC task is “criterion-free.” What does this dichotomy mean? Characterizing yes/no tasks as criterion-dependent captures the fact that observers typically adopt different criteria as to how strong the internal signal must be before they respond “yes,” irrespective of the actual strength of the internal signal. If a strict criterion is adopted, the observer will only respond “yes,” when the internal signal is relatively strong, whereas if a loose criterion is adopted a weak signal is sufficient. The adoption of a particular criterion might result from an unconscious bias, or it might be part of a conscious strategy. For example, observers might consciously bias their responses toward “yes” because they want to maximize the number of correct target detections or “hits,” even if this results in a number of “false alarms,” i.e., “yes” responses when the target is absent. On the other hand, they might consciously adopt a strict criterion in order to minimize the number of false alarms, even if this means fewer hits.

2AFC/2IFC tasks can also be prone to bias but in a different way. The bias in this instance is toward responding “1” (first alternative/interval) or toward “2” (second alternative/interval). However, biases of this sort are less common because the two response choices are on an “equal footing.” With 2AFC/2IFC the observer knows that on every trial a target will be presented, so the option of consciously trading off hits and false alarms does not arise.

When biases occur in forced-choice tasks the sensitivity of an observer cannot be measured simply as the proportion correct responses. Chapter 6 explains why this is so and describes an alternative measure, d' (“d-prime”), that is arguably more valid under such circumstances.

There is, however, another more general meaning to the terms criterion-free and criterion-dependent. Occasionally, one hears that Type 1 tasks are criterion-free and Type 2 tasks are criterion-dependent. This usage has parallels with the objective–subjective dichotomy that is described in the next section, so we will discuss it implicitly there.

2.3.6 “Objective” versus “Subjective”

Although rarely put into print, the terms objective and subjective are common parlance among psychophysicists, so it is worth examining their meanings. The terms tend to be value-laden, with objective being “good” and subjective “bad.” Whether or not this is intended, the objective versus subjective dichotomy is inherently problematic when applied to psychophysics. All psychophysical experiments are in one sense subjective, because they measure what is going on inside the head, and if this is the intended meaning of the term, then the objective–subjective dichotomy as applied to psychophysics is redundant. However, the dichotomy is often used in reference to other dichotomies, for example Class A versus Class B, Type 1 versus Type 2, forced-choice versus nonforced-choice, and criterion-dependent versus criterion-free.

Take Type 1 versus Type 2. For some researchers, judgments that cannot be evaluated as correct or incorrect are more subjective (or less objective) than those that can. This view stems from the fact that Type 1 judgments are evaluated against an external benchmark; the stimulus on each trial really is present or absent, or really is left- or right-oblique. The benchmark for tasks where there is no correct or incorrect response, on the other hand, is purely internal; the line only appears to be longer, or the patch only appears to be brighter.

For other researchers, however, the objective–subjective distinction is more to do with the method of data collection than the nature of the measurement itself. Some argue that forced-choice methods are inherently more objective than nonforced-choice methods, irrespective of whether they are Type 1 or Type 2. According to this point of view, both the contrast detection threshold and Muller–Lyer illusion experiments are more objective when using a forced-choice than an adjustment procedure.

Why might forced-choice experiments be considered more objective than nonforced-choice experiments? A potential reason is that forced-choice methods provide more accurate estimates of thresholds and PSEs than those obtained from nonforced-choice methods. Accuracy, in this context, refers to how close the measure is to its “true” value. How does one determine whether one method is more accurate than another? This is not an easy question to answer, particularly for PSEs. Another possible reason why forced-choice methods might be considered more objective is that they are more precise, where precision refers to the variability in

the measurement. With the method of adjustment, precision is typically calculated from the variance or standard deviation of the observer's settings, with a small standard deviation implying high precision. With forced-choice methods, precision is typically measured by the steepness or slope of the psychometric function (see [Figure 2.7](#)). The slope of the psychometric function is inversely proportional to the standard deviation parameter in the function used to fit the data, so again a small standard deviation implies high precision (see Chapter 4 for details). In principle, therefore, one could compare the precisions of adjustment and forced-choice procedures and on this basis decide whether one method is more objective than the other. However, even this is problematic. Suppose, for example, that the forced-choice procedure proved to be the more precise, but the experiment took much longer. One could argue that the superior precision was due to the longer experimental time, not the difference in method per se.

All of the above arguments lead us to conclude that the distinction between objective and subjective is too loosely defined and inherently problematic to use as a basis for classifying psychophysical experiments.

2.3.7 “Detection” versus “Discrimination”

The terms “detection” and “discrimination” are used variously to characterize tasks, measures, procedures, and experiments. For example, one might carry out a “detection experiment” using a “detection task” to obtain a “detection measure.” The term detection is most frequently used to characterize experiments that measure thresholds for detecting the presence of a stimulus, for example a contrast detection threshold. However, the “null” stimulus in a detection experiment is not necessarily a blank field. In curvature detection experiments the null stimulus is a straight line, as illustrated at the top of [Figure 2.11](#). Similarly, in stereoscopic depth detection experiments, the null stimulus lies at a depth of zero, i.e., in the fixation plane, and in a motion detection experiment the null stimulus is stationary.

The term discrimination, on the other hand, is generally reserved for experiments in which neither of the two discriminands (the stimuli being discriminated) is a null stimulus. Thus, in a curvature discrimination experiment, illustrated at the bottom of [Figure 2.11](#), both stimuli in the forced-choice pair are curved, and the task is to decide which stimulus

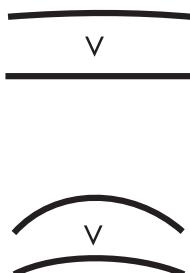


FIGURE 2.11 Top: the task is to identify which of the two stimuli is curved. The task is sometimes termed curvature detection, sometimes curvature discrimination. Bottom: the task is to identify which stimulus is the more curved. This task is invariably termed curvature discrimination.

is more curved. Similarly, in a stereoscopic depth discrimination experiment both stimuli have nonzero depth, and the task is to decide which is nearer (or further). In a motion discrimination experiment both stimuli are moving, and the task is to decide which is moving faster (or slower).

This being said, the terms detection and discrimination tend to be interchangeable. For example, the curvature task illustrated at the top of [Figure 2.11](#) is sometimes termed curvature detection ([Kramer and Fahle, 1996](#)) and sometimes curvature discrimination (e.g., [Watt and Andrews, 1982](#)), even though one of the discriminands is a straight line. Consider also the contrast discrimination experiment illustrated in [Figure 2.12](#). The aim here is to measure the just-noticeable difference (JND) in contrast between two, above threshold stimuli. Typically, one of the contrasts, say the one on the left in the figure, is fixed and termed the baseline or pedestal contrast. The other stimulus is varied to find the JND. One can think of this experiment in two ways. On the one hand it measures a discrimination threshold between two contrasts, while on the other hand it measures a detection threshold for an increment in contrast added to a pedestal. In [Figure 2.12](#) the pedestal and pedestal-plus-increment are presented to the observer at the same time, a procedure sometimes termed the “pulsed-pedestal” paradigm (e.g., [Lutze et al., 2006](#)). Alternatively, the pedestals are first presented together, and then after a short duration the increment is added to one of the pedestals, a procedure that has been termed the “steady-pedestal” paradigm (e.g., [Lutze et al., 2006](#)). One could make the argument that the two paradigms should be considered discrimination and detection, respectively, but in reality there is no hard-and-fast rule here and both paradigms could be considered as either detection or discrimination. Be prepared to be flexible in the use of these terms!

Two psychophysical terms closely related to detection and discrimination are “recognition” and “identification.” The term recognition is generally used in experiments involving relatively complex stimuli such as faces, animals, and household objects, where the task is to select from two or more objects an object either recently shown or long ago memorized. For example, in a prototypical face-recognition experiment, a briefly presented test face is

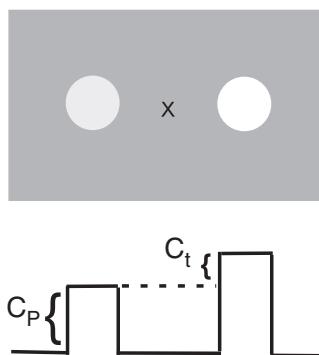


FIGURE 2.12 The task of the subject is to indicate the patch with the higher contrast. The lower contrast patch on the left is fixed in contrast and is termed the pedestal contrast C_p . The variable contrast patch is the one on the right. The task can be regarded as either contrast “discrimination” or contrast increment “detection.” The contrast increment is the test contrast, C_t .

followed by two or more comparison faces from which the observer must choose the test face (e.g., [Wilbraham et al., 2008](#)). This type of procedure is known as “match-to-sample.” Another type of face recognition task requires the observer to simply name a briefly presented famous face (e.g., [Reddy et al., 2006](#)).

The term “identification” is sometimes used instead of recognition and sometimes used instead of discrimination. Probably the most common use of the term is to characterize one of the tasks in experiments in which the discriminands differ along two dimensions, both of which must be discriminated. For example, in the type of experiment termed “simultaneous detection and identification,” the observer is presented with two intervals on each trial (i.e., 2IFC), one containing the target and the other a blank field. However, the target is one of two types of stimuli, e.g., red or green, or moving left or moving right, or near or far. The observer is required to make two judgments on each trial: one to select the interval containing the stimulus and the other to select the type of stimulus. The first judgment is usually termed detection, while the second is either termed discrimination (e.g., [Watson and Robson, 1981](#)) or identification (e.g., [Kingdom and Simmons, 1998](#)). Typically, the aim of the experiment is to decide whether the psychometric functions derived from the two types of decision are significantly different (see Chapter 9 for details).

2.3.8 “Threshold” versus “Suprathreshold”

Our final dichotomy. As with the terms detection and discrimination, “threshold” and “suprathreshold” can refer to experiments, tasks, procedures, or measures. In sensory science a threshold is roughly defined as the stimulus magnitude required to produce a new perceptual state. Traditionally, psychophysical thresholds have been divided into two categories: “absolute” and “difference.” An absolute threshold is the magnitude of a stimulus that can be just discriminated from its null, as exemplified by a contrast detection threshold ([Figure 2.12](#)). A difference threshold, on the other hand, is the magnitude of stimulus difference needed to discriminate two stimuli that are both above their individual absolute thresholds, as exemplified by a contrast discrimination threshold ([Figure 2.12](#)).

Both of the above threshold measures are performance measures. However, not all thresholds are performance measures. Consider the phenomenon of binocular rivalry. Binocular rivalry is said to occur when different stimuli presented to the two eyes are perceived to alternate in dominance (e.g., [Papathomas et al., 2005](#)). A threshold for binocular rivalry can be defined as the minimum physical difference between the stimuli needed to produce rivalry. This is an appearance measure.

The term suprathreshold has more than one definition. One definition is that it is any nonthreshold experiment, task, procedure, or measure. According to this definition the contrast matching and Muller–Lyer experiments described above are suprathreshold, but the contrast discrimination, Vernier acuity, and curvature discrimination experiments are not, because they measure thresholds. However, the term suprathreshold can also refer to any experiment/task/procedure/measure that involves stimuli that are all individually above their own detection threshold. According to this definition, the contrast discrimination, Vernier acuity, and curvature discrimination experiments are also suprathreshold. Once again, one has to be prepared to be flexible when interpreting these terms.

2.4 CLASSIFICATION SCHEME

The first four levels of our proposed scheme are illustrated in Figure 2.13; a fifth level is added in the next chapter. Let us recap the meaning of these categories. Any experiment, task, or procedure is performance-based if it measures something that affords a comparison in terms of aptitude. Thus, a contrast detection experiment is performance-based, because it affords the claim that contrast sensitivity is better in central compared to peripheral vision. Similarly, Vernier acuity affords the claim that Vernier acuity is better in the young than in the old, and so also speed discrimination because it affords the claim that speed discrimination is better at low than at high speeds. Appearance-based experiments, on the other hand, measure the apparent magnitude of some stimulus dimension. Thus, an experiment that measures the Muller–Lyer illusion measures the apparent difference in line length between the two figures, while the asymmetric brightness-matching experiment measures the apparent brightness of a patch surrounded by an annulus. Given that the same task can be used to obtain both performance and appearance measures, the performance-versus-appearance dichotomy speaks primarily to the “goal” of a psychophysical experiment and the “measure” it provides. We regard performance and appearance measures of sensory function as equally important to our understanding of sensory processes, and in the rest of the book we have attempted to balance their respective treatments.

Thresholds (which here include precisions) are the best-known performance measures, but performance measures also include proportion correct, d 's (d -primes), measures of accuracy, and reaction times. The most common appearance-based measures are PSEs (derived from matching procedures) and perceptual scales (derived from scaling procedures). Therefore, the third level in the scheme highlights thresholds, accuracies, reaction times, PSEs, and scales.

The fourth-level division into forced-choice and nonforced-choice is intended to shift the emphasis of the scheme from the measurement goal of a psychophysical experiment to its

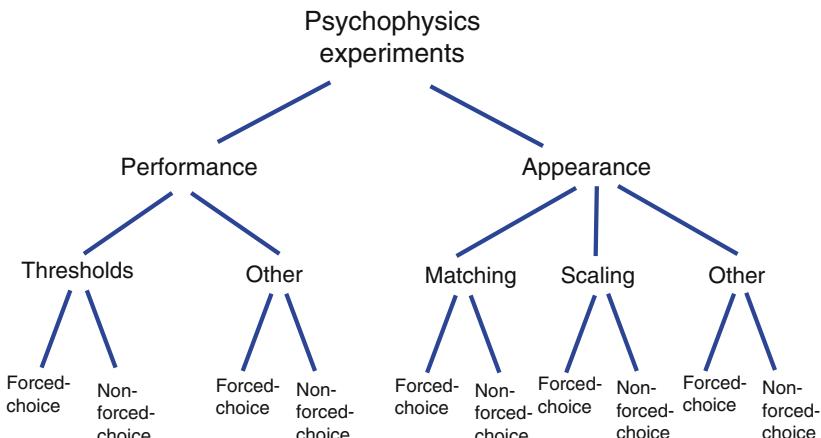


FIGURE 2.13 The initial stages of a scheme based on the performance-appearance distinction. An expanded version of the scheme is provided in the following chapter.

procedural form. In the next chapter a fifth level is added, a division by the number of stimuli presented per trial, providing the final framework for systematically examining a wide range of psychophysical procedures.

FURTHER READING

A discussion of Brindley's distinction between Class A and Class B observations can be found in [Brindley \(1970\)](#) and [Gescheider \(1997\)](#). Sperling has written a short guide to Type 1 and Type 2 ([Sperling, 2008](#)), although see also [Galvin et al. \(2003\)](#) for a somewhat different interpretation. Discussions of yes/no versus forced-choice procedures from the standpoint of Signal Detection Theory can be found in [McNicol \(2004\)](#), [MacMillan and Creelman \(2005\)](#), and [Wickens \(2002\)](#). A good example of the congruency of threshold and scaling measures can be found in [Whittle \(1992\)](#), while a discussion of the incongruity between performance and appearance measures can be found in the study of curvature perception in the introduction of [Gheorghiu and Kingdom \(2008\)](#).

EXERCISES

1. Categorize the following observations as Class A or Class B.
 - a. Choosing a previously shown face from a set of five alternatives (a match-to-sample face recognition task).
 - b. Deciding whether a particular purple is more reddish or more bluish.
 - c. Measuring the effect of contrast on the perceived speed of a moving object.
 - d. Measuring the just-noticeable-difference between the lengths of two lines.
 - e. Naming a briefly presented famous face.
 - f. Measuring the reaction time to the onset of a grating.
 - g. Measuring the threshold for identifying that an image of an everyday scene has been artificially stretched.
 - h. Measuring the duration of the motion-after-effect (the illusory reversed motion seen in an object following adaptation to a moving object).
2. Which of the following could be measured using a Type 1 forced-choice task (i.e., with a correct and an incorrect response on each trial)?
 - a. Estimating the perceived speed of a moving pattern.
 - b. Bisecting a line into two equal halves.
 - c. Deciding whether a particular purple is more reddish or more bluish.
 - d. Measuring the just-noticeable-difference between the curvature of two lines.
 - e. Discriminating male from female faces.
3. Make a table with nine rows labeled by the dichotomies described in the chapter and six columns a–f. For each of the following tasks, consider which alternative in each dichotomy, if at all, is appropriate and include your answer in the table.
 - a. The observer adjusts the contrast of a patch until it looks just-noticeably-brighter than another patch.
 - b. The observer presses a button in response to a decremental change in contrast and his/her reaction time is measured.

- c. The observer chooses from two colors the one appearing more yellowish.
- d. The observer adjusts the speed of a drifting grating until it matches the perceived speed of another drifting grating with a different spatial frequency (the spatial frequency of a grating is the number of cycles of the grating per unit visual angle).
- e. The observer selects on each trial which of two depth targets appears to lie in front of the fixation plane.
- f. The observer identifies whether the face presented on each trial is male or female.

References

- Badcock, D.R., Westheimer, G., 1985. Spatial location and hyperacuity: the centre/surround localization contribution has two substrates. *Vision Res.* 25, 1259–1267.
- Blakeslee, B., McCourt, M.E., 1997. Similar mechanisms underlie simultaneous brightness contrast and grating induction. *Vision Res.* 37, 2849–2869.
- Brindley, G.S., 1970. Physiology of the Retina and Visual Pathway. Williams and Wilkens, Baltimore, MD.
- Cavanagh, P., Tyler, C.W., Favreau, O.E., 1984. Perceived velocity of moving chromatic gratings. *J. Opt. Soc. Am. A* 1, 893–899.
- Dakin, S.C., 2001. Information limit on the spatial integration of local orientation signals. *J. Opt. Soc. Am. A* 18, 1016–1026.
- Dakin, S.C., Williams, C.B., Hess, R.F., 1999. The interaction of first- and second-order cues to orientation. *Vision Res.* 39, 2867–2884.
- Fowlkes, C.C., Martin, D.R., Malik, J., 2007. Local figure-ground cues are valid for natural images. *J. Vis.* 7 (8), 1–9.
- Galvin, S.J., Podd, J.V., Draga, V., Whitmore, V., 2003. Type 2 tasks in the theory of signal detectability: discrimination between correct and incorrect decisions. *Psychon. Bull. Rev.* 10, 843–876.
- Georges, M.A., 1992. Human vision combines oriented filters to compute edges. *Proc. R. Soc. B* 249, 235–245.
- Gescheider, G.A., 1997. Psychophysics: The Fundamentals. Lawrence Erlbaum Associates, Mahwah, NJ.
- Gheorghiu, E., Kingdom, F.A.A., 2008. Spatial properties of curvature-encoding mechanisms revealed through the shape-frequency and shape-amplitude after-effects. *Vision Res.* 48, 1107–1124.
- Hong, S.W., Shevell, S.K., 2004. Brightness induction: unequal spatial integration with increments and decrements. *Vis. Neurosci.* 21, 353–357.
- Kingdom, F.A.A., Simmons, D.R., 1998. The missing-fundamental illusion at isoluminance. *Perception* 27, 1451–1460.
- Kingdom, F.A.A., Whittle, P., 1996. Contrast discrimination at high contrasts reveals the influence of local light adaptation on contrast processing. *Vision Res.* 36, 817–829.
- Kramer, D., Fahle, M., 1996. A simple mechanism for detecting low curvatures. *Vision Res.* 36 (10), 1411–1419.
- Li, H.-C.O., Kingdom, F.A.A., 1999. Feature specific segmentation in perceived structure- from-motion. *Vision Res.* 39, 881–886.
- Lutze, M., Pokorny, J., Smith, V.C., 2006. Achromatic parvocellular contrast gain in normal and color defective observers: implications for the evolution of color vision. *Vis. Neurosci.* 23, 611–616.
- Macmillan, N.A., Creelman, C.D., 2005. Detection Theory: A User's Guide. Lawrence Erlbaum Associates, Mahwah, NJ.
- McGraw, P.V., McKeefry, D.J., Whitaker, D., Vakrou, C., 2004. Positional adaptation reveals multiple chromatic mechanisms in human vision. *J. Vis.* 4, 626–636.
- McNicol, D., 2004. A Primer of Signal Detection Theory. Lawrence Erlbaum Associates, Mahwah, NJ.
- Morgan, M.J., Hole, G.J., Glennerster, A., 1990. Biases and sensitivities in geometric illusions. *Vision Res.* 30, 1793–1810.
- Papathomas, T.V., Kovacs, I., Conway, T., 2005. Interocular grouping in binocular rivalry: basic attributes and combinations (Chapter 9). In: Alais, D., Blake, R. (Eds.), *Binocular Rivalry*. MIT Press, Cambridge, MA.
- Rayleigh, L., 1881. Experiments on colour. *Nature* 25, 64–66.
- Reddy, L., Reddy, L., Koch, C., 2006. Face identification in the near-absence of focal attention. *Vision Res.* 46, 2336–2343.
- Shevell, S.K., Sun, Y., Neitz, M., 2008. Protanomaly without darkened red is deutanopia with rods. *Vision Res.* 48, 2599–2603.

- Sperling, G.B., 2008. Type I and Type II Experiments. http://aris.ss.uci.edu/HIPLab/ProSem202c/UCI_access/READINGS/Type_1_and_Type_2_Expts.pdf.
- Sperling, G., Dosher, B.A., Landy, M.S., 1990. How to study the kinetic depth experimentally. *J. Exp. Psychol. Hum. Percept. Perform.* 16, 445–450.
- Thomas, P.B., Mollon, J.D., 2004. Modelling the Rayleigh match. *Vis. Neurosci.* 21, 477–482.
- Watson, A.B., Robson, J.C., 1981. Discrimination at threshold: labeled detectors in human vision. *Vision Res.* 21, 1115–1122.
- Watt, R.J., Andrews, D.P., 1982. Contour curvature analysis: hyperacuties in the discrimination of detailed shape. *Vision Res.* 22, 449–460.
- Watt, R.J., Morgan, M.J., 1983. The use of different cues in vernier acuity. *Vision Res.* 23, 991–995.
- Whittle, P., 1992. Brightness, discriminability and the “crispening effect”. *Vision Res.* 32, 1493–1507.
- Wickens, T.D., 2002. Elementary Signal Detection Theory. Oxford University Press, Oxford, New York.
- Wilbraham, D.A., Christensen, J.C., Martinez, A.M., Todd, J.T., 2008. Can low level image differences account for the ability of human observers to discriminate facial identity? *J. Vis.* 8 (15), 5.1–5.12.

Varieties of Psychophysical Procedures*

Frederick A.A. Kingdom¹, Nicolaas Prins²

¹McGill University, Montreal, Quebec, Canada; ²University of Mississippi, Oxford, MS, USA

OUTLINE

3.1 Introduction	37	3.4 Further Design Details	52
3.2 Performance-Based Procedures	39	3.4.1 Method of Constant Stimuli	52
3.2.1 Thresholds	39	3.4.2 Adaptive Procedures	53
3.2.2 Nonthreshold Tasks and Procedures	45	3.4.3 Timing of Stimulus Presentation	53
3.3 Appearance-Based Procedures	45	Further Reading	54
3.3.1 Matching	45	References	54
3.3.2 Scaling	48		

3.1 INTRODUCTION

As the proverb goes, the devil lies in the details. In this chapter we delve into the details of psychophysical procedures and consider their relative advantages and disadvantages. By “procedure” we mean both the observer’s task and the method of data collection, in other words the “front-end” of a psychophysics experiment. Subsequent chapters will deal with the “back-end,” i.e., the data. The procedures described in this chapter are organized according to the performance-versus-appearance classification scheme that was advocated in Chapter 2. Figure 3.1 expands the scheme to include a further level of categorization based

*This chapter was primarily written by Frederick Kingdom.

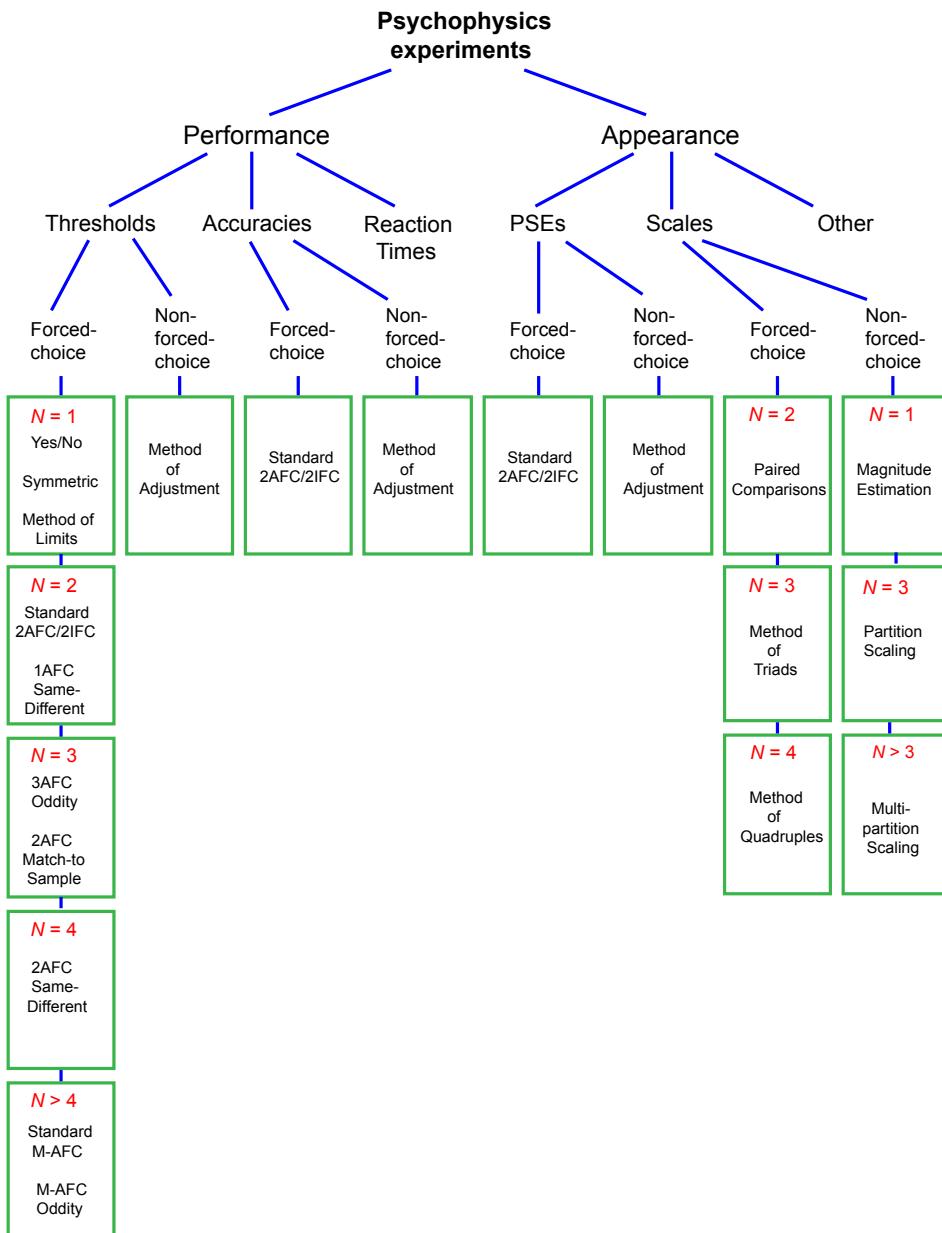


FIGURE 3.1 Expanded scheme for classifying psychophysical experiments.

on the number of stimuli presented per trial, N . N seemed to us to be the natural way to extend the scheme to incorporate the many types of procedure discussed here.

Recall the meaning of the major categories in the classification scheme. Performance-based procedures measure aptitude or “how good one is” at a particular visual task. In Chapter 2

we included in this category most types of threshold measure, proportion correct, d' , measures of precision, accuracies, and reaction times. We pointed out that while many types of performance measure could be obtained using Type 1 tasks, i.e., tasks with a correct and an incorrect response on each trial, not all performance measures were Type 1. For example, reaction times, some measures of precision, measures of accuracy, and certain types of threshold measure could be obtained from Type 2 procedures, i.e., that had no correct or incorrect responses.

Appearance-based procedures, on the other hand, generally measure the apparent magnitude (relative or absolute) of some stimulus dimension. Appearance-based procedures can only be Type 2. This does not imply that appearance-based measurements are less useful or less valid than performance-based measurements for understanding sensory function. Both types of measurement are arguably necessary to characterize the system.

Forced-choice procedures, as defined here, refer to procedures in which the observer is required on each trial to make a response selected from two or more prespecified options. Our definition of forced-choice is not restricted to situations where two or more stimuli are presented per trial; a single-stimulus-per-trial presentation with two response options is here regarded as forced-choice. Moreover, the term forced-choice can apply to both performance-based and appearance-based procedures.

3.2 PERFORMANCE-BASED PROCEDURES

3.2.1 Thresholds

3.2.1.1 **Forced-Choice Threshold Procedures**

Although we have chosen to categorize forced-choice procedures according to the number of stimuli presented per trial, N , recall from Chapter 2 that the acronyms AFC and IFC are not prefixed by N , but M , the number of stimulus alternatives per trial. In many types of forced-choice procedure, N and M are the same, but in some, such as the same-different and match-to-sample procedures discussed later, they are different.

Consider the various ways one might measure an orientation discrimination threshold for grating patches. By way of example, assume that the goal is to measure the minimum discriminable difference in orientation between a left-oblique and a right-oblique patch of grating. The first thing to note is that the potential number of stimuli that could be presented during a trial is infinite. For example, the display screen could be divided into 100 squares using an 11x11 grid of lines, with 99 locations containing, say, the left-oblique grating and one location containing the right-oblique “target” grating. The task for the observer would be to choose the location containing the target, and the response would be scored as either correct or incorrect. In principle this seems fine, but consider what the task would involve. During each trial the observer would need to scan all 100 locations in order to be sure not to miss the target. Therefore, each trial would invariably take several seconds or longer. Assuming one wanted to collect enough data to obtain a reasonable estimate of the threshold, the experiment would take a long time to complete. The procedure therefore seems impractical, unless of course one specifically wants to study how observers perform with large N displays. For most purposes, however, a small value of N is preferable. In the limit $N=1$,

but there can be disadvantages to $N = 1$. In fact, as we shall see, there are both advantages and disadvantages to each of $N = 1, 2, 3, 4$, and $N > 4$.

Figure 3.2 summarizes the common varieties of performance-based forced-choice tasks using small N , applied to the orientation discrimination experiment. Note how the value of M , which prefixes the acronym AFC, is not always the same as N . In the following sections we discuss in turn the various options illustrated in the figure.

3.2.1.1.1 $N = 1$ (ONE STIMULUS PER TRIAL)

METHOD OF LIMITS Although rarely used these days, the method of limits is a simple way to obtain a rough estimate of a threshold. It is probably most useful for getting a handle on the appropriate stimulus levels to use with a more rigorous method. The method of limits may also be desirable in situations where the experimenter needs to maintain close verbal contact with the observer, for example, with young children or clinically impaired persons, or indeed in any circumstance where it is difficult for the observer to be "in the driving seat."

In the method of limits the observer is presented with a series of temporally or spatially demarcated stimuli of increasing (ascending method of limits) or decreasing (descending method of limits) magnitude. The series may also include a null or baseline stimulus at one end of the continuum. If, for example, one wanted to measure a contrast detection threshold, the ascending series might be contrasts of, say, 0, 0.01, 0.02, 0.04, 0.08, etc. For our orientation discrimination example the series might be grating patch orientations of 0, 0.25, 0.5, 0.75, 1.0, 1.25, etc., degrees. On each presentation the observer is required to report "yes" or "no," depending on whether the stimulus appears noticeably different from the null or baseline level (zero in both examples). The threshold in each case is the stimulus magnitude at which the response switches from "no" to "yes" and/or vice versa. This is a Type 2 performance procedure, because the observer's response is never evaluated in terms of whether it is correct or incorrect. Typically, the ascending and descending series are presented alternately and the thresholds from each averaged.

A potential disadvantage of the method of limits is that the observer may become accustomed to reporting that they perceive (or not) a stimulus, and as a result continue to give the same response even at stimulus magnitudes that are higher (or lower) than the "real" threshold. This is termed the error of habituation. Conversely, the observer may anticipate that the stimulus is about to become detectable, or undetectable, and make a premature judgment. This is called the error of expectation. Errors due to habituation and expectation may be minimized by averaging thresholds from ascending and descending series.

YES/NO The yes/no procedure is employed primarily for measuring detection thresholds. Typically, half the trials contain the target stimulus and half the no-target stimulus, and the task for the observer is to respond "yes" or "no" on each trial. Since the responses are evaluated as either correct or incorrect, the procedure is Type 1. As in all forced-choice tasks, the order of presentation of the target-present and target-absent trials must be random, or quasi-random. With quasi-random presentation a rule is introduced that precludes long sequences of either target-present or target-absent trials.

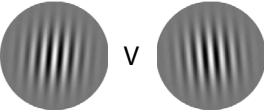
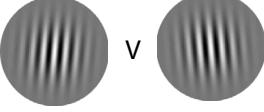
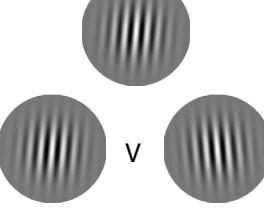
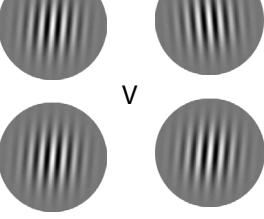
<i>N</i>	Task name	Stimuli per trial	Task
1	1AFC Symmetric		"Left-oblique or right-oblique?"
2	2AFC Standard		"Which one left-oblique?"
2	1AFC Same-Different		"Same or different?"
3	3AFC Oddity		"Which one the oddity?"
3	2AFC Match-to-Sample		"Which of the bottom pair is the same (or different from) the top one?"
4	2AFC Same-Different		"Which pair, top or bottom, is different (or same)?"

FIGURE 3.2 Different methods for measuring an orientation discrimination threshold. *N* = number of stimuli presented on each trial. Note that the number that prefixes the acronym AFC (alternative-forced-choice) is *M*, the number of stimulus alternatives presented per trial.

Yes/no tasks are particularly prone to the effects of bias. Observers may adopt, intentionally or unintentionally, different criteria as to how much sensory evidence they require before being prepared to give a “yes” response. If they adopt a strict criterion, they will respond “yes” only on those trials where they are very confident that the target is present. On the other hand if they adopt a loose criterion, they will respond “yes” on the flimsiest of evidence. Experimenters sometimes use the yes/no task because it is criterion-dependent, for example, in order to study the effect of incentives on performance. The incentive might be to maximize the number of “hits”—these are “yes” responses when the target is present—or minimize the number of “false alarms”—these are “yes” responses when the target is absent. In situations where the observer is biased toward either responding “yes” or “no,” the proportion of correct decisions is a poor measure of how sensitive the observer is to the target stimulus. To circumvent this problem, experimenters typically prefer the signal detection measure d' (“d-prime”) calculated from the proportions of hits and false alarms. The method is described in Chapter 6.

SYMMETRIC $N = 1$ forced-choice procedures can also be used when the two discriminands are “symmetric,” as in the orientation discrimination task illustrated at the top of [Figure 3.2](#). Here, the two discriminands are left- and right-oblique grating patches but with only one of the discriminands presented on a trial. Because the two discriminands are “equal but opposite,” it is less likely that observers will be biased toward responding one more than the other. Hence, many experimenters treat symmetric $N = 1$ tasks as “bias-free” and use proportion correct as the measure of performance. However, to be sure, one should analyze the data in a similar way to the yes/no task, as described in Chapter 6. To minimize the possibility of bias it is important to make observers aware that the discriminands are presented an equal number of times or with equal probability.

The main advantages of the symmetric $N = 1$ task are that a large number of responses can be collected within a relatively short time, and that the task imposes minimum cognitive load. Inexperienced observers often find this task one of the easiest. Typically, the experimenter presents the observer with different stimulus magnitudes during an experimental session, either by the method of constants or by an adaptive procedure (see below).

3.2.1.1.2 $N = 2$

STANDARD 2AFC/2IFC In what is the most popular design in performance-based psychophysics, observers are presented on each trial with two stimuli and are required to select one as the target ([Figure 3.2](#)). In one form of the task the two stimuli are presented together on the screen (2AFC), while in the other the two stimuli are presented in the same display position but in temporal order (2IFC). For a given stimulus exposure time and session time one can gather twice as many responses with 2AFC as compared to 2IFC, but there are potential drawbacks. If the stimuli are intended only to be seen in parafoveal or peripheral vision, 2AFC is the preferred method because when the stimuli are placed on either side of fixation, the observer is less inclined to make an eye movement to one or the other stimulus and as a result unintentionally foveate it. Such a temptation is harder to resist with 2IFC! If, on the other hand, the observer is required to scan both stimuli foveally, then 2IFC is preferable. 2IFC might also be preferable for two other reasons. First, with 2AFC, if the stimulus presentation time is very short (<1 s) observers may become frustrated while attempting to scan the

two stimuli within the time allocated. Second, if the presentation time is very long the time-advantage of 2AFC may be lost. Typically, proportion correct is used as the measure of performance with 2AFC/2IFC, but observers can sometimes be biased toward responding to one location/interval more than the other, in which case proportion correct is not a good measure and d' should be used (Chapter 6).

1AFC SAME-DIFFERENT Observers are presented with a pair of stimuli on each trial, with half the trials containing a pair that is the same and half the trials a pair that is different. The task is to choose the “same” or “different” pair (Figure 3.2). The main reason for using a same-different task is that the observer does not need to know the basis on which the discriminands differ. This is desirable in a number of situations. One situation is when the experimenter is not sure on what basis observers discriminate the stimuli, for example, if the stimuli are faces with different expressions, and the experimenter is reluctant to give precise instructions as to what observers should look for. Another situation is when the experimenter wants to present observers with multiple discriminand pairs from across a wide range of a given stimulus dimension. For example, one might wish to obtain an average measure of orientation discrimination ability across a range of orientations or an average measure of color discrimination ability across a range of colors. In these circumstances it is preferable not to burden observers with having to learn the basis for discriminating each stimulus pair, which is especially problematic with circular dimensions such as orientation or color.

In the 1AFC version of the same-different task only two stimuli are involved, say S_1 and S_2 . Hence there are two Same combinations, S_1S_1 and S_2S_2 , and two Different combinations, S_1S_2 and S_2S_1 . All four combinations are typically presented an equal number of times or with equal probability during a session. Because the two discriminands (Same and Different) are not symmetric, this task is particularly prone to the effects of bias, in this instance a tendency toward responding “same” or a tendency toward responding “different.” Thus, it is advisable to analyze the data to take into account any bias (see Chapter 6). The less-bias-prone 2AFC version of same-different is described later. The 1AFC same-different task is popular in animal experiments for determining an animal’s ability to recognize a previously shown object. When employed for this purpose, the two stimuli are presented in temporal order and the animal is typically rewarded after correctly identifying either the Same or the Different stimulus (e.g., [Vallentin and Nieder, 2008](#)).

3.2.1.1.3 $N = 3$

3AFC ODDITY In the oddity task, sometimes termed “odd-man-out,” all stimuli bar one are the same and the observer selects the stimulus that is different (Figure 3.2). As with the same-different task, an attractive feature of the oddity task is that the observer does not need to know the basis on which the stimuli differ. The minimum N for an oddity task is 3, and this version, sometimes termed the “triangular method,” is undoubtedly the most popular (e.g., [Huang et al., 2006](#)). Oddity tasks can be either three-alternative (3AFC) or three-interval (3IFC). With the 3AFC version the three stimuli are best positioned in a triangular arrangement on the screen (e.g., [Pitchford and Mullen, 2005](#)).

Are there disadvantages to the oddity task? Some observers find it difficult and frustrating. In the case of the 3IFC version, for example, the observer needs to hold in short-term memory

three pieces of information prior to making a decision, and they often report difficulty remembering “what the first stimulus looked like.” The 3AFC version avoids this problem, providing observers are given plenty of time to compare all three stimuli, and probably the most successful version of the oddity task is the 3AFC version with unlimited stimulus exposure. However, many experimenters prefer the 2AFC match-to-sample or the 2AFC same-different task to the 3AFC oddity task, for reasons now discussed.

2AFC MATCH-TO-SAMPLE The observer first views a “sample” stimulus and then selects the sample from one of two “match” stimuli. As with the oddity and same-different tasks, the observer does not need to know the basis on which the stimuli differ. Match-to-sample tasks are particularly popular in animal (e.g., [Jordan et al., 2008](#)), child vision ([Pitchford and Mullen, 2005](#)), and cognitive vision studies, such as studies of face recognition (e.g., [Wilbraham et al., 2008](#)). A particularly attractive feature of the match-to-sample task is that it can be used to study recognition memory, since the time delay between sample and match can be varied. Part of the reason for the task’s popularity is that it is easy for human observers to understand and for animals to learn. This may in part be due to the fact that the “same as” concept is easier to grasp than the “different from” concept needed with both the oddity and same-different tasks. The match-to-sample task is also less cognitively demanding than the oddity task, because there is one less alternative to choose from.

3.2.1.1.4 $N = 4$

2AFC/2IFC SAME-DIFFERENT In this form of the same-different task, the two pairs of stimuli, Same and Different, are presented together on a trial, and the observer selects the pair that is Different (or Same). This version of same-different is less prone to bias than the 1AFC version described earlier, and for this reason is arguably preferable. Because there are four stimuli per trial, a popular scenario is to present the two members of each pair together on the display but in temporal order (e.g., [Yoonessi and Kingdom, 2008](#)): presenting four stimuli one after the other is likely be too cognitively demanding. Observers often prefer the 2AFC same-different task to the 3AFC oddity task because, although the former involves one extra stimulus, there is one less alternative to have to choose from.

3.2.1.1.5 $N > 4$

M-AFC TASKS Although small- N forced-choice procedures are generally preferable to large- N ones, some experimental questions demand large N , and for these the standard forced-choice, oddity, and match-to-sample tasks can be used. The M -AFC match-to-sample task in particular is a very flexible tool, offering a myriad of design possibilities. With it one can test the observers’ ability to select a sample from not just two, but from a large number of stimulus states, for example, a red object from an array of green, red, yellow, blue, etc. objects. Moreover, the stimuli can be defined along multiple dimensions, such as color, form, motion, and depth. For example, the observer might be required to select a red T-shape from an array of green O-, yellow B-, red T-, blue Z-, etc. shapes. Another variant is to require observers to select the match that has one attribute in common with the sample even though it differs in other attributes. For example, the match might have the same color

as the sample, but have a different form and motion, or the match might have the same motion as the sample, but a different form and color (e.g., [Pitchford and Mullen, 2001](#)).

3.2.1.2 Nonforced-Choice Thresholds

3.2.1.2.1 METHOD OF ADJUSTMENT

The method of adjustment is rarely used nowadays to obtain performance measures, since forced-choice procedures are easy to set up on a computer and are widely regarded as superior. However, the method of adjustment can be useful for obtaining a rough threshold estimate in order to guide the choice of stimulus magnitudes for an in-depth experiment, especially when there are a large number of different conditions to be measured (e.g., [Nishida et al., 1997](#)).

3.2.2 Nonthreshold Tasks and Procedures

3.2.2.1 Accuracies and Reaction Times

Accuracy refers to how close a measure is to its true value and can be measured using both forced-choice and method-of-adjustment. Examples of accuracy measures are described in Chapter 2.

Reaction times refer to the time taken for an observer to respond to the onset or offset of a stimulus. An important measure of aptitude, reaction times are often an accompaniment to other performance measures such as proportion correct (e.g., [Ratcliff and Rouder, 2009](#)). In visual search, reaction times are widely employed to measure the time observers take to find a target among a set of distractors (e.g., [Treisman and Gelade, 1980](#); [McLlhagga, 2008](#)). The analysis of reaction times and its value for understanding psychological processes is a large topic that is outside the scope of this book; some useful references are given at the end of the chapter.

3.3 APPEARANCE-BASED PROCEDURES

All appearance-based procedures are Type 2, since there can never be a correct or incorrect judgment about appearance. We have chosen to divide appearance procedures into matching and scaling and then subdivide each of these into forced-choice and nonforced-choice. Note, however, as stated in the previous chapter, that matching and scaling procedures constitute only a fraction of the procedures available to measure stimulus appearance. Matching procedures aim to measure the point of subjective equality (PSE) between two stimuli. Although matching procedures can be used to derive perceptual scales, it is scaling procedures that are explicitly designed to uncover the relationship between the perceived and physical magnitudes of a stimulus dimension. We first consider matching procedures.

3.3.1 Matching

In Chapter 2 we described a number of matching experiments. The Rayleigh match determined the combinations of wavelengths that matched a single, narrowband wavelength in

both brightness and hue. The brightness-matching experiment determined the luminance of a disc that matched the brightness of a test disc surrounded by an annulus. The Muller–Lyer illusion experiment determined the length of a line in one figure that matched the perceived length of the line in another. And the Vernier experiment determined the offset of two lines that made them appear aligned. Each experiment measured some form of PSE between physically different stimuli. Although the term “matching” conjures up an image of an observer adjusting something until it looks like something else, three of the above experiments used a forced-choice procedure rather than the method of adjustment. With a forced-choice matching procedure the observer makes a comparative judgment of two stimuli on each trial, for example, which stimulus looks brighter, which stimulus looks longer, etc., but the goal is always to establish a PSE.

3.3.1.1 *Forced-Choice Matching*

3.3.1.1.1 $N = 2$: MATCHING USING 2AFC/2IFC

The reader is again referred to the examples of the brightness-matching, Muller–Lyer, and Vernier acuity experiments described in Chapter 2. There is little to add here except to emphasize that a forced-choice procedure enables the experimenter to derive a full psychometric function, and thus to obtain estimates of parameters besides PSEs and precisions, such as the errors on these parameters. Full details of how to obtain parameter estimates from appearance-based psychometric functions are provided in Chapter 4.

3.3.1.2 *Nonforced-Choice Matching*

3.3.1.2.1 $N = 2$: MATCHING BY ADJUSTMENT

Adjustment is still widely employed to obtain PSEs. Observers freely adjust one stimulus, termed the “match,” “adjustable,” or “variable” stimulus, until it appears equal along the dimension of interest to the “test” stimulus. If enough matches are made the variance or standard deviation of the settings can be used to provide a measure of precision.

3.3.1.2.2 $N = 2$: NULLING BY ADJUSTMENT

A variant on matching that often uses the method of adjustment is “nulling” or “cancellation.” In some instances nulling and matching can be considered to be two sides of the same coin. Consider, for example, the brightness-matching experiment illustrated in Figure 2.5. One can think of the annulus as inducing an “illusory” brightness in the test patch, because even though the luminance of the test patch remains fixed, its brightness changes with the luminance of the annulus. However, instead of the observer adjusting the luminance of the match patch in order to match the brightness of the test patch for each annulus luminance, the observer can instead “null” or “cancel” the effect of the annulus by adjusting the test luminance to match that of the fixed-in-luminance match patch. By the same token, if the observer adjusted the length of the central line of one of the Muller–Lyer figures (say the one with acute fins) until it matched that of the length of the line in the other Muller–Lyer figure (the one with obtuse fins), one could say that the illusion was being nulled or canceled.

The difference between nulling and matching emerges forcefully when applied to the grating-induction illusion illustrated in [Figure 3.3 \(McCourt, 1982\)](#). In the top left figure

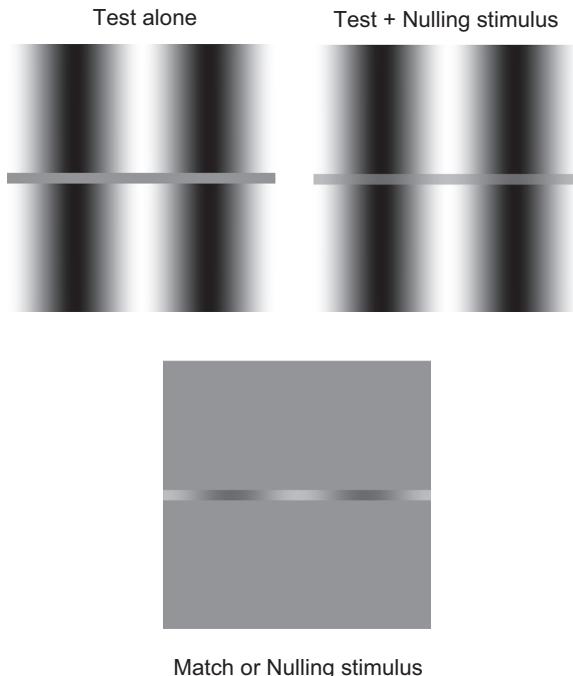


FIGURE 3.3 Matching versus nulling. Top left: grating induction stimulus. The horizontal gray stripe running through the middle of the luminance grating is uniform yet appears modulated in brightness due to simultaneous brightness contrast. Bottom middle: an adjustable grating with similar spatial dimensions to the induced grating can be used to match its apparent contrast. Top right: the same grating, however, can instead be used to null or cancel the induced grating when added to it. Note that the cancellation is not perfect, because of the limitations of reproduction. See text for further details.

one observes an illusory modulation in brightness in the gray stripe that runs horizontally through the grating. The modulation is illusory because the gray stripe is physically uniform in luminance. The illusory modulation is an instance of the well-known phenomenon termed “simultaneous brightness contrast.” Notice how the illusory brightness modulation is out-of phase with the real luminance modulation in the surround grating (i.e., the ordering of bright and dark is opposite). The apparent contrast of the illusory or “induced” modulation depends on a number of factors, and to study these factors one needs a method that measures the size of the induction. Two possible methods are illustrated in [Figure 3.3](#). The matching procedure uses a second grating with similar spatial dimensions to the induced grating, as shown in the bottom middle figure. The observer adjusts the contrast of the matching grating until it appears equal in contrast to that of the induced grating. The contrast of the gratings is typically measured using the metric of contrast known as Michelson contrast, defined as $(L_{\max} - L_{\min})/(L_{\max} + L_{\min})$, where L_{\max} and L_{\min} are the maximum and minimum luminances of the grating. Thus, with the matching procedure, the magnitude of brightness induction is measured by the contrast of the matching grating at the PSE. In the nulling procedure, on the other hand, the second grating is added to the

induced grating and its contrast is adjusted until the induced grating just disappears (McCourt and Blakeslee, 1994)—this is illustrated in the top right figure. Note that with the nulling procedure the phase of the added grating must be opposite to that of the induced grating in order for the cancellation to work, as in the figure (this is not necessary for the matching procedure). With the nulling procedure, the contrast of the nulling grating that cancels the induced grating is the measure of the size of the induction.

3.3.2 Scaling

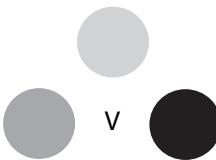
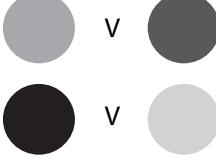
3.3.2.1 Types of Perceptual Scale

Recall that a perceptual scale describes the relationship between the perceptual and physical magnitudes of a stimulus dimension. There are three types of perceptual scale that are most relevant to psychophysics: ordinal, interval, and ratio. In an ordinal perceptual scale, stimulus magnitudes are numbered according to their rank order along a perceptual continuum. However, the difference between any pair of numbers does not necessarily correspond to the magnitude of the perceptual difference. For example, consider a stimulus with three contrasts: 0.1, 0.7, and 0.8. On an ordinal scale these might be numbered 1, 2, and 3, but this does not imply that the perceptual difference between the 0.1 and 0.7 contrasts is the same as the perceptual difference between the 0.7 and 0.8 contrasts. On the contrary, the perceptual differences will likely be different. To represent the perceptual differences between these pairs of contrasts, an interval or ratio scale is required. In an interval scale, the differences between numbers correspond to perceptual differences, even though the numbers themselves are arbitrary. Using the example of the three contrasts above, an interval scale might be 1, 5, and 6. This time the numbers capture the observation that the perceptual difference between the first and second contrasts—a difference of four scale units—is four times greater than the perceptual difference between the second and third contrasts—a difference of one scale unit. However, the interval scale could just as easily be written 4, 12, and 14, since these numbers embody the same difference-relations as the 1, 5, and 6 scale. Formally, an interval scale can be transformed without loss of information by the equation $aX + b$, where X is the scale value, and a and b are constants.

The limitation of an interval scale is that it does not capture the perceived relative magnitudes of the stimulus dimension. For example, interval scale values of 1 and 5 do not indicate that the second value is five times the perceived magnitude of the first. Perceptual scales that capture relative perceived magnitude are known as ratio scales and can be transformed only by the factor aX .

The relationship between perceived and physical contrast is an example of a one-dimensional perceptual scale. However, perceptual scales can be two-dimensional. The best-known example of a two-dimensional perceptual scale is a color space (such as the CIE), in which each color is defined by a point on a plane with an X and a Y coordinate, and where the distance between points corresponds to the perceived distance in hue or perceived chromaticity. Two-dimensional perceptual scales are invariably interval scales. [Figure 3.4](#) summarizes the main varieties of one-dimensional interval-scaling tasks that are now described.

Forced-choice

N	Task name	Stimuli per trial	Task
2	Paired Comparisons		"Which patch is brighter?"
3	Method of Triads		"Which of the bottom pair of patches is most similar (or different) to the top patch?"
4	Method of Quadruples		"Which pair of patches, top or bottom, are more similar (or different)?"

Nonforced-choice

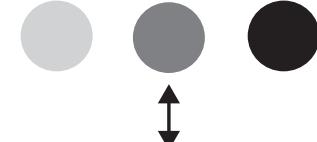
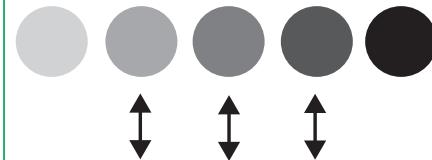
N	Task name	Stimuli per trial	Task
3	Partition Scaling		"Adjust the brightness of the middle patch until it appears mid-way between the anchors either side"
>3	Multi-partition Scaling		"Adjust the brightnesses of the patches between the anchors until all patches are at equal perceptual intervals"

FIGURE 3.4 Types of scaling task for deriving interval scales, applied to the example of brightness scaling. In the nonforced-choice methods in the lower panel the double arrows refer to disks whose luminances are freely adjusted by the observer.

3.3.2.2 **Forced-Choice Scaling Procedures**

3.3.2.2.1 $N = 2$: PAIRED COMPARISONS

The simplest forced-choice method for deriving a perceptual scale is the method of paired comparisons. If the stimulus space is sampled only coarsely, paired comparisons can only provide an ordinal perceptual scale. For example, suppose one wants to rank order, say, 10 photographs of faces according to how happy each face looks. On each trial observers are shown two faces drawn from the set and asked to choose the happier face. There would be a total of $(10^2 - 10)/2 = 45$ possible face pairs or twice this number if every pair was shown in both order. On each trial the face selected to be the happier is given a score of one, while the other face is given a score of zero. By repeating the procedure for all possible pairs of faces, the 10 faces can be rank-ordered by perceived happiness according to their accumulated scores.

In order to generate an interval scale using paired comparisons, however, the different stimulus levels must be close enough to ensure that the responses to any pair are not always the same. Instead, the data must be a “proportion” of times that one member of a pair is chosen over any other. With proportions as data one can estimate the perceptual distances between stimulus levels and hence generate an interval perceptual scale. Chapter 8 describes the details of the paired comparison method, along with the Palamedes routines for generating stimulus lists, simulating observer responses, and analyzing responses to produce an interval perceptual scale using paired comparisons. The chapter also includes a critical discussion of the strengths and limitations of the paired comparison method.

3.3.2.2.2 $N = 3$: METHOD OF TRIADS

The method of triads can also be used to derive either an ordinal or interval scale, but uses judgments of relative perceived similarity (or difference). Unlike the $N = 2$ paired-comparison method, the method of triads does not require prior knowledge of the dimension along which the stimuli differ.

In one version of the method of triads, one of the three stimuli is designated as the target, while the other two are designated as comparisons. The observer is required to compare the perceived similarity (or difference) between the target and each of the two comparisons and choose the pair that is the more (or less) similar. Typically, the pair perceived to be more similar would be given a value of one and the pair perceived to be less similar a value of zero. One can think of this version of the task as the appearance analog of the 2AFC match-to-sample performance task, described earlier in the chapter. In another version of the method of triads there is no designated target, and the observer compares all three possible pairs, ranking them one, two, or three. Chapter 8 provides all necessary details.

3.3.2.2.3 $N = 4$: METHOD OF QUADRUPLES

With quadruples, observers are presented with two pairs of stimuli on each trial and are required to choose the pair that is the more (or less) similar. As with the method of triads, the observer need not know the basis upon which the stimuli differ. Again the details are provided in Chapter 8.

3.3.2.2.4 $N > 4$: MULTISTIMULUS SCALING

An alternative to the paired comparison method for deriving an ordinal perceptual scale is to require observers to arrange the entire set of stimuli in rank order. The best known example of this method is the Farnsworth–Munsell 100 hue test for color deficiency. Observers are presented with a randomly arranged series of disks that vary systematically along a particular color dimension (e.g., green to red) and are asked to arrange them in order of hue (e.g., green, yellowish-green, more-yellowish-green, yellow, reddish-yellow, more-reddish-yellow … red). Their chosen arrangement is then compared to the one typically made by a person with normal color vision. One can think of the order of colors made by a person with normal color vision as the “correct” order, but it is only “correct” in relation to an internal standard, not to a physical standard as with a Type 1 experiment. The pattern of errors made by observers with the Farnsworth–Munsell test can be used to identify certain types of color deficiency.

3.3.2.2.5 MULTIDIMENSIONAL SCALING

Multidimensional scaling (MDS) is used to determine whether two or more perceptual dimensions underlie the perceived similarities between stimuli. Earlier we mentioned the CIE color space as an example of a two-dimensional representation of perceived color similarities. MDS algorithms provide multidimensional arrangements of stimuli in which the distances between stimuli correlate with their perceived dissimilarity. The method of triads and quadruples can be used to generate data for MDS (e.g., [Gurnsey and Fleet, 2001](#)). The analysis of MDS data is, however, outside of the scope of this book, but some example reading material is provided at the end of this chapter.

3.3.2.3 Nonforced-Choice Scaling Procedures

3.3.2.3.1 $N = 1$: MAGNITUDE ESTIMATION

In magnitude estimation the observer makes a direct numerical estimate of the perceived magnitude of a stimulus along the dimension of interest. Magnitude estimation produces a ratio scale if observers are instructed to allocate numbers that reflect the relative perceived magnitudes of the stimuli. In one form of magnitude estimation, the experimenter starts with a stimulus designated as an “anchor” and asks the observer to suppose that it has a perceived magnitude of, say, 50. The magnitudes of the other stimuli are then estimated relative to that of the anchor, i.e., 25 (half as much), 100 (twice as much), 175 (3.5 times as much), etc. The scale values can then be normalized to the stimulus with the lowest perceived magnitude by dividing all values by 50. Psychophysicists tend to regard magnitude estimation as a rather blunt tool, because it requires observers to translate a perceptual experience into a numeric, i.e., symbolic, representation. Observers often find magnitude estimation difficult and unsatisfactory, and for this reason other scaling methods are recommended where possible.

3.3.2.3.2 $N = 3$: PARTITION SCALING

In partition scaling, sometimes termed “euisection” or “bisection” scaling, observers adjust the magnitudes of stimuli in order to make them appear at equal perceptual intervals. Partition scaling methods therefore generate interval scales. There are a variety of partition scaling methods, and the principle behind two of them is illustrated at the bottom of

Figure 3.4. One version that is intuitively easy for the observer, but has some drawbacks, is termed by [Gescheider \(1997\)](#) as the “progressive solution.” The experimenter starts by providing the observer with two “anchors” that define the start and end points of the stimulus dimension. The observer then divides the perceptual distance between the two anchors into two equal parts by adjusting a third stimulus until it appears midway between the anchors.¹ The resulting two intervals are then each bisected in a similar manner, resulting in four intervals, and so on. This method, however, suffers from the problem that errors will tend to accumulate as the intervals become smaller.

3.3.2.3.3 $N > 3$: MULTIPARTITION SCALING

Termed by [Gescheider \(1997\)](#) as the “simultaneous solution” and referred to here as multipartition scaling ([Figure 3.4](#)), observers are first presented with a full range of stimuli together on the display. Two stimuli at the ends of the range serve as anchors, and observers adjust the remaining stimuli until they appear to be at equal perceptual intervals. Recall [Whittle's \(1992\)](#) multipartition scaling experiment described in Chapter 2. The aim of the experiment was to derive an interval scale of brightness for discs of adjustable luminance arranged in the form of a spiral on a uniform gray background. The anchor discs were set to the lowest and highest luminances available on the monitor and observers adjusted the luminances of the remaining discs until they appeared to be at equal brightness intervals. Intuitively, this is not an easy task, since adjustment to any one disc would tend to “throw out” previous adjustments, requiring a number of iterations to achieve a satisfactory solution.

3.4 FURTHER DESIGN DETAILS

3.4.1 Method of Constant Stimuli

In any forced-choice procedure, whether performance-based or appearance-based, the question arises as to how to present the different magnitudes of a stimulus during an experimental session. One popular solution is the method of constant stimuli, or as it is sometimes termed, the “method of constants.” In this method, the stimulus magnitude on each trial is randomly selected from a predefined set. For a performance-based experiment, the range is typically chosen to straddle the expected threshold value in order that performance ranges from near-chance to near-100% correct. For example, in a standard 2AFC procedure with threshold defined at the 75% correct level, performance should range from close to 50% to close to 100%, with roughly equal numbers of stimulus magnitudes producing less than and greater than 75% correct. This generates data that, when fitted with the appropriate psychometric function, provides the most accurate estimates of the threshold as well as other parameters, such as the slope. Full details of the procedures for fitting psychometric functions are described in Chapter 4. The choice of stimulus set usually requires some pilot work to obtain a rough estimate of the threshold, and the method of adjustment is useful for doing this.

¹Note that the bisection scaling task is different from the bisection acuity task described in Chapter 2. The latter is a performance-based task that measures the accuracy and/or precision of the bisection.

The method of constant stimuli can also be used in conjunction with appearance-based procedures. For forced-choice matching experiments in which the PSEs are estimated from a psychometric function, the above considerations equally apply, though this time the data are not proportions correct but proportions of times one stimulus is perceived to be greater than the other along the dimension of interest.

3.4.2 Adaptive Procedures

To avoid the problem of inappropriately chosen stimulus sets, adaptive (or staircase) procedures are often used instead of the method of constant stimuli. In an adaptive procedure the stimulus magnitude on each trial is selected by an algorithm that analyzes the previous trial responses, in some cases to “zero in” on the threshold, in others to pick stimulus intensities that are expected to maximize the information gain regarding the value of the threshold. Some adaptive procedures can be used in conjunction with conventional methods for fitting psychometric functions, enabling estimates of both the threshold and slope to be obtained. Adaptive methods can be used in conjunction with both performance-based and appearance-based tasks. Adaptive procedures are the subject of Chapter 5.

3.4.3 Timing of Stimulus Presentation

The timing of stimulus presentations is very important in psychophysics and for the test observer can make the difference between an experiment that feels difficult and frustrating and one that feels comfortable and engaging. To illustrate what’s at stake, take a prototypical 2IFC task. [Figure 3.5](#) is a schematic of the temporal arrangement and terminology. The example is of an observer-paced trial, in which each trial is triggered by the observer’s response to the previous trial. In general, 2IFC tasks are best when self-paced, as this gives the observer control over the pace of the experiment, without disrupting the critical temporal parameters.

The choice of within-trial temporal parameters is crucial for making a task feel comfortable. For example, if the first stimulus of the forced-choice pair is presented too soon after

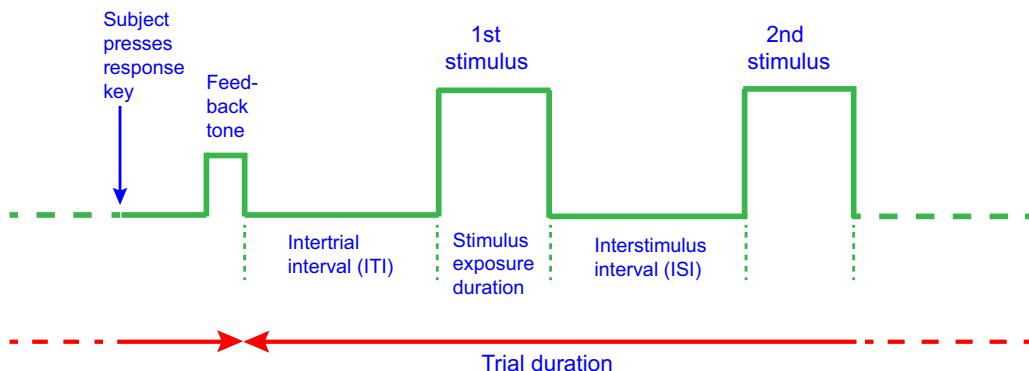


FIGURE 3.5 Example timing of stimulus presentation during a typical 2IFC trial.

the observer responds to the previous forced-choice pair, the observer can become confused as to what his/her response is “attached to”; the response may become associated in the observer’s mind with the stimulus that follows the response rather than with the stimulus that precedes it. An appropriate interstimulus interval (ISI) is also important to minimize both forward and backward masking effects between stimuli. There is no hard and fast rule here, and the experimenter is encouraged to try out different ISIs until the task feels comfortable. As a rule of thumb, a stimulus exposure duration of 250 ms, an ISI of 500 ms, and an intertrial-interval (ITI) of 1000 ms is a good starting point.

FURTHER READING

An excellent and user-friendly introduction to much of what is discussed here can be found in [Gescheider \(1997\)](#). Reviews of the use of reaction times in psychological research can be found in [Pachella \(1974\)](#) and [Meyer et al. \(1988\)](#). Multidimensional scaling is discussed in [Borg and Groenen \(2005\)](#).

References

- Borg, I., Groenen, P.J.F., 2005. Modern Multidimensional Scaling: Theory and Applications. Springer, New York.
- Gescheider, G.A., 1997. Psychophysics: The Fundamentals. Lawrence Erlbaum Associates, Mahwah, NJ.
- Gurnsey, R., Fleet, D.J., 2001. Texture space. *Vision Res.* 41, 745–757.
- Huang, P.-C., Kingdom, F.A.A., Hess, R.F., 2006. Only two phase mechanisms, \pm cosine, in human vision. *Vision Res.* 46, 2069–2081.
- Jordan, K.E., MacLean, E., Brannon, E.M., 2008. Monkeys match and tally quantities across senses. *Cognition* 108, 617–625.
- McCourt, M.E., 1982. A spatial frequency dependent grating-induction effect. *Vision Res.* 22, 119–134.
- McCourt, M.E., Blakeslee, B., 1994. A contrast matching analysis of grating induction and suprathreshold contrast perception. *J. Opt. Soc. Am. A* 11, 14–24.
- McIlhagga, W., 2008. Serial correlations and 1/f power spectra in visual search reaction times. *J. Vis.* 8 (9), 5.1–5.14.
- Meyer, D.E., Osman, A.M., Irwin, D.E., Yantis, S., 1988. Modern mental chronometry. *Biol. Psychol.* 26, 3–67.
- Nishida, S., Ledgeway, T., Edwards, M., 1997. Dual multiple-scale processing for motion in the human visual system. *Vision Res.* 37, 2685–2698.
- Pachella, R.G., 1974. The interpretation of reaction time in information-processing research. In: Kantowitz, B.H. (Ed.), *Human Information Processing: Tutorials in Performance and Cognition*. Lawrence Erlbaum Associates, Hillsdale, NJ, pp. 41–82.
- Pitchford, N.J., Mullen, K.T., 2001. Conceptualization of perceptual attributes: a special case for color? *J. Exp. Child Psychol.* 80, 289–314.
- Pitchford, N.J., Mullen, K.T., 2005. The role of perception, language, and preference in the developmental acquisition of basic color terms. *J. Exp. Child Psychol.* 90, 275–302.
- Ratcliff, R., Rouder, J.N., 2009. Modeling response times for two-choice decisions. *Psychol. Sci.* 9, 347–356.
- Treisman, A.M., Gelade, G., 1980. A feature-integration theory of attention. *Cogn. Psychol.* 12, 97–136.
- Vallentin, D., Nieder, A., 2008. Behavioral and prefrontal representation of spatial properties of monkeys. *Curr. Biol.* 18, 1420–1425.
- Whittle, P., 1992. Brightness, discriminability and the “crispening effect.” *Vision Res.* 32, 1493–1507.
- Wilbraham, D.A., Christensen, J.C., Martinez, A.M., Todd, J.T., 2008. Can low level image differences account for the ability of human observers to discriminate facial identity? *J. Vis.* 8 (15), 5.1–5.12.
- Yoonessi, A., Kingdom, F.A.A., 2008. Comparison of sensitivity to color changes in natural and phasescrambled scenes. *J. Opt. Soc. Am. A* 25, 676–684.

Psychometric Functions*

Frederick A.A. Kingdom¹, Nicolaas Prins²

¹McGill University, Montreal, Quebec, Canada; ²University of Mississippi, Oxford, MS, USA

OUTLINE

4.1 Introduction	55	4.3 Section B: Theory and Details	71
4.2 Section A: Practice	56	4.3.1 Psychometric Function Theories	71
4.2.1 Overview of the Psychometric Function	56	4.3.2 Details of Function Types	79
4.2.2 Number of Trials and Stimulus Levels	57	4.3.3 Methods for Fitting Psychometric Functions	84
4.2.3 Types and Choice of Function	59		
4.2.4 Methods for Fitting Psychometric Functions	64	Further Reading	116
4.2.5 Estimating the Errors	65	Exercises	116
4.2.6 Estimating the Goodness-of-Fit	69	References	117

4.1 INTRODUCTION

Psychometric functions, or PFs, relate the behavior on a given psychophysical task (e.g., proportion of correct responses, proportion of trials perceived brighter) to some physical characteristic of the stimulus (e.g., contrast, length). Typically, although not always, one measures a PF in order to determine one or more parameters that summarize the behavior, e.g., a threshold contrast or a point of subjective equality. Chapter 2 showed examples of PFs and the parameters determined by them. In this chapter we will introduce the reader to methods of determining the chosen parameters from a PF. It is important to note that PFs can be fitted

*This chapter was primarily written by Nicolaas Prins.

to the data for both performance-based and appearance-based psychophysical tasks, and by and large the procedures for fitting PFs are common to both classes of data.

The chapter is divided into two sections. The first section introduces the reader to the general procedures involved in fitting PFs, determining the chosen parameters from the fits, and getting estimates of how well the PFs have been fit as well as the variability of the estimated parameters. The second section considers the underlying theory behind PFs, fitting PFs, and parameter estimation. The reader may choose to read the second section or to skip it without loss of continuity.

4.2 SECTION A: PRACTICE

4.2.1 Overview of the Psychometric Function

[Figure 4.1](#) illustrates the general idea. The figure shows hypothetical data from an experiment aimed at measuring a contrast detection threshold. The data were obtained from a two-interval forced-choice (2IFC) task using the method of constant stimuli. The graph plots the proportion of correct responses for each stimulus contrast. Note that the contrast values on the abscissa are arranged at equal logarithmic intervals. However, other arrangements such as linear spacing of values are often used. The observer performed 100 trials for each stimulus contrast. Threshold contrast is defined as the contrast at which the proportion correct response reaches some criterion, here 0.75 or 75%. In order to obtain the value corresponding to the threshold a continuous function has been fitted to the data. The function in this example is known as a log-Quick function and is one of a variety of functions that can be used to fit PFs. To fit the log-Quick curve to the data the computer iteratively searched through a range of possible values of two parameters, α (alpha) and β (beta). The α parameter determines the overall position of the curve along the abscissa and for the log-Quick function corresponds to the contrast value at which the proportion correct is halfway between the lower and upper asymptote, here 0.75 or 75% correct. The β parameter determines the slope or gradient of the curve. The parameters α and β are properties of the observer and we will never know their exact values. Rather, the fitting procedure found estimates of the values of α and β that

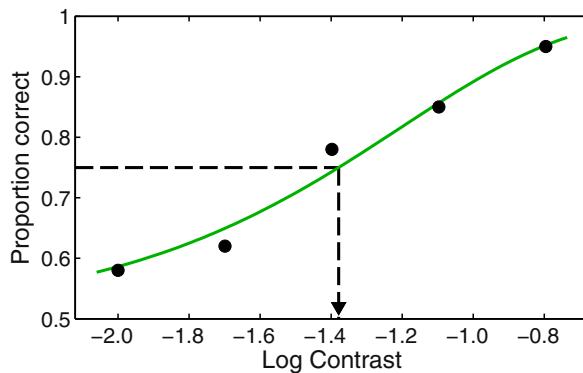


FIGURE 4.1 Example of a PF from a hypothetical experiment aimed at measuring a contrast detection threshold. The threshold is defined here as the stimulus contrast at which performance reaches a proportion correct equal to 0.75. Data are fitted using a log-Quick function.

TABLE 4.1 Six values describing a fitted psychometric function

Threshold $\hat{\alpha}$	Slope $\hat{\beta}$	SE threshold	SE slope	Deviance	<i>p</i> -value
-1.3789	0.9079	0.0704	0.1582	1.1728	0.7652

generated a curve that best matched the experimental data. We use a “hat” over the symbol for a parameter to mean “estimate of” that parameter. Thus, the value of $\hat{\alpha}$ for the best-fitting curve is the estimate of the true contrast threshold, α , and $\hat{\beta}$ is the estimate of the true slope β .

Four additional values make up the complete description of the PF. Two of these, the “standard error” (SE) of the threshold and the SE of the slope, are measures of the precision, or rather imprecision, of $\hat{\alpha}$ and $\hat{\beta}$, i.e., how far they are likely to be from the “true” value of α and β . Put another way, they are estimates of the errors associated with our estimates of α and β . The remaining two measures, “deviance” and its associated *p*-value, are used to determine whether the fitted function provides an adequate model of the data. We will discuss goodness-of-fit briefly in [Section 4.2.6](#) and in much more detail in Chapter 9. [Table 4.1](#) gives the values of all six of these measures for the PF in [Figure 4.1](#).

This example PF illustrates the key components to measuring and fitting a PF. In summary these are (1) choosing the stimulus levels; (2) selecting the function to fit the data; (3) fitting the function; (4) estimating the errors on the function’s parameter estimates; and (5) determining the goodness-of-fit of the function. In what follows we consider these components in turn.

4.2.2 Number of Trials and Stimulus Levels

4.2.2.1 Number of Trials

How many trials are needed to estimate a PF? As a rule, the more trials there are the more accurate the estimates will be on the fitted parameters, such as the threshold, slope, or point of subjective equality (PSE). So the answer to the question primarily rests on how precise one wants one’s parameter estimates to be. If one anticipates that different experimental conditions will produce very different thresholds or PSEs, then this can be demonstrated in fewer trials than if one anticipates that there will be slight differences in thresholds or PSEs. In addition, curve-fitting procedures might not converge on a fit if there are insufficient data. Although there is no hard-and-fast rule as to the minimum number of trials necessary, 400 trials is a reasonable number to aim for when one wants to estimate both the threshold and the slope of the PF.

4.2.2.2 Range of Stimulus Levels

As a general rule of thumb, for a performance-based task one wants to choose a set of stimulus levels that will result in performance that ranges from just above chance to just under 100% correct. If more than one stimulus level produces approximately chance performance this means that the lower end of the stimulus range needs to be shifted to a higher level. Similarly, if more than one stimulus level produces approximately 100% correct performance, the highest stimulus level needs to be shifted to a lower level. There is no need to use many, finely spaced stimulus levels. Concentrating responses at just a few appropriately distributed stimulus levels should suffice to obtain reliable estimates of the parameters of a PF.

We will have much more to say about the number of trials needed, as well as the range of stimulus values to use, in Chapter 5. Chapter 5 will discuss adaptive testing methods that, in essence, aim to increase the efficiency of the experimental procedure. That is, they aim to gather the greatest amount of information as possible about the PFs' parameters of interest while using as few trials as possible. They do this by presenting the stimuli at levels that are expected to provide the most information possible about the parameters of interest.

4.2.2.3 Linear versus Logarithmic Spacing of Stimulus Levels

An issue that always seems to come up is how to space the values of the independent variable. The two choices are usually linear or logarithmic (log) spacing. Which one should one choose? One of the reasons given for log spacing is that it allows for a greater range of values. However, this makes no sense—you can have as big a range with linear spacing as with log spacing. Whether using linear or log spacing, the bigger the range, the bigger the interval between values. A more sensible reason for using logarithmic spacing is that you want relatively small intervals at the low and relatively large intervals at the high end of the range. One reason for wanting the interval to increase with the stimulus value is because this gives a closer approximation to how intervals might be represented in the brain. The relationship between the physical and internal representation of a dimension is called the “transducer function.” In the auditory and visual domains, these transducer functions are generally decelerating, such as that shown in [Figure 4.2](#). That is, as stimulus intensity increases, constant increases in stimulus intensity lead to smaller and smaller increases in internal intensity. Of course, the precise form of the bow-shape varies between dimensions, and if one knew the shape exactly, one could space the corresponding x -axis value accordingly. But, given that most dimensions have a bow-shaped transducer function, log spacing of values is a “good bet.”

To derive a set of logarithmically-spaced values you need to decide on the first and last values of the series (call these a and b) and how many values you want (call this n). The i th value of the sequence ($i = 1, 2, \dots, n$) is given by the equation

$$x_i = 10^{[\log a + (i-1)\log(b/a)/(n-1)]} \quad (4.1)$$

Some high-level programming languages will have functions that implement [Eqn \(4.1\)](#). In MATLAB®, for example, the function `logspace` implements [Eqn \(4.1\)](#):

```
>>StimLevels = logspace(log10(a),log10(b),n)
```

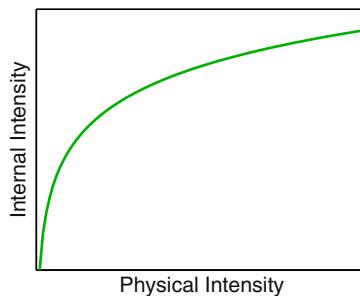


FIGURE 4.2 Typical transducer function between the physical intensity of a stimulus and the corresponding subjective or internal intensity.

For example, suppose you want five values of contrast, ranging from 0.025 to 0.8. Thus $a = 0.025$, $b = 0.8$, and $n = 5$. Substituting these values in the above command will output:

```
StimLevels = 0.0250 0.0595 0.1414 0.3364 0.8000
```

Note that the values are not actual log values, just values that are logarithmically spaced. The sequence is sometimes known as a geometric series, because the ratio of any pair of adjacent values is the same—you can check this yourself (note that the ratios will not be exactly the same because the numbers are only given with a precision of four decimal places). A lot of confusion exists as to whether to use log-transformed values or raw values when fitting a PF. It all depends on what scale the specific function that you fit “expects” your stimulus values to be reported in. For some commonly used forms of PFs two versions exist; one expects stimulus values to be reported on a linear scale and the other expects stimulus values to be reported on a logarithmic scale. For example, the log-Quick function we fitted above has a linear variant known as the Quick function. These two functions will result in equivalent fits to data as long as the stimulus values are supplied on the scale that the function expects. The Quick function expects stimulus values on a linear scale and will report the threshold value on a linear scale, while the log-Quick function expects stimulus values that are log-transformed and reports the threshold on a log-transformed metric. One can take the antilog of the threshold value returned by the log-Quick function to obtain the threshold value on a linear scale. In the literature, both versions are often referred to simply as the Quick function. While this is somewhat understandable since they are just two forms of the same function, it does result in a lot of confusion. We will consistently refer to the form that expects log-transformed stimulus values as the log-Quick function and to the linear variant as the Quick function. See [Box 4.5](#) in [Section B](#) of this chapter for more information on this issue.

4.2.3 Types and Choice of Function

4.2.3.1 Types of Function

In this section we introduce five functions that can be used to model psychometric data: Cumulative Normal; Logistic; Weibull (and its log-version Gumbel or “log-Weibull”); Quick (and its log-version log-Quick); and Hyperbolic Secant. Formal details of the equations of these functions, including their strengths and limitations, will be provided in [Section B](#). It might be beneficial to familiarize yourself with software that evaluates PFs as you continue in this section. [Box 4.1](#) explains how to use the Palamedes toolbox ([Prins and Kingdom, 2009](#)) to evaluate PFs. To illustrate the shapes of the different shapes of functions, [Figure 4.3\(a\)](#) shows an example of a set of data fitted with these functions. In [Figure 4.3\(a\)](#) stimulus contrast is spaced logarithmically, while in [Figure 4.3\(b\)](#) stimulus contrast is spaced linearly. Note that the curves in the two figures are identical but are merely plotted differently. The functions all have the familiar sigmoidal shape when stimulus contrast is spaced logarithmically. Note that the Weibull, Gumbel, Quick, and log-Quick all correspond to the same curve in [Figure 4.3](#), since these four functions are merely different parameterizations of the same curve. A lot of confusion exists among researchers regarding the Weibull family of PFs and we discuss the relations among them in some depth in [Box 4.5](#) in [Section B](#) of this chapter. As can be seen, the Logistic, Cumulative Normal, and Hyperbolic Secant are virtually

BOX 4.1**EVALUATING PSYCHOMETRIC FUNCTIONS IN PALAMEDES**

The function that allows one to find values of PFs in the Palamedes toolbox is of the general form:

```
y = PAL_[NameOfFunction] (paramValues, x);
```

where [NameOfFunction] can be CumulativeNormal, Logistic, Weibull, Gumbel, Quick, logQuick, or HyperbolicSecant. paramValues is a vector that contains values of the parameters of the PF ($\alpha, \beta, \gamma, \lambda$), and x is a scalar, vector, or array of any size containing the values at which the function should be evaluated. Try, for example, generating six values for the Logistic, first setting the stimulus levels to range from 1 through 6, as follows:

```
>>StimLevels = [1:1:6];  
  
>>pcorrect = PAL_Logistic([3 1 0.5 0],StimLevels)
```

The output is:

```
pcorrect = 0.5596 0.6345 0.7500 0.8655 0.9404 0.9763
```

The command:

```
>>plot(StimLevels, pcorrect, 'ko');
```

will generate a crude plot of the PF.

The vector paramsValues does not need to contain values for all four parameters of the PF but does need to contain at least the values for the threshold and the slope. If paramsValues contains only two entries, they are interpreted as values for the threshold and slope, respectively, and the guess rate and lapse rate are assumed to be 0. If a third value is provided it is interpreted as the guess rate, and a fourth will be interpreted as the lapse rate. As an example, in the function call

```
>>pcorrect = PAL_Logistic([3 1 0.5],StimLevels)
```

the vector passed to the function contains three values only, and as a result the lapse parameter is assumed to equal 0. The function thus returns the same results as above. Try generating pcorrect values using some of the other types of PF, and also investigate the effect of changing the four parameters: α, β, γ , and λ .

The above usage of the PF routines in Palamedes gives the probability of a “positive” response (e.g., “correct” or “yes”) given a stimulus intensity and particular values for the four parameters of the PF. The routines implementing the PFs in Palamedes can also perform the inverse operation: given the values for the four parameters and a specific probability of a positive response, the functions can return the stimulus intensity at which the supplied probability of a positive response occurs. In order to use the inverse PF, give the probability of a positive response as the second argument and add the string ‘inverse’ as the third

BOX 4.1 (*cont'd*)

argument to the function. For example, the following call returns the stimulus intensities at which the probabilities of a positive response equal 0.7, 0.8, and 0.9 for a Logistic function with $\alpha = 0$, $\beta = 1$, $\gamma = 0.5$, and $\lambda = 0.01$.

```
>>StimLevels = PAL_Logistic([0 1 0.5 0.01],[0.7 0.8 0.9],'inverse')
```

Finally, these functions can be set up to return the value of the first derivative of the function at any stimulus intensity. In order to do this, pass the string 'derivative' as the third argument to the function. For example, the following call returns the values of the first derivative at stimulus intensities $-1, 0, 1$, and 2 for a Logistic function with $\alpha = 0$, $\beta = 1$, $\gamma = 0.5$, and $\lambda = 0.01$.

```
>>FirstDerivs = PAL_Logistic([0 1 0.5 0.01],[-1 0 1 2],'derivative')
```

identical. The Weibull curve is somewhat distinct from the other curves in the figure. The estimate of the thresholds at the 0.75 correct level would be near identical for all fitted curves in Figure 4.3.

In the Introduction (Section 4.1) we introduced two parameters that were estimated by the fitting procedure: α and β . These parameters describe properties of the underlying sensory mechanism. Two other parameters, however, are needed to specify the PF fully. These are γ (gamma) and λ (lambda). These parameters do not correspond to properties of the underlying sensory mechanism but rather describe chance-level performance and lapsing, respectively. We will discuss the two parameters in turn.

The parameter γ is known as the guessing rate. In Section B we argue that this is a bit of a misnomer, since it is believed that an observer never truly guesses. Nevertheless, if an

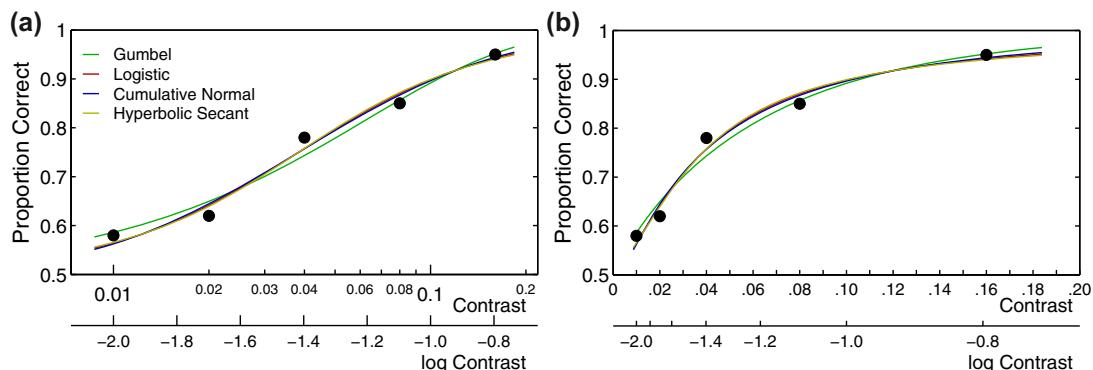


FIGURE 4.3 Example fits of four different PF functions. In (a) stimulus contrast is logarithmically spaced and in (b) contrast is linearly spaced.

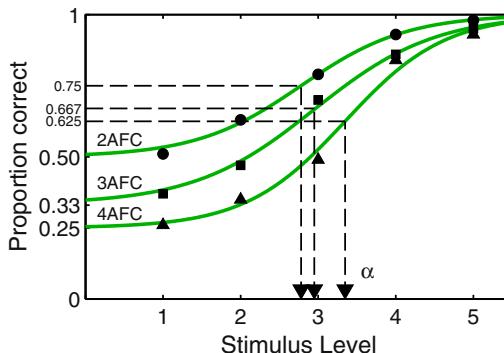


FIGURE 4.4 Example PFs for a 2AFC, 3AFC, and 4AFC task. A Logistic function has been fit to each set of data by using a different value for the guessing parameter, γ (0.5, 0.33, and 0.25, respectively). The lapse rate λ was set to 0 for all three. The threshold α corresponds to proportion correct of 0.75, 0.667, and 0.625, respectively.

observer were to guess on a trial there is a certain probability that the guess would be correct. For example, in a performance-based task γ is simply assumed to equal the reciprocal of the number of alternatives in the forced-choice task or $1/m$ in an M -AFC task (remember that m corresponds to the number of response choices in an M -AFC task. For all tasks described in this chapter $m = M$). Thus, for 2AFC γ is $1/2$ (0.5), for 3AFC it is $1/3$, etc. Figure 4.4 shows examples of the Logistic function fitted to performance-based 2, 3, and 4AFC data, with γ respectively set to 0.5, 0.33, and 0.25. Notice how the range of proportion correct on the y axis is different for the three plots. The proportion correct responses at the threshold parameter α varies with the guessing rate.

The guess rate parameter is also important when fitting PFs for 2AFC data in which responses are coded not in terms of proportion correct but as proportions of one judgment over another, for example the proportion of “brighter” responses in a brighter-versus-darker brightness task or the proportion of “left” responses in a left-versus-right vernier alignment task. In such experiments the resulting proportions range from 0 to 1, and the fitted PFs can be used to obtain either appearance-based measures such as PSEs (as in the brightness matching experiment) or performance-based measures such as accuracy (as in the vernier alignment experiment) and precision (as in both the brightness matching and vernier alignment experiments) (see Chapter 2, Section 2.3.3). Figure 4.5 shows data from a hypothetical appearance-based task. Suppose the aim of the experiment is to find the PSE for the length of two bars of different width. One bar is fixed in length, the other is varied. On each trial the task is to judge which bar appears longer. The length of the variable bar is shown on the abscissa. The ordinate of Figure 4.5 gives the proportion of times the variable bar is perceived as the longer. Thus, a y value of 0.0 means that the variable bar was always perceived as shorter than the fixed bar, while a value of 1.0 means that the variable bar was always perceived as longer than the fixed bar. The PSE is the length of the variable bar that would be perceived just as many times shorter as it is perceived longer than the fixed bar. In other words, it would be perceived as longer on a proportion of 0.5 of the trials. To estimate the PSE parameterized by α , we have fitted a Logistic function and set the guessing rate γ to 0. Threshold α will then correspond to the length of the variable bar, which would be perceived to be longer than the fixed bar on a proportion of 0.5 of trials. Some

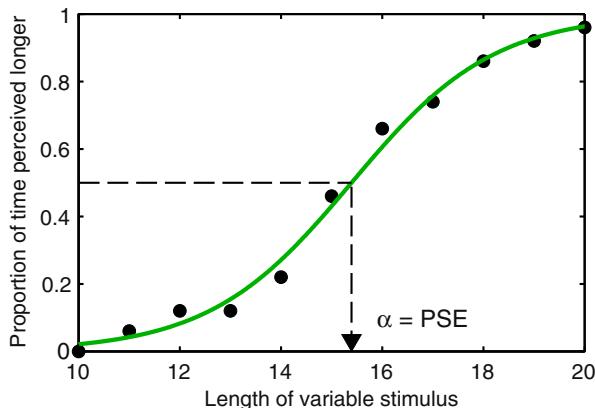


FIGURE 4.5 PF for an appearance-based task.

prefer to plot data in appearance-based tasks on graphs in which the ordinate ranges from -1 to $+1$, presumably so as to have a score of 0 correspond to the PSE, which has some intuitive appeal. This can simply be achieved by rescaling the plot in Figure 4.5. However, for the purposes of “fitting” the function, an ordinate scale of 0 – 1 needs to be used. In the framework of theories of the PF (Section 4.3.1 discusses two of these theories) the y -values correspond to the probabilities of observing one response rather than another and as such are constrained to have a value between 0 and 1 , inclusive.

The fourth parameter associated with a PF, λ , is known as the lapse rate. On a small proportion of trials, observers will respond independently of stimulus level. For example, observers may have missed the presentation of the stimulus, perhaps due to a sneeze or a momentary lapse of attention. On such trials, observers may produce an incorrect response even if the stimulus level was so high that they would normally have produced a correct response. As a result of these lapses, the PF will asymptote to a value that is slightly less than 1 . The upper asymptote of the PF corresponds to $1 - \lambda$. Note that if a lapse is defined as a trial on which the observer misses the presentation of a stimulus and consequently guesses, lapse rate is really not the appropriate term for the parameter λ . Rather, λ corresponds to the probability of responding incorrectly as a result of a lapse (but on some lapse trials, the observer will respond correctly by guessing). We will discuss this issue in more detail in Section B of this chapter. In tasks in which the responses are not coded as proportion correct but rather as proportions of making one judgment over another, as in, for example, a Vernier alignment task, performance may never reach a proportion equal to 0 due to the phenomenon of lapses. For such tasks the guess-rate parameter γ should also be thought of as a lapse rate.

Of the four parameters of the PF, researchers are typically interested only in threshold α and slope β , in that only α and β tell us something about the underlying sensory mechanism. The guessing rate γ is typically determined by the psychometric procedure (2AFC, 3AFC, etc.), and lapse rate λ tells us not about the sensory mechanism, but rather something about such things as the alertness or motivation of the observer. Researchers will usually allow only α and β to vary during the fitting procedure and assume fixed values for γ and λ . Parameters

that are allowed to vary during fitting are referred to as “free parameters;” those that are not allowed to vary are referred to as “fixed parameters.” The guessing rate in an M -AFC task can in most cases safely be assumed to equal $1/m$ (remember, m is the number of response choices and may not always be the same as M). However, it is debatable whether it is reasonable to assume any fixed value for the lapse rate. Researchers often implicitly assume the lapse rate to equal 0. Even the most experienced and vigilant observer, however, will occasionally respond independently of stimulus level. When it is assumed that the lapse rate equals 0, but lapses do in fact occur, this may produce a significant bias on the threshold and slope parameters (e.g., [Swanson and Birch, 1992](#); [Wichmann and Hill, 2001](#)). The bias may be largely avoided if we allow a few lapses to occur by assuming that the lapse rate equals some fixed small value, such as 0.02. An alternative, of course, is to make the lapse rate a free parameter, thereby estimating its value from the data. The issue will be discussed in more detail in [Section B](#) of this chapter (specifically in [Box 4.6](#)). [Box 4.1](#) discusses how to evaluate PFs for different values of the four parameters of the PF using the Palamedes toolbox.

Note that the proportion correct (or “yes” or whatever is used) is a function of all four of the parameters of the PF. However, remember that only parameters α (threshold) and β (slope) characterize the sensory mechanism, while parameters γ (guess rate) and λ (lapse rate) do not. Since it is (almost always) the sensory mechanism that is of interest to researchers, the level of performance should be reported in terms of threshold α and/or β , and not in terms of a threshold defined as the stimulus intensity that corresponds to a certain proportion correct. The latter is a function of all four parameters and characterizes not only the sensory mechanism but also the properties of the task (2AFC, 3AFC, etc.) and nonsensory characteristics of the observer.

4.2.3.2 Choice of Function

What function should one choose? Essentially there are two criteria. The first is that one chooses the function based on an *a priori* theory of the “true” internal shape of the PF. Different theories lead to the use of different functions, although the different functions that are in use are very similar ([Figure 4.3](#)), such that in practice the choice of function is often made based on convenience. In [Section 4.3.2](#) we provide some of the theoretical background that might inform one as to which type of function one might want to choose, based on a *priori* considerations. The second criterion is based on a *posteriori* considerations, specifically using the function that most easily and accurately fits the data. Many practitioners, rightly or wrongly, base their choice on this second criterion.

Once a researcher has decided on which function should be used to model the data, the next step is to find the values of the parameters of that function that describe the data best. To this problem we turn in the next section.

4.2.4 Methods for Fitting Psychometric Functions

There are different methods to find the best-fitting curve to data. The methods differ with respect to the criterion by which “best-fitting” is defined. Here, we will discuss the most commonly used method for fitting the PF. It uses a maximum likelihood (ML) criterion, which defines the best-fitting PF to be the PF that would be most likely to replicate the experiment exactly as it was completed by the human observer. For a detailed discussion of the

theory behind this procedure as well as a second, related procedure (Bayesian estimation) the reader is referred to [Section B](#) of this chapter. Here we will discuss, in the most general of terms, the basic idea behind the “maximum likelihood” method.

Unfortunately, in most situations there is no analytical way in which to find the best-fitting PF when the maximum likelihood criterion is used. Instead, fits are accomplished using a search through possible combinations of parameter values. This is not an entirely straightforward process. For example, for some datasets no maximum likelihood solution may exist. Moreover, for datasets for which a unique maximum likelihood solution exists, other fits may exist that may appear to be the maximum likelihood solution but in fact are not (so-called “local maxima”). A good-fitting procedure should accurately determine whether a unique maximum likelihood solution exists and if it does, it should be able to find it and avoid local maxima. Many fitting problems can be avoided by an understanding of how the fitting procedure works. In [Box 4.7](#) we go over the procedure in detail and present an example scenario in which fitting fails. Here we will briefly review some common mistakes researchers make when they fit a PF to some data. Some of these mistakes are made even before data collection starts.

It is important that the data collected are appropriate to answer the questions that the researcher wants to ask. In other words, a researcher should first determine which of the parameters of the PF are of interest and adjust the experimental procedure accordingly. For example, a researcher who wishes to estimate the value of the threshold and slope parameter will need to collect far more data compared to the researcher who is interested in an estimate of the threshold only. In our experience, by far the most common cause of failed or (seemingly) nonsensical fits is a mismatch between the data and the questions that the researcher asks of these data. We are approached on a fairly regular basis by researchers asking why a particular fit they attempted failed. Almost invariably in these cases, the data simply do not contain enough information needed to support estimation of the parameters that the researcher wishes to estimate. The solution is either to increase the amount of data collected or to decrease the number of parameters that the researcher wishes to estimate.

Other than the sheer number of trials, the choice of stimulus intensities should also match the questions the researcher intends to ask of the data. For example, a common error is for researchers to make the lapse rate a free parameter in the fitting procedure when the data contain virtually no information as to what the value of the lapse rate might be. In such cases, the resulting estimate for the lapse rate may be determined almost exclusively by sampling error ([Prins, 2012](#)). In such situations a comparison between thresholds and/or slopes obtained in different experimental conditions may be negatively affected by making the lapse rate a free parameter. [Box 4.2](#) explains how to fit a PF to data using the Palamedes toolbox.

4.2.5 Estimating the Errors

Because the estimates of parameters α and β are based on a limited number of trials, they are indeed only estimates of the “true” values of α and β , the exact values of which we will never know. So even if we repeated the experiment under identical conditions, the estimated values of α and β would not come out exactly the same, due to the fact that we have a noisy brain. It would be useful, therefore, to obtain some sort of estimate of how much we might expect our estimates of α and β to vary from their true values. We could of course get a good idea of this by repeating our experiment, say 1000 times, obtaining estimates of α and β for

BOX 4.2

FITTING PSYCHOMETRIC FUNCTIONS USING A MAXIMUM LIKELIHOOD CRITERION IN PALAMEDES

In this box, we will demonstrate usage of the Palamedes routine `PAL_PFML_Fit`, which can be used to fit a PF to some data using a maximum likelihood criterion. The example we will discuss will fit a Logistic function to a performance-based 2AFC task. First, we have to set up a series of vectors that contain the data. There are three that are required. As in [Box 4.1](#), we create a vector, `StimLevels`, which contains the values for the stimulus intensities that were used. The second vector we will label `NumPos`, which contains the number of trials in which the observer gave a correct response. The third vector we will label `OutOfNum`, which contains the number of trials that were used for each stimulus level in `StimLevels`. You can use any other name for any or all of these vectors if you like.

```
>>StimLevels = [.01 .03 .05 .07 .09 .11];
>>NumPos = [59 53 68 83 92 99];
>>OutOfNum = [100 100 100 100 100 100];
```

Note that it is not necessary to group the trials that used identical stimulus intensities as we did above. It might be more convenient to supply vectors in which each entry corresponds to a single trial (in this example, the three vectors would each be of length 600). Next we have to specify the type of function we wish to use. The following command assigns the Logistic function to the variable `PF` as a MATLAB inline function. Other functions can be substituted for `Logistic`. Type '`help PAL_PFML_FIT`' to see a listing of available functions. You can also write your own PF and use it; just make sure the argument structure is identical to any of the functions already available in Palamedes.

```
>>PF = @PAL_Logistic;
```

`paramsFree` specifies which of the four parameters— α , β , γ , and λ —are free parameters, i.e., parameters that the algorithm will attempt to find the best-fitting values for. We put 1 for a free parameter and 0 for a fixed parameter. Here we will make the threshold and slope parameters free parameters and use fixed values for the guess and lapse rates. Hence we have:

```
>>paramsFree = [1 1 0 0];
```

`PAL_PFML_Fit` will perform an iterative search for the values for the free parameters. In order to find good starting values for the iterative search, `PAL_PFML_Fit` will first perform a brute force search through a parameter grid that we specify (see [Box 4.7](#) for details on the fitting process). For each of the four parameters of the PF we specify a range of values to be included in the parameter grid.

```
>>searchGrid.alpha = [0.01:0.001:0.11];
>>searchGrid.beta = logspace(0,3,101);
>>searchGrid.gamma = 0.5;
>>searchGrid.lambda = 0.02;
```

BOX 4.2 (*cont'd*)

The above specifies a grid containing 10,201 combinations of parameter values (101 values for $\alpha \times 101$ values for $\beta \times 1$ value for $\gamma \times 1$ value for λ), each of which specifies a possible PF. You can use any name instead of `searchGrid`, but the field names (e.g., `alpha`) must be exactly as given above. `PAL_PFML_Fit` will first find the best-fitting PF among these 10,201 PFs, then use that PF as the starting point for the iterative search. Note that we chose to space the values in `searchGrid.beta` logarithmically between 1 (10^0) and 1000 (10^3). This is not necessary, but a logarithmic spacing of slope values generally makes more sense than a linear spacing (see [Section 4.3.3.1.3](#)). When choosing which parameter values to include in the search grid one should consider the following. First, it is better to make the ranges too wide than it is to make them too narrow. One should be confident that the “true” parameter values are contained within the ranges specified. For the threshold values we simply chose to use the lowest stimulus intensity used as our lower limit and the highest stimulus intensity used as our upper limit. What would make a good range for slope values is less obvious. Slope values differ significantly between the different forms of PF (e.g., a Logistic function with $\beta = 10$ is a lot less steep than a Gumbel with $\beta = 10$), and the slope value will depend on the unit of measurement of your stimulus intensity. If you are confident regarding which order of magnitude your beta value is in, then you might throw a wide net around your best guess (this is essentially what we did above). If, on the other hand, you are not confident at all as to what value the slope might have, it is a good idea to plot a few PFs with different slope values and visually compare them with your data before you decide on the range to use for beta values. The second issue to consider when choosing parameter values to include in the search grid is computational expense. The search grid will contain all possible combinations of parameter values. The phenomenon of combinatorial explosion might make the full grid quite large in terms of its memory usage, especially when we add the guess and/or lapse rate to the list of free parameters.

Now we can run the curve-fitting procedure as follows:

```
>>[paramsValues LL exitflag] = PAL_PFML_Fit(StimLevels, ...
NumPos, OutOfNum,searchGrid, paramsFree,PF)
```

The output is

```
paramsValues =
0.0584 71.0135 0.5000 0.0200

LL =
-281.0842

exitflag =
1
```

`paramsValues` gives the parameter estimates that define the best-fitting PF. Note that the guess rate and lapse rate are given as the values that we specified. The meaning of `LL` will be

Continued

BOX 4.2 (*cont'd*)

given in [Section B](#) of this chapter. The value of one for `exitflag` means that the fit was successful.

Although it is the values of $\hat{\alpha}$ and $\hat{\beta}$ that are important, it's nice to see what the fitted function looks like. The following creates a graph showing the data and the smooth fitted function.

```
>>PropCorrectData = NumPos./OutOfNum;
>>StimLevelsFine = [min(StimLevels):(max(StimLevels)- ...
min(StimLevels))./1000:max(StimLevels)];
>>Fit = PF(paramsValues, StimLevelsFine);
>>plot(StimLevelsFine,Fit,'g-','linewidth',2);
>>hold on;
>>plot(StimLevels, PropCorrectData,'k.','markersize',40);
>>set(gca, 'fontsize',12);
>>axis([0 .12 .4 1]);
```

The graph should look like [Figure B4.2.1](#). Note that the graph plots proportion correct, not number correct, against stimulus level. Note from the figure that the estimate of α ($\hat{\alpha} = 0.0562$) corresponds to the stimulus level at which the fitted function is at 0.74 (i.e., $\gamma + 0.5(1 - \gamma - \lambda)$) proportion correct.

`PAL_PFML_Fit` has a number of optional arguments. Here we merely provide a listing and a brief description of each. To find out in detail how to use any of the options type '`help PAL_PFML_Fit`' or take a look at some of the Demo programs in the `PalamedesDemos` folder.

`searchOptions`: The user may specify the characteristics of the iterative search process by which the function finds the best-fitting estimates of the free parameters, for example,

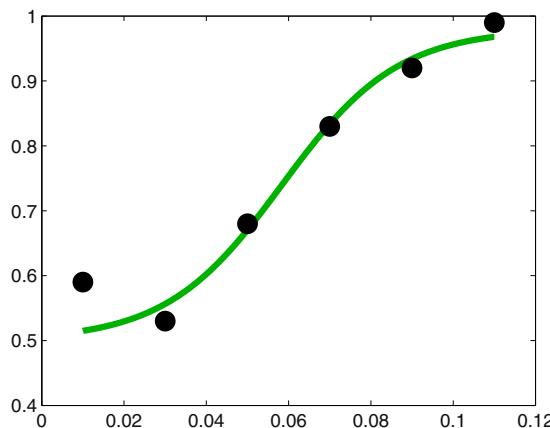


FIGURE B4.2.1 Plot generated by the code in this box.

BOX 4.2 (*cont'd*)

desired precision with which the parameter values are estimated. [Box 4.7](#) explains in some detail how the iterative search proceeds and how the search characteristics affect the search.

`lapseLimits` and `guessLimits`: The user may constrain the values of the guess and/or lapse rates to have a value within a specific range such as to avoid, for example, values outside of the range [0 1].

`lapseFits`: In case the experiment contained a stimulus intensity that was so high that it may be safely assumed that any error made with these intensities was the result of a lapse, a different fitting scheme can be specified (e.g., [Swanson and Birch, 1992](#); [Prins, 2012](#)) that will improve the fit. See [Box 4.6](#) for more detail.

`gammaEQlambda`: In tasks in which responses are coded as proportions of making one judgment over another (e.g., a vernier alignment task) and observed proportions of positive trials vary from around 0 to 1, lapses will prevent both the upper asymptote of the function from reaching 1 and the lower asymptote from reaching 0. As such, a single lapse rate should be estimated to describe both effects. The `gammaEQlambda` option allows for this.

each experiment, and then calculating the variance or standard deviation of the values across all experiments. Unfortunately we do not have the time to do this, but we can get a rough estimate of the likely variability in these parameters from just one set of data.

The preferred method for doing this is called “bootstrap analysis,” and the details of the method are given in [Section B](#) of this chapter. The basic idea behind bootstrap analysis is that instead of having our human observer test the experiment many times over, we have a computer do this. The computer randomly generates many sets of hypothetical data based on the actual experimental data obtained. Each new hypothetical data set is then fitted with the chosen function and estimates of α and β are obtained. The standard deviations of the α and β estimates across all the sets is then calculated, and these are the estimates of the “standard errors” of the parameters. [Box 4.3](#) demonstrates how to derive a standard error using a bootstrap in the Palamedes toolbox.

4.2.6 Estimating the Goodness-of-Fit

The goodness-of-fit is a measure of how well the fitted PF accounts for the data. Using a very limited rule of thumb, if the data fall precisely along the fitted PF then this would be indicative of a good fit, whereas if the data points fall some way away from the fitted PF, this would indicate a bad fit. Generally speaking, a goodness-of-fit test can tell you whether there is enough evidence to believe that the assumptions that your fitted PF makes are incorrect. For example, if your fit assumes that the lapse rate of your observer equals zero, but it is in fact much higher than zero, this may result in a poor goodness-of-fit. A bad fit may also occur when you assume that the shape of the PF is a log-Quick function, but the actual function has a shape unlike the log-Quick.

BOX 4.3

ESTIMATING THE STANDARD ERRORS OF THE PARAMETER ESTIMATES OF A PSYCHOMETRIC FUNCTION IN PALAMEDES USING BOOTSTRAP

The function in the Palamedes toolbox that implements bootstrapping is `PAL_PFML_BootstrapParametric`. It requires that the PF fitting routine has already been run and requires many of the same input arguments that the curve-fitting routine described in the previous section required. The routine will simulate running the experiment that you based your parameter estimates on many times. It will need the arguments `StimLevels` and `OutNum`. In the simulations, the program will act as the best-fitting PF to your data. Thus, we provide the parameter estimates we obtained above from `PAL_PFML_Fit`. The routine will fit all simulated data sets and it needs to be specified which parameters are free and which are fixed. We do this using the `paramsFree` argument. We will also specify what form of PF should be used using the `PF` argument. Additionally, we need to specify how many times the routine should simulate the experiment. The argument `B` does this. The larger `B` is, the better the error estimate will be, but also the longer the routine will take to complete. Setting `B` to 400 should give an acceptable degree of accuracy on the error estimate, and it should also lead to an acceptable completion time. Finally, we specify a search grid through which the routine will perform a brute-force search before the iterative parameter search starts. In order to determine the standard errors for the parameters estimates derived in [Box 4.2](#), initialize `StimLevels`, `OutNum`, `paramsFree`, `PF`, and `searchGrid` exactly as you did there. Also, create a vector `paramsValues` that contains the best-fitting parameter estimates (one way to do this is by performing the fit using `PAL_PFML_Fit` and assigning the parameter estimates to `paramsValues`, as you did in [Box 4.2](#)). Finally, we need to initialize `B` and call the function

```
>>B = 400;
>>[SD paramsSim LLSim converged] = ...
PAL_PFML_BootstrapParametric (StimLevels, OutNum, paramsValues, ...
paramsFree, B, PF, 'searchGrid', searchGrid);
```

In this example, the semicolon has been appended to the last line to prevent it from displaying the results of all 400 simulated fits. To inspect the standard deviation of the estimates, type

```
>>SD
```

An example output might be

```
SD =
0.0040 15.4723 0 0
```

The four values are the estimates of the errors of our estimates of α , β , γ , and λ . Of course, the values are only nonzero for the estimates of the free parameters α and β . If you run the routine again and type out `SD`, the error estimates will be slightly different, because they will be based on

BOX 4.3 (*cont'd*)

a new set of simulated datasets. The larger the value of B , the closer will be the error estimate to the “true” error. As its name suggests, the function `PAL_PFML_BootstrapParametric` performs what is known as a parametric bootstrap. An alternative is to perform a nonparametric bootstrap using the function `PAL_PFML_BootstrapNonParametric`. An explanation of the distinction will have to wait for [Section B](#) of this chapter. Usage of `PAL_PFML_BootstrapNonParametric` is very similar to that of `PAL_PFML_BootstrapParametric`; type `'help PAL_PFML_Bootstrap NonParametric'` for more information. Both `PAL_PFML_BootstrapParametric` and `PAL_PFML_BootstrapNonParametric` have the same optional arguments as `PAL_PFML_Fit` (see [Box 4.2](#)).

The goodness-of-fit is determined by comparing two models statistically. For that reason, we will discuss the details of the procedure and the underlying rationale in Chapter 9, which deals with statistical model comparisons. For now we will merely state that the measure of goodness-of-fit is a so-called statistical p -value. It will always have a value between 0 and 1. Generally speaking, the greater the p -value, the better the model describes the data (but there is much more to it; see [Box 9.6](#)). By somewhat arbitrary convention, researchers agree that the fit is unacceptably poor if the p -value is less than 0.05. [Box 4.4](#) demonstrates how to derive the goodness-of-fit p -value using the Palamedes toolbox.

4.3 SECTION B: THEORY AND DETAILS

4.3.1 Psychometric Function Theories

As discussed in [Section A](#), the PF relates performance on a psychophysical task (e.g., probability of a correct response) to some characteristic of the stimulus (e.g., stimulus contrast). Following general consensus, we will denote performance on the task as a function of stimulus intensity x by $\psi(x)$. The shape of the PF is remarkably similar across a wide variety of tasks and is typically well-described by a sigmoidal function. More often than not, however, we are not directly interested in the measured performance in our experiment. Rather, we are interested in the sensitivity of the sensory mechanism underlying this performance. We will use $F(x; \alpha, \beta)$ (or simply $F(x)$) to symbolize the function relating the performance of the underlying sensory mechanism to stimulus intensity x . Under the High-Threshold Theory (HTT; [Section 4.3.1.1](#)), the value of $F(x)$ corresponds to the probability that the underlying sensory mechanism is able to detect or identify a stimulus of intensity x . [Section 4.3.2](#) discusses various models for $F(x)$ and its two parameters. $F(x)$ cannot be measured directly by psychophysical methods and can only be inferred from performance as we measure it, $\psi(x)$.

Thus, it is worth considering, in some detail, how $\psi(x)$ and $F(x)$ might be related. We will consider this issue first in the context of HTT as this is related in a straightforward manner to

BOX 4.4

DETERMINING THE GOODNESS-OF-FIT OF A PSYCHOMETRIC FUNCTION MODEL IN PALAMEDES

The goodness-of-fit routine in the Palamedes toolbox is `PAL_PFML_GoodnessOfFit` and requires the best-fitting parameter estimates found earlier by the PF fitting routine (Box 4.2). Briefly, the routine simulates the experiment many times, each time mimicking the human observer's behavior using the best-fitting PF determined using `PAL_PFML_Fit`. It then compares the "goodness" of the fit to the human observer's data to those of the fits of the ideal, simulated datasets. The routine uses the same arguments as the error estimation routines described in Box 4.3, and these must of course all be defined. Here is an example implementation:

```
>>B = 1000;
>>[Dev pDev DevSim converged] = PAL_PFML_GoodnessOfFit(StimLevels,
    NumPos, OutOfNum, paramsValues, paramsFree, B, PF, 'searchGrid',
    searchGrid);
```

Note the semicolon to prevent a full printout. Here also, `B` determines the number of simulations on which to base `pDev`. Once again, the higher the value assigned to `B`, the better the estimate of `pDev` will be, but the longer the routine will take to complete. After running the routine, type `Dev` and `pDev` to display the deviance and associated *p*-value:

```
Dev =
5.0221
pDev =
0.2810
```

`pDev` will have a slightly different value each time you run the function, because of the stochastic nature of the bootstrap.

Of course, one can put all the various components described in Boxes 4.2, 4.3, and 4.4 together into a single Matlab m-file. The `PalamedesDemos` folder contains an m-file that does just that. The m-file is named `PAL_PFML_Demo` and can be executed simply by typing its name at the command prompt. First, the program will prompt the user to select either a parametric or a nonparametric bootstrap. It then fits a Logistic function to 2AFC data using the maximum likelihood criterion. The routine uses the same inputs `StimLevels`, `NumPos`, `OutofNum`, `paramsValues`, `paramsFree`, and `searchGrid` as in the above examples and outputs the estimates of the six values described above: threshold α , slope β , SEs of both α and β , goodness-of-fit deviance, and associated *p* value. Finally it generates a graph of the data and fitted function. The routine performs a total of 1401 model fits, so it will require a bit of time to complete. The modest laptop computer on which this sentence is being written just completed this routine in a little under a minute.

the most commonly used expression of the relation between $\psi(x)$ and $F(x)$. We will then discuss how “Signal Detection Theory” (SDT) relates internal sensory mechanisms to the PF. Other theories exist, and we refer the interested reader to [Green and Swets \(1966\)](#) for a more complete discussion. We will discuss HTT and SDT again in some detail in relation to summation measures, the topic of Chapter 7.

4.3.1.1 High-Threshold Theory

Let us imagine a simple 2IFC experiment in which the observer is presented on each trial with two intervals, one containing a stimulus, the other containing no stimulus. The stimulus interval is often denoted S (for signal), whereas the blank interval is denoted N (for noise). The observer is to determine, on each trial, which of the two intervals contained the stimulus.

According to HTT, whether or not the sensory mechanism will detect the stimulus on any trial is determined by the amount of sensory evidence accumulated by the visual system as a result of the presentation of the stimulus. One may think of sensory evidence as some aggregate of the activity of a population of neurons selective for the to-be-detected stimulus. Due to external and internal noise, the amount of sensory evidence accumulated will fluctuate randomly from stimulus presentation to stimulus presentation, such that any given stimulus may give rise to varying amounts of sensory evidence. Let us assume that the mean amount of sensory evidence resulting from the presentation of a stimulus is a linear function of stimulus intensity x : $\mu(x) = \pi + \rho x$. Let us further assume that the random fluctuations in sensory evidence are distributed according to a Gaussian (or “normal”) distribution. The situation is depicted in [Figure 4.6](#). This figure shows the probability density with which a stimulus at intensity k generates different amounts of sensory evidence. Also shown is the probability density associated with the interval that does not contain the stimulus (i.e., stimulus intensity $x = 0$). It can be seen in the figure that the probability density function of the stimulus with intensity $x = k$ is centered around a higher average degree of sensory evidence. In general, increasing stimulus intensity will move the probability density function to higher degrees of expected sensory evidence.

According to HTT, the sensory mechanism will detect the stimulus when the amount of sensory evidence exceeds a fixed internal criterion or threshold. As its name implies, HTT assumes that the internal threshold is high. More specifically, the threshold is assumed to be high enough such that the probability that the threshold is exceeded when $x = 0$ (i.e., by noise alone) is effectively 0. This idea is reflected in the figure by the threshold being beyond the grasp of the $x = 0$ stimulus, thus the noise interval will never result in sensory evidence in excess of the threshold. It is this critical assumption of the high-threshold model that sets it apart from so-called low-threshold theories. Another critical assumption of HTT is that the

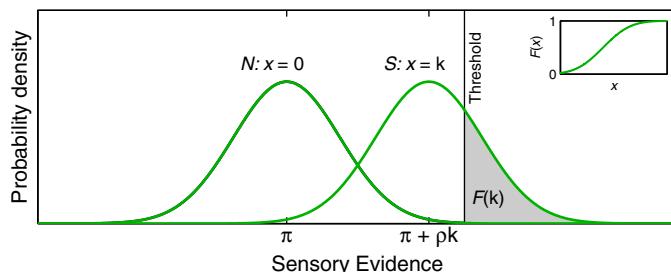


FIGURE 4.6 High-Threshold Theory and the PF.

decision process has no access to the exact amount of sensory evidence accumulated in case the threshold is not exceeded. The decision is based on binary information only: either the sensory evidence was in excess of the threshold, or the sensory evidence was not in excess of the threshold. This second critical assumption sets HTT apart from SDT (Section 4.3.1.2 and Chapter 6). Given the assumptions we have made, function $F(x)$, which under HTT describes the probability that the threshold will be exceeded by a stimulus of intensity x , will be the Cumulative Normal distribution. Function $F(x)$ is shown in the inset in Figure 4.6.

The decision process is straightforward. Since the threshold cannot be exceeded by the noise interval, the sensory mechanism does not generate “false alarms.” Thus, when the threshold is exceeded in one of the two intervals, it must have been because the signal was presented during that interval. In this situation the observer will identify the interval in which the stimulus was presented correctly. On those trials where the signal fails to exceed the threshold, however, the observer is left to guess which interval contained the signal. In this example, the observer will generate a correct response with a probability of 0.5 when the sensory evidence fails to exceed the internal threshold. In general, the probability of producing a correct response based on guessing is $1/m$ in an M -AFC task. As mentioned in Section A, the guess rate is conventionally denoted γ .

We need to make one more consideration before we are ready to consider how $\psi(x)$ and $F(x)$ relate according to HTT. In Section A, we mentioned that on each trial there is a small probability of an incorrect response that is independent of x . This probability is commonly referred to as the lapse rate and is typically symbolized by λ . Lapses may occur because, for example, the observer did not witness the stimulus presentation (sneezes are often blamed). Another reason for a lapse might be a “response” or “finger” error, in which the sensory mechanism may have identified the correct stimulus interval but, for some reason or another, the observer presses the incorrect response button. Anybody who has ever participated in psychophysical experiments will recognize that sometimes our thumbs seem to have a mind of their own.

We will illustrate how $\psi(x)$ and $F(x)$ are related in Figure 4.7. This figure depicts the various series of events that lead to correct and incorrect responses. Starting at the top node, we separate the trials on which a lapse occurs (with probability λ^*) from those where

$$\psi(x; \alpha, \beta, \gamma, \lambda) = \gamma + (1 - \gamma - \lambda^* + \gamma\lambda^*)F(x; \alpha, \beta) = \gamma + (1 - \gamma - \lambda)F(x; \alpha, \beta)$$

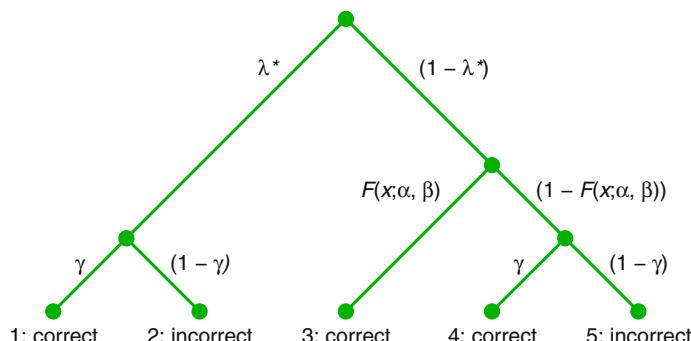


FIGURE 4.7 Relation between $F(x; \alpha, \beta)$ and $\psi(x; \alpha, \beta, \gamma, \lambda)$ according to high-threshold theory.

a lapse does not occur ($1 - \lambda^*$). We use the symbol λ^* to distinguish this probability from the lapse rate, as defined above. Above, we defined the lapse rate as the probability of an incorrect response, which is independent from x (the most common definition in the literature). However, in [Figure 4.7](#) we use λ^* to symbolize the probability that the observer responds independently of stimulus intensity x (for example, resorts to a guess when the stimulus was not witnessed due to a sneeze). In such a situation, the response might still be correct with a probability equal to the guess rate, γ . This sequence actually corresponds to path one in the figure; the observer lapses (λ^*), then guesses correctly (γ). Since these two consecutive events are independent, the probability of this sequence is simply the product of the probabilities of the individual events (i.e., $\lambda^*\gamma$). In path 2, the observer lapses and resorts again to a guess, but this time guesses wrong. The probability of this sequence is $\lambda^*(1 - \gamma)$.

On those trials where the observer does not lapse (with probability $1 - \lambda^*$), the sensory threshold will be exceeded with probability $F(x; \alpha, \beta)$. If this happens, the observer responds correctly and this completes path 3. The probability of this sequence of events equals $(1 - \lambda^*)F(x; \alpha, \beta)$. In path 4, the observer does not lapse ($1 - \lambda^*$), the sensory threshold is not exceeded ($1 - F(x; \alpha, \beta)$), and the observer resorts to a guess and guesses correctly (γ). The probability of this series of events is $(1 - \lambda^*)(1 - F(x; \alpha, \beta))(\gamma)$. Path 5 is identical to path 4, except that the observer guesses incorrectly. The probability with which this sequence occurs equals $(1 - \lambda^*)(1 - F(x; \alpha, \beta))(1 - \gamma)$.

Paths 1, 3, and 4 all result in a correct response. Since the five paths are mutually exclusive (only one can occur on any given trial), the probability of any one of these three paths occurring is the sum of the probabilities of the individual paths. Thus

$$\psi(x; \alpha, \beta, \gamma, \lambda^*) = \lambda^*\gamma + (1 - \lambda^*)F(x; \alpha, \beta) + (1 - \lambda^*)(1 - F(x; \alpha, \beta))(\gamma),$$

which simplifies to

$$\psi(x; \alpha, \beta, \gamma, \lambda^*) = \gamma + (1 - \gamma - \lambda^* + \lambda^*\gamma)F(x; \alpha, \beta) \quad (4.2a)$$

Remember that the symbol λ^* refers to the probability with which the observer responds independently of the value of x , for example because the trial was not witnessed. On such trials a correct response may still occur by lucky guessing. More commonly, the following expression is used:

$$\psi(x; \alpha, \beta, \gamma, \lambda) = \gamma + (1 - \gamma - \lambda)F(x; \alpha, \beta) \quad (4.2b)$$

The symbol λ used in this expression corresponds to the probability that is independent of x with which the observer will generate an incorrect response. The value $(1 - \lambda)$ corresponds to the upper asymptote of $\psi(x)$.

The parameter λ^* is more easily interpreted behaviorally than λ . For example, the value of λ^* for an observer who sneezes on every tenth trial and resorts to a guess on those trials is simply $1/10$ (0.1). The value of λ , on the other hand, will also depend on the guess rate. In a 2AFC task, for example, the observer who sneezes on every 10th trial is expected to guess correctly on one-half (γ) of the sneeze trials, and thus $\lambda = \lambda^*(1 - \gamma) = (0.1)(0.5) = 0.05$. [Figure 4.8](#) displays $\psi_W(x; \alpha, \beta, \gamma, \lambda)$, where the subscript W indicates that function F is the Weibull function ([Section 4.3.2.3](#)), $\alpha = 1$, $\beta = 3$, $\gamma = 0.25$, and $\lambda = 0.05$.

Note that in the above and in [Figure 4.7](#) we have really only considered lapses that are the result of not having witnessed the stimulus presentation. We have ignored lapses that occur

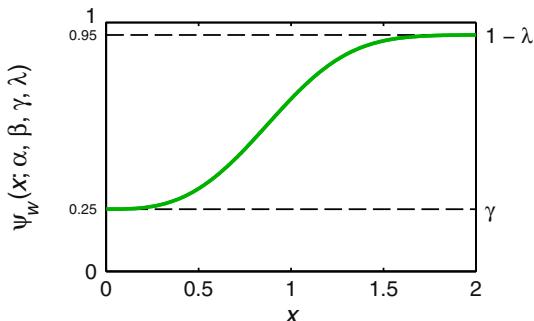


FIGURE 4.8 $\psi_w(x; \alpha, \beta, \gamma, \lambda)$, where F is modeled by the Weibull function $F_W(x; \alpha, \beta)$, threshold $\alpha = 1$, slope $\beta = 3$, guess rate $\gamma = 0.25$, and lapse rate $\lambda = 0.05$.

due to finger errors. We did this for the sake of simplicity. Including finger errors into our discussion above would double the number of paths in Figure 4.7 for a total of 10. For example, one path would be: observer does not lapse (i.e., witnessed stimulus presentation), the sensory threshold is subsequently not exceeded, and the observer then guesses incorrectly but makes a finger error, resulting in a correct response. Note that things get even more complicated in case there are more than two response alternatives. The finger error could then actually result in a correct or an incorrect response. Finger errors are not distinguishable from “sensory” lapses based on the pattern of responses obtained and so there is little reason to complicate our discussion above by treating perceptual and response lapses separately.

4.3.1.2 Signal Detection Theory

The assumption, critical to HTT, that the amount of sensory evidence accumulated is unavailable to the decision process unless it exceeds some internal threshold stands in direct contrast to a central tenet of SDT. According to SDT, there is no such thing as a fixed internal threshold. Instead, SDT makes the assumption that for all stimulus intensities x (including $x = 0$), the sensory mechanism generates a graded signal, corresponding to the degree of sensory evidence accumulated. A decision process follows, which considers the magnitude of this sensory evidence on both S and N intervals. If we again consider the 2IFC task from above, under the SDT framework both the noise and the signal intervals result in a degree of sensory evidence. This degree of evidence is again subject to external and internal noise, such that it will vary randomly from occasion to occasion even when identical stimuli are used. The situation is depicted in Figure 4.9. Under the SDT framework, the decision process has access to the degree of sensory evidence accumulated in both of the intervals. We may think of any presentation of a stimulus as a sample from the probability density function associated with the stimulus. Even in the absence of a stimulus, differing degrees of sensory evidence result, and we may think of the presentation of the noise interval as a sample from the probability density function associated with the noise stimulus. Thus, each of the two intervals on any trial gives rise to sensory evidence, the amount of which is made known to the decision process. The decision rule of the observer is based directly on the relative amplitude of the two samples and is rather simple; the observer will report that the stimulus was presented in the interval from which the greater of the two samples was obtained. It is now easy to see how an incorrect

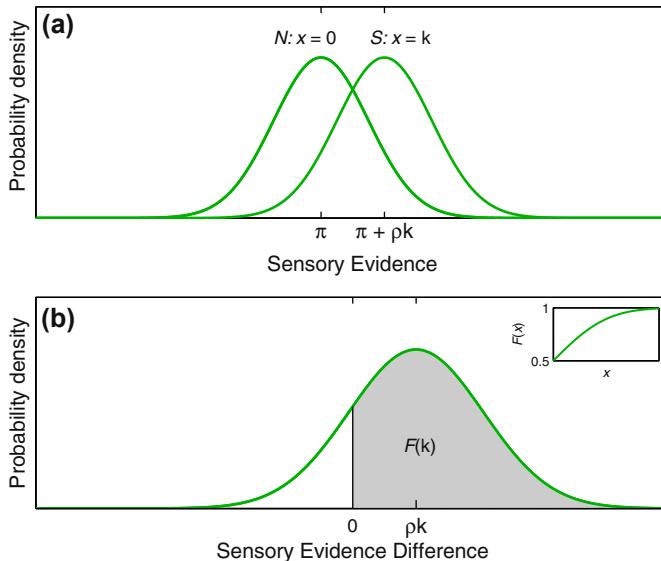


FIGURE 4.9 The relation between SDT and the PF. (a) Probability density functions for the sensory evidence obtained in the noise (N) and stimulus (S) intervals and (b) probability density function for the difference in sensory evidence obtained between the N and S intervals.

response might arise. If we refer again to Figure 4.9, there is considerable overlap between the two functions. As a consequence, it is possible that the sensory activity sampled during the noise interval is greater compared to the activity sampled during the signal interval.

How is the probability of a correct response related to stimulus intensity? In order to generate a specific form of the PF we need to make a few assumptions. Let us again assume that the mean sensory activity (μ) is a linear function of stimulus intensity level x : $\mu(x) = \pi + \rho x$, and that the variance is independent of stimulus level and equal to σ^2 . In other words, the probability density function describing the sensory activity in noise intervals is normal, with mean π and variance σ^2 : $N(\pi, \sigma^2)$ and that in stimulus intervals is $N(\pi + \rho x, \sigma^2)$. This situation is depicted schematically in Figure 4.9(a).

Thus, each trial may be thought of as taking a sample from $N(\pi, \sigma^2)$ in the noise interval and a sample from $N(\pi + \rho x, \sigma^2)$ in the stimulus interval. We assume the observer utilizes a simple (but, given the assumptions, optimal) decision rule; the sample with the greater value was obtained in the stimulus interval. Thus, the response will be correct if the sample taken during the stimulus interval has a value greater than the sample taken during the noise interval. It will be convenient to rephrase this assumption as follows: if the difference between the sample value derived from the signal interval and the sample value derived from the noise interval exceeds 0, the response will be correct. It is well-known that the probability density of the difference between two normally distributed variables is itself normally distributed with mean equal to the difference in means of the individual distributions and variance of the difference equal to the sum of the variances of the individual distributions. Thus, specific to this example, the difference in sensory evidence will be distributed as $N(\rho x, 2\sigma^2)$. Figure 4.9(b) shows the density function for the difference in sensory evidence on a trial in

which the signal is presented at intensity $x = k$. As noted, the stimulus interval will be correctly identified when the sampled difference in sensory activity exceeds 0. The probability with which this will occur corresponds to the shaded area in the figure. When the stimulus intensity equals 0 the difference distribution will be $N(0, 2\sigma^2)$ and the probability that the difference score will exceed 0 is 0.5. This makes sense, of course, because when stimulus intensity equals 0, N and S are identical and the probability that S will produce a greater sensory activity than N is obviously 0.5. Increasing the stimulus intensity will move the difference distribution toward higher values, which corresponds to an increase in the probability that the difference score will exceed 0. Note that, under the assumptions of SDT, the observer never truly guesses. The observer's response is on all trials determined by the relative degree of sensory evidence resulting in each of the stimulus intervals.

Under the assumptions made here, the PF will be the upper half of the Cumulative Normal density function (shown in the figure's inset). This shape of the PF is not encountered often. However, when we change our assumptions, especially with regard to the transducer function (which we above somewhat naïvely assumed to be linear), this would change the shape of the PF. The PF plotted in the figure's inset would take on the more commonly observed sigmoidal shape when we plot stimulus intensity x on a log scale, which is how stimulus intensities typically are plotted.

García-Pérez and Alcalá-Quintana (2007) have shown that Eqn (4.2b), derived above under the assumptions of HTT, is consistent also with the SDT theoretical framework. However, the terms have a different interpretation under SDT than they do under HTT. We mentioned that under SDT, an observer never truly guesses and so the guess parameter γ needs a different interpretation under the assumptions of SDT. Under SDT, a correct response occurs when the sensory activation was greater for the "signal" stimulus than it was for the noise stimulus (or noise stimuli, perhaps). The probability that this will be the case when the signal and noise stimuli are identical (i.e., at the lower asymptote) of course is again $1/m$ in an M -AFC task. Also, whereas under HTT $F(x; \alpha, \beta)$ corresponds to a probability (specifically, the probability that the decision variable will exceed the threshold value), under SDT it does not.

While the critical assumptions of HTT have largely been discredited (e.g., Nachmias, 1981) in favor of those of SDT—for a detailed discussion of the issue see, for example, Swets (1961)—we still prefer the expression given in Eqn (4.2b) since it separates the term that describes the sensory mechanism ($F(x; \alpha, \beta)$) from the terms that are determined by the task design (i.e., γ) and nonsensory observer characteristics (i.e., λ). Also, HTT lives on in our nomenclature for two of the parameters of a PF. Of course, the name "threshold" for the location parameter of a PF is based on HTT. Note that within the framework of HTT the threshold, defined as the amount of sensory evidence that needs to be exceeded before detection takes place, is closely tied to the threshold as defined by the location parameter of a PF. Under the SDT framework there exists no fixed amount of sensory evidence beyond which the stimulus is detected and below which it is not. Nevertheless, we still refer to the location parameter of a PF as "threshold."

The term "guess rate," which we use for the lower asymptote (parameter γ), also has its basis in HTT. The assumption in HTT is that the stimulus is either detected or not, and if it is not, the observer guesses. The probability of a correct response on a trial in which the observer guesses corresponds to the lower asymptote. Quite naturally, under HTT the lower

asymptote came to be referred to as the “guess rate.” Today, we still refer to the lower asymptote (i.e., γ) as the guess rate. Moreover, in this text, we sometimes take even greater liberties. For example, where we should really say: “The amount of sensory evidence accumulated while sampling from the signal presentation happened to exceed the amount of sensory evidence accumulated while sampling from the noise presentation” we might say instead: “The observer guessed correctly.”

4.3.2 Details of Function Types

Above we derived the generic formulation of the PF (Section 4.3.1.1):

$$\psi(x; \alpha, \beta, \gamma, \lambda) = \gamma + (1 - \gamma - \lambda)F(x; \alpha, \beta) \quad (4.2b)$$

As discussed there, under the high-threshold detection model, $F(x; \alpha, \beta)$ describes the probability of detection of the stimulus by the underlying sensory mechanism as a function of stimulus intensity x , γ corresponds to the guess rate (the probability of a correct response when the stimulus is not detected by the underlying sensory mechanism), and λ corresponds to the lapse rate (the probability of an incorrect response, which is independent of stimulus intensity).

Several functions are in use for $F(x; \alpha, \beta)$. We list here the most commonly used functions. Examples of all of these are shown in Figure 4.3. We will consistently use the symbol α to denote the location parameter (threshold) and the symbol β to denote the rate-of-change or slope parameter, even where this flies in the face of convention. We will also use expressions of F in which increasing values of β correspond to increasing slopes of F , even if this defies convention. Box 4.1 explains how the routines that implement the PFs in the Palamedes toolbox are used.

4.3.2.1 Cumulative Normal Distribution

The Cumulative Normal distribution is perhaps the most justifiable form of $F(x; \alpha, \beta)$ theoretically. If one assumes that the noise that underlies the variability of sensory evidence is a linear combination of many independent and alike noise sources then the total resulting noise would be approximately normally distributed, by the well-known Central Limit Theorem (e.g., Hays, 1994). The Cumulative Normal distribution is given as

$$F_N(x; \alpha, \beta) = \frac{\beta}{\sqrt{2\pi}} \int_{-\infty}^x \exp\left(-\frac{\beta^2(x - \alpha)^2}{2}\right) \quad (4.3)$$

with $x \in (-\infty, +\infty)$, $\alpha \in (-\infty, +\infty)$, and $\beta \in (0, +\infty)$. No analytical solution to the integral is known, but the distribution may in practice be approximated by a numerical method. Parameter α corresponds to the threshold: $F_N(x = \alpha; \alpha, \beta) = 0.5$. Varying α , while keeping β constant, corresponds to a rigid translation of the function. Parameter β corresponds to the reciprocal of the standard deviation of the normal distribution and determines the slope of the PF. Since $F_N(x = 0; \alpha, \beta) > 0$ and $\lim_{x \rightarrow -\infty} F_N(x; \alpha, \beta) = 0$ for all values in the domains of α and β , the Cumulative Normal would be inappropriate in a task in which $x = 0$ corresponds to an absence of signal, unless x is log-transformed.

4.3.2.2 Logistic

The Logistic function is given as

$$F_L(x; \alpha, \beta) = \frac{1}{1 + \exp(-\beta(x - \alpha))} \quad (4.4)$$

with $x \in (-\infty, +\infty)$, $\alpha \in (-\infty, +\infty)$, and $\beta \in (0, +\infty)$. Parameter α corresponds to the threshold: $F_L(x = \alpha; \alpha, \beta) = 0.5$; parameter β determines the slope of the PF. The Logistic function is a close approximation to the Cumulative Normal distribution (after a linear transformation of the slope parameter β : $\beta_L \approx 1.7/\beta_N$). An advantage of the Logistic function over the Cumulative Normal is that the former has a known closed-form integral while the latter does not. For the same reasons as outlined for the Cumulative Normal distribution, the Logistic is inappropriate when a stimulus intensity $x = 0$ corresponds to an absence of signal, unless x is log-transformed.

4.3.2.3 Weibull

The Weibull function is given as

$$F_W(x; \alpha, \beta) = 1 - \exp\left(-\left(\frac{x}{\alpha}\right)^\beta\right) \quad (4.5)$$

with $x \in [0, +\infty)$, $\alpha \in (0, +\infty)$, and $\beta \in (0, +\infty)$. Threshold α corresponds to $F_W(x = \alpha; \alpha, \beta) = 1 - \exp(-1) \approx 0.6321$; parameter β determines the slope in conjunction with α . That is, changing the value of α will alter the slope (and the shape) of the function, even when β is held constant. However, when plotted against $\log x$, a change in α will result in a rigid translation of the function when β is held constant. $F_W(0; \alpha, \beta) = 0$ for all α, β in their respective domains. Note that since the domain of x includes nonnegative numbers only, the Weibull should not be used when x is measured in logarithmic units. The Gumbel (or “log-Weibull”) function should be used instead.

[Quick \(1974\)](#) has argued that the Weibull provides an excellent approximation to the PF when performance is determined by probability summation among channels with normally distributed noise. Much confusion exists in the literature regarding functions in the Weibull family (i.e., the Weibull, Gumbel, Quick, and log-Quick functions). See [Box 4.5](#) for information on how these functions relate to each other.

4.3.2.4 Gumbel (Also Known as Log-Weibull)

$$F_G(x; \alpha, \beta) = 1 - \exp(-10^{(\beta(x - \alpha))}) \quad (4.6)$$

with $x \in (-\infty, +\infty)$, $\alpha \in (-\infty, +\infty)$, and $\beta \in (0, +\infty)$. Threshold α corresponds to

$F_G(x = \alpha; \alpha, \beta) = 1 - \exp(-1) \approx 0.6321$. The Gumbel function is the analog of the Weibull function when a log-transform on x is used. For that reason, in the literature the Gumbel function is often referred to as the log-Weibull function or, somewhat confusingly, simply as the Weibull function. See [Box 4.5](#) for more information on the Gumbel function and how it is related to other functions in the Weibull family.

BOX 4.5

THE WEIBULL FAMILY OF PSYCHOMETRIC FUNCTIONS

A lot of confusion exists regarding the Weibull and related functions, specifically the Gumbel (or log-Weibull), the Quick, and the log-Quick functions. In this box we explain the differences and similarities between the functions. Let's say you perform a 2AFC contrast detection task. You use five stimulus contrasts: 0.01, 0.02, 0.04, 0.08, and 0.16. You present 100 trials at each of the five stimulus contrasts, and the number of trials in which the stimulus was detected are 58, 62, 78, 85, and 95, respectively, for the five stimulus contrasts.

If used correctly, the four functions in the Weibull family (Weibull, Gumbel, Quick, and log-Quick) will all fit the exact same PF to these data. The data and fits are shown in [Figure B4.5.1](#). The four functions differ in two respects: (1) whether stimulus intensity values are measured in the original, linear metric or whether they have been logarithmically transformed and (2) the value to which the functions evaluate when $x = \alpha$. We will discuss these differences in turn. [Table B4.5.1](#) summarizes this.

The Weibull and the Quick functions expect the stimulus intensities— x in [Eqn \(4.2\)](#)—to be defined on a linear scale (i.e., the values of x to use are exactly like the values given above). An important thing to consider here is that both the Weibull and the Quick evaluate to 0 when the stimulus intensity equals 0 (thus the PF or values of ψ will evaluate to γ , the guess rate, at $x = 0$). In other words, a stimulus intensity of 0 should correspond to a complete absence of signal, which is the case here: a value of 0 for contrast means there is no contrast and there is nothing to be detected. Moreover, negative values for stimulus intensities are meaningless. The Gumbel and the log-Quick, on the other hand, expect stimulus intensities that have been log-transformed. In other words, the values of x to use are the log-transformed values of the intensities given above, i.e., approximately -2 , -1.70 , -1.40 , -1.10 , and -0.80 , respectively. The threshold parameter α is expressed in the same units as x . Parameter β does not depend on the units of measurement of the stimulus intensity.

Whereas the Weibull and the Gumbel evaluate to $1 - e^{-1}$ (approximately 0.6321) at threshold (i.e., $F(x = \alpha; \alpha, \beta) = 1 - e^{-1} \approx 0.6321$), the Quick and the log-Quick evaluate to $1 - 2^{-1}(0.5)$ (i.e., $F(x = \alpha; \alpha, \beta) = 1 - e^{-2} = 0.5$). There are some advantages to having the threshold correspond to the value at which the function evaluates to 0.5 (as the Quick and log-Quick do). Many other PFs evaluate to 0.5 at threshold (e.g., the Cumulative Normal, Hyperbolic Secant, and Logistic) and thus it is easier to compare these functions to the Quick and log-Quick functions than it is to compare them to the Weibull and Gumbel functions.

When we fit the data above using fixed values for the guess and lapse rate of 0.5 and 0.01, respectively, the numerical values of the ML estimates for α are different between all four of the PFs in the Weibull family. They are: $\hat{\alpha}_W = 0.0594$, $\hat{\alpha}_G = -1.2260$, $\hat{\alpha}_Q = 0.0402$, and $\hat{\alpha}_{lQ} = -1.397$ (the subscript merely identifies the function fitted). The ML estimates for β are identical:

$$\hat{\beta}_W = \hat{\beta}_G = \hat{\beta}_Q = \hat{\beta}_{lQ} = 0.9380.$$

Continued

BOX 4.5 (cont'd)

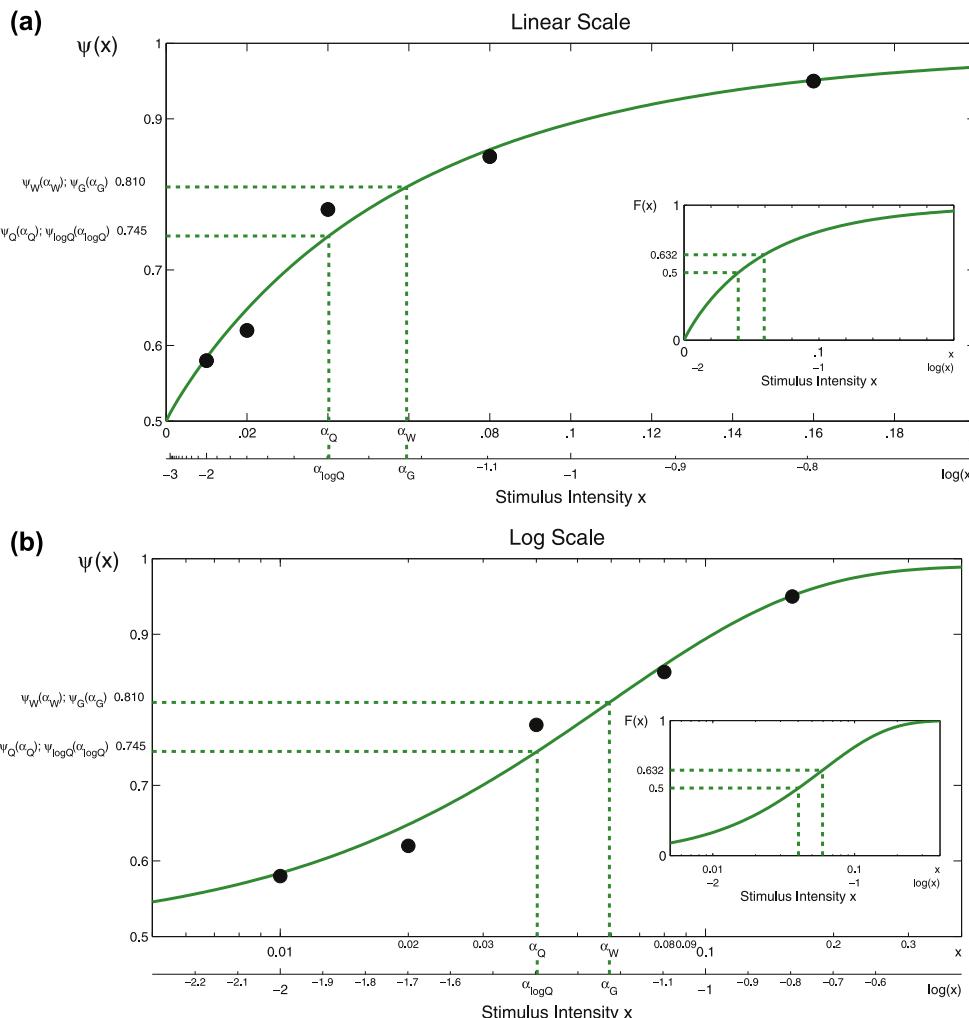


FIGURE B4.5.1 The relationships among the PFs in the Weibull family: Weibull, Gumbel, Quick, and log-Quick functions. Functions are plotted on linear scale (a) as well as logarithmic scale (b).

TABLE B4.5.1 The characteristics by which the functions in the Weibull family of psychometric functions differ

Scale of measurement of x and α			
	Linear	Logarithmic	
$F(x = \alpha; \alpha, \beta)$	$1 - e^{-1} (\approx 0.6321)$	Weibull	Gumbel ("log-Weibull")
	$1 - 2^{-1} (= 0.5)$	Quick	Log-Quick

BOX 4.5 (*cont'd*)

As mentioned, however, these four fits in fact all correspond to one and the same PF, namely the green curve in both panels of [Figure B4.5.1](#). Note that the parameter values differ only in the value of the fitted threshold. The value of the threshold estimate for the Gumbel is simply the log-transform of the threshold estimate for the Weibull (i.e., $\log_{10}(0.0594) = -1.226$). Thus the Weibull and the Gumbel are identical except that the Weibull uses a linear scale for contrast whereas the Gumbel uses a logarithmic scale. Similarly, the Quick and the log-Quick functions are identical to each other except that the Quick operates on a linear scale but the log-Quick operates on a logarithmic scale. The difference between the Weibull and Gumbel on the one hand and the Quick and the log-Quick on the other is that, for the former pair, F (in the equation above) evaluates to $0.6321(1 - e^{-1})$ at threshold, whereas for the latter pair F evaluates to 0.5.

Note that two PFs of the same kind (i.e., both are Weibull or both are Gumbel, etc.) that differ in the value for the threshold only are (rigid) translations of each other when plotted on a logarithmic scale, whereas they will differ in their shape when plotted on a linear scale.

So, which of the four functions should you use? In the end it doesn't matter much because, used correctly, they all lead to the exact same fit. However, when a standard error is determined using either a standard bootstrap procedure ([Section 4.3.3.1.3](#)) or a Bayesian procedure ([Section 4.3.3.2.3](#)), different results will be obtained depending on whether one uses log-transformed stimulus intensities (Gumbel and log-Quick) or linear stimulus intensities (Weibull and Quick). Our advice is to follow a simple rule of thumb: if you have chosen to space your stimulus intensities equally on a logarithmic scale in the experiment, use the Gumbel or log-Quick both in your fitting procedure and your standard error estimation procedure. If your stimulus intensities were spaced equally on a linear scale, use the Weibull or the Quick function.

4.3.2.5 Quick

The Quick function is given as

$$F_Q(x; \alpha, \beta) = 1 - 2^{\left(-\left(\frac{x}{\alpha}\right)^{\beta}\right)} \quad (4.7)$$

with $x \in [0, +\infty)$, $\alpha \in (0, +\infty)$, and $\beta \in (0, +\infty)$. Note that the Weibull and Quick functions are identical save for the base of the exponent. Use of the base 2 in the Quick function makes threshold α correspond to $F_Q(x = \alpha; \alpha, \beta) = 1 - 2^{-1} = 0.5$. Parameter β determines the slope. See [Box 4.5](#) for more information on the Quick function and how it is related to other functions in the Weibull family.

4.3.2.6 Log-Quick

$$F_{LQ}(x; \alpha, \beta) = 1 - 2^{(-10^{(\beta(x-\alpha))})} \quad (4.8)$$

with $x \in (-\infty, +\infty)$, $\alpha \in (-\infty, +\infty)$, and $\beta \in (0, +\infty)$. Threshold α corresponds to $F_{LQ}(x = \alpha; \alpha, \beta) = 1 - 2^{-1} = 0.5$. Parameter β determines the slope. The log-Quick function is the analog of the Quick function when a log-transform on x is used. See Box 4.5 for more information on the log-Quick function and how it is related to other functions in the Weibull family.

4.3.2.7 Hyperbolic Secant

$$F_{HS}(x; \alpha, \beta) = \frac{2}{\pi} \tan^{-1} \exp\left(\frac{\pi}{2} \beta(x - \alpha)\right) \quad (4.9)$$

with $x \in (-\infty, +\infty)$, $\alpha \in (-\infty, +\infty)$, and $\beta \in (0, +\infty)$. Threshold α corresponds to $F_{HS}(x = \alpha; \alpha, \beta) = 0.5$. Parameter β determines the slope. Use of the Hyperbolic Secant is relatively rare in the psychophysical literature. We include it here for completeness.

4.3.2.8 The Spread of Psychometric Functions

We have used β in all of the forms of the PF above to symbolize the parameter, which determines the steepness of the PF. Because β affects the steepness of the PF, it is often referred to as the slope parameter or simply “the slope” of the PF. This is not entirely proper as β does not directly correspond to the slope of the function as it is defined in calculus. Moreover, values of beta cannot be compared directly between the different forms of PF. For example, a Cumulative Normal function with $\beta = 2$ is much steeper compared to a Logistic function with $\beta = 2$. A common measure related to the steepness of PFs is the spread (or support). The spread will actually have an inverse relation to the slope of the PF. Remember that all of the PFs display asymptotic behavior. That is, as stimulus intensity increases, ψ asymptotes toward $1 - \lambda$ but will never actually attain that value. Similarly, as stimulus intensity decreases, ψ asymptotes toward γ (with the exception of the Weibull and Quick functions, which are not defined for $x < 0$ and whose value at $x = 0$ actually equals γ). As such, we cannot define spread as the range of stimulus intensities within which ψ goes all the way from the lower asymptote γ to the upper asymptote $1 - \lambda$. Instead, we pick an arbitrary number δ (e.g., 0.01) and define the spread to be that stimulus range within which ψ goes from $\gamma + \delta$ to $1 - \lambda - \delta$. Formally, if we let σ symbolize spread

$$\sigma = \psi^{-1}(1 - \lambda - \delta; \alpha, \beta, \gamma, \lambda) - \psi^{-1}(\gamma + \delta; \alpha, \beta, \gamma, \lambda), \quad (4.10)$$

where $\psi^{-1}(y; \alpha, \beta, \gamma, \lambda)$ is the inverse of the PF $\psi(x; \alpha, \beta, \gamma, \lambda)$. The value of δ must, of course, have a value between 0 and $(1 - \gamma - \lambda)/2$.

4.3.3 Methods for Fitting Psychometric Functions

The raw data resulting from a psychophysical experiment are the proportions of correct responses measured at a number of different stimulus intensities x . Each of these is based

on a limited number of trials and hence is only an estimate of the true probability with which the observer generates a correct response. We assume that the true probabilities of a correct response as a function of x are given by Eqn (4.2b). Since, in most situations, we are interested in describing the properties of the underlying sensory mechanism, we are interested only in determining the values of the threshold (α) and slope (β) of the function $F(x; \alpha, \beta)$. The guess rate (γ) is usually known ($1/m$ in an M-AFC task). The lapse rate (λ) is unknown but is considered to be a nuisance parameter as it tells us nothing about the sensory mechanism per se. We may, of course, attempt to estimate it, but when we do so it is to improve our estimates of α and β (although situations might be imagined where the lapse rate is of interest for its own sake, in which case α and β might be considered nuisance parameters). We might also be interested in the precise shape of the function $F(x; \alpha, \beta)$. For example, it might be of theoretical interest to determine whether the Weibull function, say, provides a better fit to our data compared to the Logistic function.

We have a number of methods available to us to find the best-fitting PF to our data. These methods differ ultimately with respect to the criterion by which “best-fitting” is defined. We will discuss two different methods in some detail. The first method uses the criterion of maximum likelihood to define best-fitting, and the second uses a Bayesian criterion.

4.3.3.1 Maximum Likelihood Criterion

4.3.3.1.1 A SIMPLE ONE-PARAMETER EXAMPLE

Let us start with a simple example to introduce the concept of “likelihood.” Imagine that we have a coin and we wish to estimate the parameter corresponding to the probability that our coin lands “heads” on any given flip of the coin. We will designate this parameter a . We perform a (rather modest) experiment that consists of flipping the coin 10 times. After each flip, we note whether it landed “heads” (H) or “tails” (T). The results of our 10 trials are respectively:

HHTHTTHHHTH

The likelihood function associated with our parameter of interest is

$$L(a|\mathbf{y}) = \prod_{k=1}^N p(y_k|a) \quad (4.11)$$

(e.g., Hoel et al., 1971), where a is a potential value for our parameter of interest, $p(y_k|a)$ is the probability of observing outcome y on trial k given or (perhaps more appropriately in this context) “assuming” value a for our parameter, and N is our total number of trials (here, $N = 10$). In our example, it is obvious that $p(y_k = H|a) = a$ and $p(y_k = T|a) = 1 - a$. Equation (4.11) utilizes what is known as the multiplicative rule in probability theory (sometimes referred to as the “and rule”), which states that the probability of observing two or more events is equal to the product of the probabilities of the individual events when the events are independent. Thus, the likelihood associated with, say, $a = 0.4$ is

$$\begin{aligned}
 L(a|\mathbf{y}) &= \prod_{k=1}^N p(y_k|0.4) \\
 &= p(y_1 = H|0.4) \cdot p(y_2 = H|0.4) \cdot p(y_3 = T|0.4) \cdot \dots \cdot p(y_{10} = H|0.4) \\
 &= (0.4) \cdot (0.4) \cdot (1 - 0.4) \cdot \dots \cdot (0.4) = (0.4)^6 \cdot (0.6)^4 \\
 &\approx 0.000531
 \end{aligned}$$

In other words, the likelihood $L(0.4 | \mathbf{y})$ is calculated as the probability of observance of the outcome of 10 flips, exactly as they occurred in our experiment, from a coin for which α is known to be 0.4. Importantly, contrary to intuitive appeal perhaps, it would be inappropriate to consider $L(a | \mathbf{y})$ a probability, although we calculate it as such. In the context of our experiment we cannot think of $L(a | \mathbf{y})$ as the probability of obtaining our experimental outcome, simply because our experiment is over and there is no uncertainty (anymore) as to the outcome. Thus, our obtained value for $L(0.4 | \mathbf{y})$ does not give us information about the outcome of our completed experiment. Rather, we calculate it to gain information about the value for α . Our obtained value for $L(0.4 | \mathbf{y})$, however, is also most certainly not the probability of parameter α having the value 0.4. Thus, $L(a | \mathbf{y})$ is not the probability of anything, and using the term “probability” would be inappropriate. Instead, we use the term “likelihood.” The likelihood function is a function of a and we may calculate $L(a | \mathbf{y})$ for any value of a . Figure 4.10 plots $L(a | \mathbf{y})$ as a function of a across the range $0 \leq a \leq 1$ (since a represents a probability, it must have a value within this range). As the term implies, the maximum likelihood estimate of parameter α is that value of a that maximizes the likelihood function $L(a | \mathbf{y})$. In our example, $L(a | \mathbf{y})$ is at maximum when a equals 0.6. Thus, $\hat{\alpha} = 0.6$ is our maximum likelihood estimate of α .

It may be noted that Eqn (4.11) calculates the probability of observance of an exact “ordered” sequence of, for example, heads and tails (more generally, “successes” and “failures”),

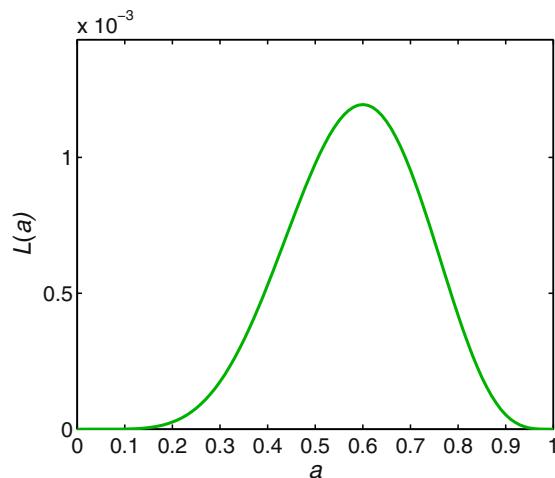


FIGURE 4.10 Plotted is the likelihood as a function of a , the (hypothesized) value for the probability of observance of heads on any given flip of our coin with unknown α .

on a series of N independent trials. Some authors opt to include the binomial coefficient in Eqn (4.11) to arrive at the probability of observing the numbers of successes and failures in any order:

$$L(a|\mathbf{y}) = \frac{N!}{(N-m)!} \prod_{k=1}^N p(y_k|a) \quad (4.12)$$

where m is the number of successes (heads). However, since the value of the binomial coefficient is determined entirely by the observed outcome of the experiment and does not depend on a , inclusion of the binomial coefficient amounts merely to a linear rescaling of the values of $L(a|\mathbf{y})$ and thus will not affect our estimate of α .

One should not be discouraged by the small values of $L(a|\mathbf{y})$ obtained. Even the likelihood for our best estimate of α (0.6) amounts to a mere 0.0012. In other words, a coin for which $\alpha = 0.6$ would, when flipped 10 times, have a probability of only 0.0012 of generating the sequence of outcomes we have observed in our experiment. This seems so unlikely that it might be tempting to conclude that our estimate $\hat{\alpha} = 0.6$ is not a very good one! This conclusion would be inappropriate, however. In effect, we have witnessed the occurrence of an event and have calculated post hoc the probability that this event would occur under certain assumptions (specifically, for a range of values of a). However, as argued above, this probability can be interpreted neither as the probability of our experiment resulting in the observed sequence nor as the probability of α having the value of 0.6.

4.3.3.1.2 THE PSYCHOMETRIC FUNCTION AND THE LIKELIHOOD FUNCTION

Typically, we wish to estimate two parameters of the PF: its threshold (α) and its slope (β). Thus the likelihood function is now a function of two parameters and becomes

$$L(a, b|\mathbf{y}) = \prod_{k=1}^N p(y_k|x_k; a, b), \quad (4.13)$$

where $p(y_k|x_k; a, b)$ is the probability of observance of response y (in an M -AFC task typically “correct” or “incorrect”) on trial k given stimulus intensity x_k and assuming threshold $\alpha = a$ and slope $\beta = b$ of the PF. Let us again imagine a modest experiment: an observer is to decide which of two temporal intervals contains a low-intensity visual stimulus. Five stimulus intensities x are used and the observer is tested four times at each of the five stimulus intensities. Table 4.2 presents the responses observed for each of the 20 trials (1: correct; 0: incorrect). Also listed for each response is the likelihood for two, somewhat arbitrary, assumed PFs. These individual trial likelihoods are, of course, simply equal to $\psi(x_k; a, b, \gamma, \lambda)$ if the response is correct or $1 - \psi(x_k; a, b, \gamma, \lambda)$ if the response is incorrect. One of the functions is characterized by $a = 1, b = 1$, the other by $a = 10, b = 1$. We assume the guess rate γ equals 0.5, and the lapse rate λ equals 0. Since γ and λ are fixed we will denote the probability of a correct response as $\psi(x_k; a, b)$ for purposes of brevity. The two PFs for which we explicitly calculate the likelihoods are shown in Figure 4.11. Following Eqn (4.13), the likelihood based on all 20 responses is simply calculated as the product of the likelihoods based on all individual trials. These overall likelihoods are also listed in the table. The interpretation of the likelihood here is analogous to that in the previous section. For example, the value $L(a = 1, b = 1 | \mathbf{y}) = 4.078 \times 10^{-5}$ can be interpreted as the probability that an observer whose true underlying

TABLE 4.2 The log-transformed stimulus level ($\log(x)$), the observed outcome of the trial (y , 0 = incorrect, 1 = correct), the probability of a correct response for two assumed PFs ($\psi(x_k; a = 1, b = 1)$ and $\psi(x_k; a = 10, b = 1)$), and the likelihood of each of the observed outcomes for both assumed PFs ($p(y_k | x_k; a = 1, b = 1)$ and $p(y_k | x_k; a = 10, b = 1)$) are shown for each of 20 trials (also shown are the likelihoods for the two PFs considered across the entire experiment ($L(1, 1 | y)$ and $L(10, 1 | y)$))

K	$\log(x)$	y	$\psi(x_k; a = 1, b = 1)$	$p(y_k x_k; a = 1, b = 1)$	$\psi(x_k; a = 10, b = 1)$	$p(y_k x_k; a = 10, b = 1)$
1	-2	1	0.5596	0.5596	0.5237	0.5237
2	-2	0	0.5596	0.4404	0.5237	0.4763
3	-2	1	0.5596	0.5596	0.5237	0.5237
4	-2	0	0.5596	0.4404	0.5237	0.4763
5	-1	0	0.6345	0.3655	0.5596	0.4404
6	-1	1	0.6345	0.6345	0.5596	0.5596
7	-1	1	0.6345	0.6345	0.5596	0.5596
8	-1	1	0.6345	0.6345	0.5596	0.5596
9	0	1	0.7500	0.7500	0.6345	0.6345
10	0	1	0.7500	0.7500	0.6345	0.6345
11	0	0	0.7500	0.2500	0.6345	0.3655
12	0	1	0.7500	0.7500	0.6345	0.6345
13	1	1	0.8655	0.8655	0.7500	0.7500
14	1	1	0.8655	0.8655	0.7500	0.7500
15	1	1	0.8655	0.8655	0.7500	0.7500
16	1	0	0.8655	0.1345	0.7500	0.2500
17	2	1	0.9404	0.9404	0.8655	0.8655
18	2	1	0.9404	0.9404	0.8655	0.8655
19	2	1	0.9404	0.9404	0.8655	0.8655
20	2	1	0.9404	0.9404	0.8655	0.8655
				$L(1, 1 y) = 4.078 \times 10^{-5}$		$L(10, 1 y) = 2.654 \times 10^{-5}$

PF is characterized by $\alpha = 1$ and $\beta = 1$ would generate the exact sequence of responses as that produced by our observer. Again, we should not be discouraged by the minute value of the likelihood: even if our observed proportion of correct for each of the stimulus levels would be perfectly predicted by the PF, our likelihood would be rather minute.

Of course, we may calculate the likelihood for any particular combination of a and b . Figure 4.12 presents a contour plot of $L(a, b | y)$ as a function of $\log a$ and $\log b$ across the

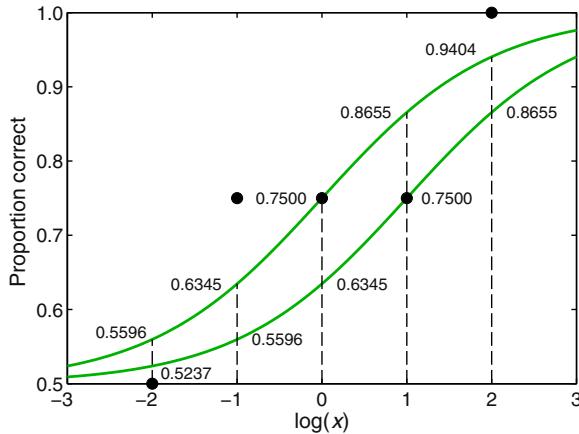


FIGURE 4.11 Shown is the probability correct (ψ) for two PFs (both are Logistic, one characterized by $\alpha = 1$, $\beta = 1$, the other by $\alpha = 10$, $\beta = 1$) as a function of log stimulus intensity ($\log(x)$). Also shown are the results of our experiment as the proportion of correct responses for each of the five stimulus intensities used.

ranges $\log(a) \in [-2, 2]$ and $\log(b) \in [-1, 2]$. The two specific PFs whose likelihoods are calculated in Table 4.2 are indicated in the figure by the square symbols. Analogous to the previous one-parameter coin-flipping experiment described above, the maximum likelihood estimates of the threshold and slope of the PF are those values of a and b that maximize $L(a, b | y)$.

In practice, we perform our calculations using log-transformed probabilities:

$$LL(a, b | \mathbf{y}) = \sum_{k=1}^N \log_e p(y_k | x_k; a, b) \quad (4.14)$$

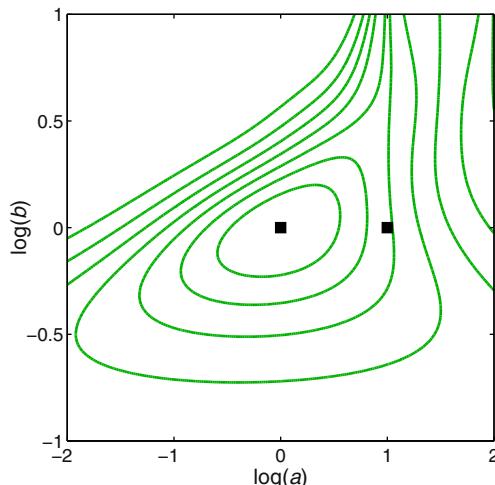


FIGURE 4.12 Shown is a contour plot of the likelihood function as a function of assumed threshold a and slope b . Square symbols correspond to the two PFs considered in Table 4.2 and shown in Figure 4.11. Contour lines correspond to $L(a, b | \mathbf{y}) = 0.5 \times 10^{-5}, 1 \times 10^{-5}, \dots, 4 \times 10^{-5}$.

where $LL(a, b | \mathbf{y})$ is the “log likelihood” and $p(y_k | x_k; a, b)$ is as defined above. Since the log-transform is monotonic, the maximum value of $LL(a, b | \mathbf{y})$ and that of $L(a, b | \mathbf{y})$ will occur at corresponding values of a and b . One reason for performing our calculations on the log-transform is that, with increasing N , likelihoods become vanishingly small and may, in many practical applications, become too small to be represented (other than as “0”) as (64-bit) data type “double” in MATLAB (the smallest positive number that can be represented by a double is 2.22507×10^{-308}).

Note that, in the above, the probability of a correct response for the observer on any trial is assumed to be a function only of stimulus intensity. In other words, the probability of a correct response given stimulus intensity x is assumed to be identical, regardless of whether it is the first trial an observer performs, or the last trial, or any in between. We will call this the “assumption of stability.” Due to practice and fatigue effects, the assumption of stability is almost certainly never strictly true. Another assumption that is implicitly made, when we assume that the probability of a correct response is a function of stimulus intensity only, is what we refer to as the “assumption of independence.” The assumption of independence states that whether an observer responds correctly on any trial is affected by stimulus intensity only and is not affected by whether the observer responded correctly or incorrectly on any other trial. In practice, the assumption of independence is also almost certainly never true. An example of a violation of this assumption would be if, after a series of incorrect responses, an observer becomes frustrated and increases her concentration and attention in order to optimize the probability of getting the next trial correct.

We may include the guess rate and/or the lapse rate as parameters in the log likelihood function in order to estimate them as well. In an M -AFC task the guess rate is known, but situations can be imagined where the guess rate is not known and it may need to be estimated. In practice, we do not know what the value of the lapse rate is. In theory, we can estimate the lapse rate, but we might run into problems when we attempt to do so. The issue of the lapse rate is discussed in some detail in [Box 4.6](#).

An analytical solution to the problem of finding the maximum value of the log likelihood is in most practical situations difficult, if not impossible, to accomplish. However, we may use an iterative search algorithm to find the maximum value of the log likelihood and the corresponding estimates of free parameters to any desired finite degree of precision. For example, the Nelder–Mead simplex method ([Nelder and Mead, 1965](#)) is well-suited to this problem. A solid understanding of the fitting process will help to prevent poor fitting practices and avoid failed fits. [Box 4.7](#) explains the details of maximum likelihood fitting.

4.3.3.1.3 ERROR ESTIMATION

The above section describes how to find the best-fitting parameters of a PF using the maximum likelihood criterion. Because our estimates are based on what is necessarily a limited number of observations we realize, though, that they are exactly that: estimates. The estimates we have obtained from our sample will not be exactly equal to the true parameter values. If we were to repeat our experiment under exactly identical conditions and estimate our parameter values again, our second set of estimates would also not be equal to the true parameters, nor would they be equal to our original estimates. This is simply due to noise; our observers have a noisy brain, our stimuli may have a stochastic element, etc.

BOX 4.6
THE LAPSE RATE

The lapse rate parameter is generally considered a nuisance parameter. Even though (generally) we do not care about its value—see, e.g., [van Driel et al. \(2014\)](#) for an exception—not estimating its value leads to systematic errors in the estimates of the parameters we do care about. The underlying problem is that there is some redundancy between the lapse rate parameter and the threshold and slope parameters. As a result, the values of the lapse rate, the threshold, and the slope are correlated. This is illustrated in [Figure B4.6.1](#). This figure shows the likelihood function of some hypothetical data. The placement of stimuli was typical of the placement by the adaptive Psi method (Chapter 5). The larger volume contains the highest 75% of likelihoods, while the smaller volume contains the highest 10% of likelihoods. The correlation among the parameters displays itself in the figure by the slant of the high-likelihood regions. For example, low lapse rates go together with high thresholds and shallow slopes, while high lapse rates go together with low thresholds and steep slopes. Note also that the volumes are rather stretched out along the direction of the lapse rate axis. What this means is that there is not a lot information as to the value of the lapse rate. The data are fairly consistent with a lapse rate around 0.01; they are also fairly consistent with a lapse rate around 0.07. The uncertainty in the value of the lapse rate is in and of itself not a problem; after all, we do not care about its value. However, due to the correlations among parameters, uncertainty in the

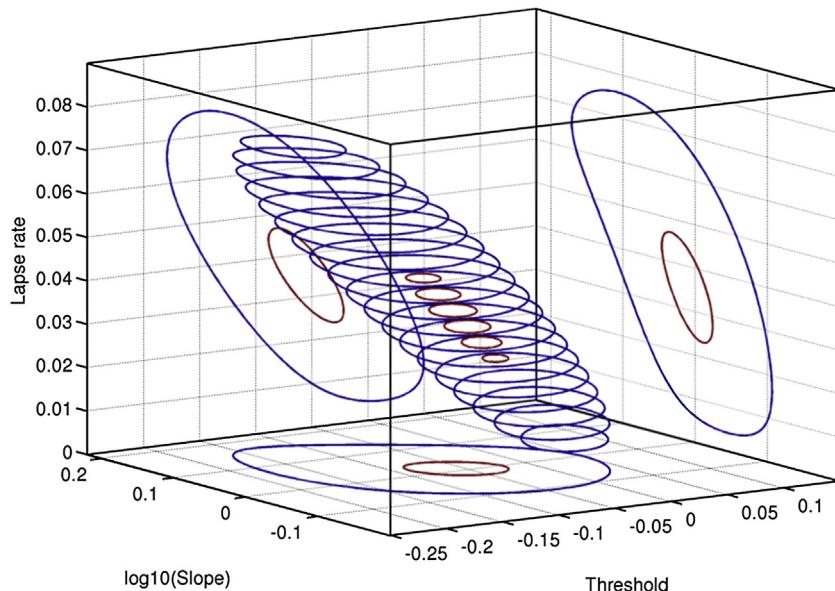


FIGURE B4.6.1 The likelihood space of a PF as a function of threshold, $\log_{10}(\text{slope})$, and lapse rate for some hypothetical data that used a placement pattern stereotypical of the Psi-adaptive method (Chapter 5). See text for details.

Continued

BOX 4.6 (*cont'd*)

value of the lapse rate adds to uncertainty in the value of the threshold and slope parameters. To see this, let us imagine that we know the value of the lapse rate to be 0.07. This would reduce the likelihood function to the single slice for which the lapse rate equals 0.07. Our estimate for the threshold would be around -0.15 and our estimate for the log-transformed slope would be around 0.1. If, on the other hand, we knew the value of the lapse rate to be 0.1, our estimate of the threshold would be around 0 and our estimate of the log-transformed slope would be around -0.05 . In reality, however, we do not know the value of the lapse rate and, as such, our data are consistent with both of these estimates (and many in between).

Lapses have long been recognized as a problem when it comes to fitting a PF (e.g., Hall discussed the issue as early as 1981). However, until relatively recent researchers often ignored the issue by simply assuming that observers did not lapse and fixed the lapse rate at a value equal to 0. This was often done implicitly, by ignoring the issue and leaving the lapse rate parameter out of the equation of the PF that was fitted. Many studies (e.g., [Swanson and Birch, 1992](#); [Treutwein and Strasburger, 1999](#); [Wichmann and Hill, 2001](#)) have investigated the effect of lapses and have shown that fixing the lapse rate at a value other than the generating lapse rate leads to bias in the estimates of the threshold and the slope.

What can we do to minimize the effect of lapses? A few suggestions have been offered. [Treutwein and Strasburger \(1999\)](#) proposed to include the lapse rate parameter as a free parameter in the estimation procedure and demonstrated that this reduces bias. [Wichmann and Hill's \(2001\)](#) later claim that allowing the lapse rate to vary effectively eliminates bias in the threshold and slope parameters is simply false, as one of us has shown ([Prins, 2012](#)). When this strategy is used, the value of the lapse rate has to be constrained using some sort of prior, such as a uniform distribution between 0 and 0.06 ([Wichmann and Hill, 2001](#)) or a beta distribution ([Treutwein and Strasburger, 1999](#)) during the fit. One reason for this is that data obtained in a psychophysical experiment generally contain very little information as to what the value of the lapse rate may be. [Prins \(2012\)](#) has shown that for some of the placement regimens considered by [Wichmann and Hill \(2001\)](#), the estimate of the lapse rate was virtually uncorrelated with the value of the generating lapse rate. These are placement regimens in which all stimulus intensities are at relatively low levels of performance. [Swanson and Birch \(1992\)](#) proposed to include a number of "free trials," trials at which the stimulus intensity is so high that it can be assumed that an error with such trials is the result of a lapse. The lapse rate is then estimated from the proportion of incorrect responses to free trials. The remainder of the data is then fit while the lapse rate is fixed at the value estimated from the free trials. [Prins \(2012\)](#) also proposed to include a number of free trials, but to fit all parameters simultaneously using a model that assumes that incorrect responses for the free trials can only be due to lapses. Specifically, the model fitted was

$$\begin{aligned}\psi(x) &= \gamma + (1 - \gamma - \lambda)F(x; \alpha, \beta) && \text{when } x < \text{APL} \\ \psi(x) &= 1 - \lambda && \text{when } x = \text{APL},\end{aligned}$$

where APL (Asymptotic Performance Level) is the stimulus intensity for free trials. This was shown to result in estimates for the threshold and slope that were virtually free from bias.

BOX 4.6 (*cont'd*)

Another manner in which to reduce the negative effects of lapses is to fit different conditions of an experiment simultaneously allowing thresholds (and slopes) to vary between conditions, while constraining the lapse rate to be equal between conditions (see Chapter 9). The advantage is that the estimate of the lapse rate will be based on all trials in the experiment and will thus better reflect the actual lapse rate. The assumption that lapse rates are equal between different experimental conditions is debatable, of course. It might not be reasonable to assume that lapse rates are identical between conditions, especially when different conditions are measured in separate blocks of trials. The overall task difficulty may vary between conditions and this may affect the lapse rate. When trials from different experimental conditions are randomly intermixed, however, it does seem reasonable to assume that the lapse rate is equal between conditions. After all, the occurrence of a lapse is by definition independent of the characteristics of the stimulus.

One should also take into account that bias in the estimates of threshold and slope might not be a concern. In psychophysical research, researchers are generally not concerned with the absolute value of thresholds or slopes; rather, they are interested in whether and to what degree an experimental manipulation affects their values. In other words, researchers are often concerned in differences among parameter values rather than their absolute values. One of us (Prins, 2010) has shown that while absolute values of thresholds and slopes are affected systematically when the assumed or fitted value for the lapse rate differs from the actual value, the value of differences among parameter estimates obtained in different conditions are not significantly affected.

Generally speaking, there is no single rule of advice to give as to how lapses should be dealt with. We will give a few common sense rules of thumb here, most of which follow from the above considerations. First, a researcher should consider whether the absolute values of thresholds and slopes or, instead, the differences among these parameters obtained under different experimental conditions are of interest. In case of the latter, there may not be much of a concern and one may wish to use a fixed value for the lapse rate.

In case one does decide to allow the lapse rate to vary during fitting, one should design their experiment such that the data obtained will contain some information regarding the true value of the lapse rate. This is actually a rule that applies not only to the lapse rate. The general form of the rule states that a parameter should not be estimated unless the data contain sufficient information to support the estimate (see also Box 4.7). If possible, one should consider including an extremely high stimulus intensity such that trials presented at this intensity can safely be considered free trials and a model can be fitted that incorporates this assumption (Swanson and Birch, 1992; Prins, 2012). In case it can be assumed that the lapse rate is constant between different experimental conditions, one should consider fitting all conditions simultaneously while estimating a single, shared lapse rate.

In case one decides to fix the lapse rate during fitting, however, one should not include trials that use an extremely high stimulus intensity, since these trials will have the largest biasing effect on the value of the threshold and slope parameters when the assumed and actual lapse rate do not coincide. Also, when a fixed lapse rate is used, the value of 0 should be avoided (e.g., Hall, 1981). It is much better to assume a small value (e.g., 0.02) for the lapse rate when the actual lapse rate equals 0 than it is to assume a value of 0 when the actual lapse rate equals 0.02.

BOX 4.7

MAXIMUM LIKELIHOOD FITTING PROCEDURE

The best-fitting PF, according to the maximum likelihood criterion, is that PF that has the highest probability of generating the exact same responses to the stimuli that your observer generated (Section 4.3.3.1). Unfortunately, in most practical situations there is no analytical way in which to do this. Instead, a search for the best-fitting PF has to be performed. This turns out to be tricky. There are an infinite number of candidates to consider! Understanding how maximum likelihood fitting works will help one get better fits and avoid mistakes. We will explain by example. Let's say you perform a 2AFC experiment. You use nine different stimulus levels (log stimulus levels run from -0.4 to 0.4 in steps of 0.1) and 10 trials at each of these stimulus levels. The number of correct responses at these stimulus levels are $6, 6, 7, 9, 10, 7, 9, 10$, and 10 , respectively. These proportions correct are shown by the black symbols in Figure B4.7.1. It seems that an appropriate set of stimulus levels was used in that proportions correct go from near guessing levels to perfect with some intermediate values in between.

How do we find the PF that has the maximum likelihood? PFs are completely determined by their shape (Logistic, Weibull, etc.) and the values of their four parameters. In this example we used a 2AFC task and that means the guess rate equals 0.5 . Let us further assume that the lapse rate equals 0 . We will use a Gumbel (or log-Weibull) as the shape of the PF. This reduces our problem to finding that combination of a threshold value and a slope value that maximizes the likelihood. How to find it? A very commonly used algorithm is the Nelder–Mead simplex search (Nelder and Mead, 1965). Imagine a tripod standing on the side of a hill. Longitude corresponds to threshold, latitude corresponds to slope, and the height of the hill corresponds to likelihood. The tripod's goal is to get to the top of the hill. The position of each of the three feet of the tripod defines a combination of a threshold value and a slope value (in other words, each foot represents a possible PF). The triangle defined by the positions of the feet is termed a "simplex." The tripod calculates the likelihood (height of the hill) associated with each of the three PFs. It then tries to swap out the position of the lowest foot with a position that is higher up the hill. It uses a couple of simple rules to find a better position for its lowest foot. For

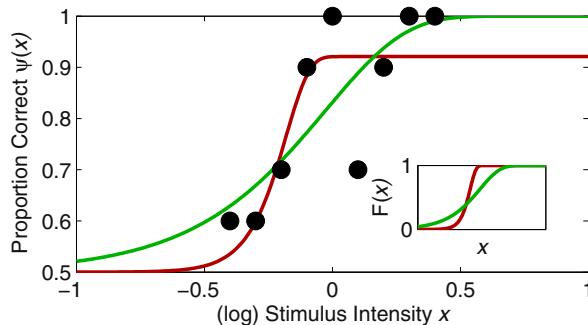


FIGURE B4.7.1 Some hypothetical data, a fit corresponding to the maximum likelihood (green) and a fit corresponding to a local maximum in the likelihood function (red).

BOX 4.7 (*cont'd*)

example, the first thing it tries is to swing its lowest foot to the other side of the line that runs between the other two feet and calculate the likelihood of the PF corresponding to this candidate position of the foot. If the foot is higher up the likelihood hill than the other two feet it will consider a second candidate position that is farther in the same direction. If this position farther out is even higher it decides to put the foot there (this “move” is called an expansion). Once the tripod has moved its lowest foot to a higher position on the hill, there is now a new lowest foot. It will now go through the same set of rules to move this new lowest foot to a higher position on the hill. This process repeats itself until it has been determined that it is near the top of the hill or until some maximum criterion number of iterations has been reached. The process generally works very well; if the tripod starts on the side of a hill, it will generally find its way to the top of the hill. [Figure B4.7.2](#) shows the uphill path the tripod makes for our problem above when started at a rather arbitrary point on the hillside. The foot positions of the initial simplex are indicated by the open circles. Note that it does indeed make it to the top. Note also that along the way the simplex changes its step size. For example, the first four moves are all “expansions” (see above) and thus the stride length increases each of the first four moves. Also note that when the tripod gets close to its target it tends to make smaller and smaller strides.

The simplex will never make it to the exact position of the highest point on the hill but it can get arbitrarily close to it. The decision to stop the search is based on the size of the simplex and is set using “tolerance” values. There are actually two tolerance values involved: TolX has to do with the values of the parameters, and TolFun has to do with the likelihood values. Simply

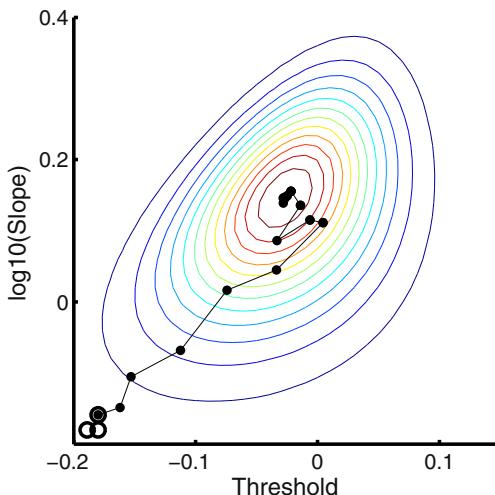


FIGURE B4.7.2 The path taken by the simplex method toward the maximum likelihood for the data shown in [Figure B4.7.1](#). The three circles in the bottom left display the (rather arbitrary) starting positions of the simplex's vertices.

Continued

BOX 4.7 (*cont'd*)

put, the search stops when all feet of the tripod are no farther than the value of TolX from the best foot in either the threshold direction or the slope direction (measured in whatever units threshold and slope are measured in) AND the log likelihood associated with the worst foot is no farther than the value of TolFun below the log likelihood associated with the best foot. In other words, when the simplex is visualized in three dimensions, with the third dimension corresponding to the value to be maximized, the search stops once the simplex fits in a box with length and width no larger than TolX and a height no more than TolFun. Both TolX and TolFun are, by default, set to 1e-6 (0.000001) in Palamedes. Once the tolerance criteria have been met the search stops and the threshold and slope values corresponding to the position of the best foot are reported as the parameter estimates. The solution converged on using the above strategy is the green curve in [Figure B4.7.1](#).

This principle will readily generalize to higher dimensional parameter spaces. For example, let's say you wish to allow the lapse rate to vary when you fit the above PF. You now have three parameters to estimate (threshold, slope, and lapse rate). Nelder–Mead will work, in principle, for parameter spaces of any dimension. The simplex will always have $n + 1$ vertices for an n -dimensional parameter space. A graphical representation of the three-dimensional (threshold \times slope \times lapse rate) likelihood function for the data in [Figure B4.7.1](#) is given in [Figure B4.7.3](#).

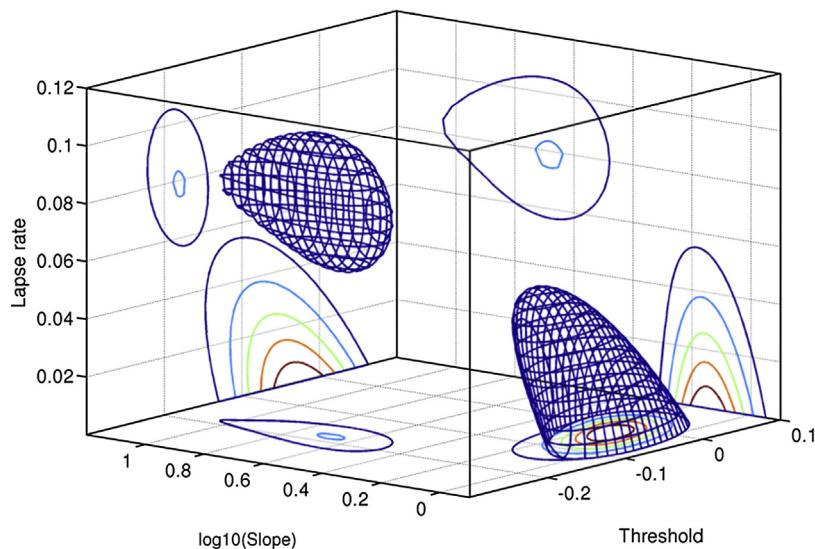


FIGURE B4.7.3 A graphical representation of the three dimensional (threshold \times slope \times lapse rate) likelihood function for the data in [Figure B4.7.1](#). The mesh solids display regions where the likelihood is greater than one-fourth the maximum likelihood.

BOX 4.7 (*cont'd*)

The mesh volumes in [Figure B4.7.3](#) contain those regions where the likelihood is greater than one-fourth the maximum likelihood and the marginal contours show marginal maxima. The hill analogy doesn't work very well in a three-dimensional parameter space, but you can now perhaps think of the simplex as a tetrapod hovering in parameter space that has, say, a thermometer at the end of each of its four "feet" and is "stepping" through a three-dimensional space looking for a heat source. Each step involves moving the coldest foot to a warmer location while keeping the other three feet in position. The observant reader will have realized the problem with the current data. There are two heat sources in the likelihood function shown in the figure, one hotter than the other. Note that there is still only one unique location that has the absolute "maximum likelihood." This happens to have a lapse rate equal to 0 and so it is the solution we found before (when we fixed the lapse rate at 0). A problem occurs if the simplex starts near the smaller heat source. Each step the tetrapod makes will be toward a warmer spot and as a result it will move toward the nearer, smaller heat source and converge on its center. Its center, by the way, corresponds to the PF shown in red in [Figure B4.7.1](#). This solution is known as a local maximum: While it is the maximum point in its immediate neighborhood, it is not the absolute maximum in the parameter space. The inherent problem is that the simplex will only feel around in its immediate proximity. As such, it will find the top of whatever hill it was positioned on when it started its search. Importantly, the simplex has no way of knowing whether the top of the hill (or the source of heat or whatever) it eventually converges on is the top of the highest hill (or the center of the biggest heat source).

Thus, in order to avoid ending up in a local maximum, it is important that we find a starting position for our simplex that is on the hillside of the highest hill in the likelihood space. One way in which to do this is to find a starting position by first performing a "brute force" search through a grid defined across the parameter space. The grid is defined by specifying a range of values for all of the free parameters. Likelihood values are then calculated for all combinations of parameter values contained in the grid. The combination with the highest likelihood serves as the starting point for the simplex. Lucky for us, today's computers can perform a brute force search through a finely spaced search grid in a very short period of time.

Note that the search grid that searches for a starting point for a simplex should include all the free parameters that will be considered by the simplex search. For example, the lesser heat source shown in [Figure B4.7.3](#) will have a higher peak likelihood than the greater heat source when we fix the lapse rate at 0.06. Thus, a brute force search through threshold and slope values that uses a fixed value of 0.06 for the lapse rate would result in an initial position for the simplex near the lesser heat source. A subsequent simplex search that includes the lapse rate but starts at the initial position we found while fixing the lapse rate at 0.06 would then converge on the maximum value in the lesser heat source. All maximum likelihood fits to single PFs in Palamedes will be performed by first evaluating all PFs contained in a user-specified grid, then performing a simplex search starting at the function in the grid that was found to have maximum likelihood.

In case a search results in a local maximum, as described above, the Nelder–Mead algorithm will eventually meet the tolerance criteria and report a successful convergence. To

Continued

BOX 4.7 (cont'd)

Nelder–Mead, simply put, reaching the top of the hill that it started on means that the search was successful. It has no way of knowing whether the peak it has reached corresponds to a local or a global maximum. This problem of local maxima, however, can be adequately dealt with using a brute-force search, as described above. Sometimes, however, the Nelder–Mead algorithm will never reach a local or global maximum. Unless we force it to give up, it would for all eternity attempt to locate a maximum. Routines that implement the Nelder–Mead algorithm will set a maximum number of iterations and/or function evaluations to be performed before giving up the search. Remember that every iteration might involve more than one function evaluation (for example, an expansion requires two new function evaluations). When a search is terminated because the maximum number of iterations or function evaluations has been exceeded and the tolerance values have not been met the search has failed. Often when this happens, the position of the simplex in parameter space is nowhere near where the researcher was expecting to find the maximum likelihood.

In order to illustrate why a search might fail we will go through an example for which the search fails. Figure B4.7.4 shows some hypothetical data from a 2 AFC experiment. There were five stimulus intensities used with 10 trials at each of these intensities. A simplex search for a threshold value and a slope value while keeping the guess and lapse rates fixed (we used 0.5 and 0.03, respectively) will not converge. When we attempted to fit a Gumbel function to these data using the Palamedes function `PAL_PFML_Fit`, the function indicated that the search failed and reported that the best-fitting parameter values at termination were -4550.9613 for the threshold and 0.000035559 for the slope. At first sight, these numbers seem outrageous and arbitrary. However, keep in mind that the Gumbel is a monotonically increasing function of the data, but that the data follow an overall downward trend. As we will argue, the simplex search failed to find a maximum because there was no maximum to be found. The data simply do not support the estimation of a threshold and a slope. The seemingly outrageous and arbitrary fit is a typical case of “garbage in, garbage out.” Or is it?

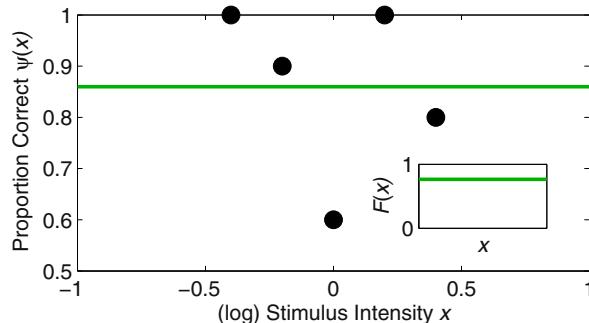


FIGURE B4.7.4 The green line shows a Gumbel function with threshold equal to -4550.9613 and slope equal to 0.000035559 , guess rate equal to 0.5, and lapse rate equal to 0.03. Within the stimulus range shown, it is asymptotically near a horizontal line at 0.86, the best fit to these data that can be obtained under the constraint that a Gumbel function is to be fit. See text for details.

BOX 4.7 (*cont'd*)

The green horizontal line in [Figure B4.7.4](#), which has a value of 0.86 (which corresponds to the overall proportion correct across all 50 trials in the example), is actually not a horizontal line. It is (a tiny part of) a Gumbel function. If by now you guessed that it is a Gumbel with threshold equal to -4550.9613 and slope equal to 0.000035559 (guess rate = 0.5 and lapse rate = 0.03) you are right. The parameter estimates the routine reported are not arbitrary at all. Even more, from a purely statistical perspective they are not outrageous, either. The PF corresponding to the parameter values returned by the routine is asymptotically near a horizontal line at the observed overall proportion correct observed. This is the best fit that can be obtained given the constraint that a Gumbel needs to be fitted. Note that there is indeed no maximum in the likelihood function. The line shown in [Figure B4.7.4](#) can always be made to provide a little better fit by making it a little straighter and more horizontal by moving the threshold to an even lower value and adjusting the slope value such that, within the stimulus intensity range, the function has values that are asymptotically close to 0.86.

Any researcher who obtains the data shown in [Figure B4.7.4](#) will readily understand that no PF can be fitted to them. It is a little harder to see why problems arise when the data look fine (say 6, 8, 8, 10, and 9 correct responses, respectively, for the five stimulus levels, each again out of 10 trials). Even though the fit to these data is fine (threshold = -0.1040 , slope = 1.836), finding a standard error using a bootstrap routine ([Section 4.3.3.1.3](#)) or determining the goodness-of-fit using Monte Carlo simulations (Chapter 9) leads to poor results. The problem is that some of the simulations performed during the bootstrap will result in simulated data, such as those shown in [Figure B4.7.4](#), resulting in failures of convergence and extreme parameter estimates for these simulations. As discussed in [Section 4.3.3.1.3](#), the standard error of a parameter is estimated as the standard deviation of the simulated parameter estimates, and standard deviations are very sensitive to outliers. Thus, even if only a single simulation results in a failed search with extreme parameter estimates, this will have a large effect on the standard error estimates.

The proper interpretation of such a result is that even though the fit to your data may have resulted in a realistic value for the slope parameter, you cannot place much confidence at all on this value. The essence of the problem with this fit is that not enough data were collected to determine reliable estimates of a threshold and a slope. Simply put, even though the fit to your data produced a realistically valued slope estimate, it is very possible that this was just luck. Consider this: an observer that behaves like the (nice) PF you fitted apparently can produce garbage data (i.e., of the kind that cannot be fit; this is why the bootstrap failed). This might suggest that a garbage observer could have produced the nice data that your observer produced. In other words, you cannot be confident that your observer is not a producer of garbage. Once again, you do not have enough data to obtain reliable estimates of a threshold and a slope.

Note that some software packages may “fail to fail” and will happily report nice-looking estimates of your parameters and their standard errors even if your data do not support these estimates. This is frustrating for us, because it seems as if Palamedes fails where other packages seem to succeed. When Palamedes fails to fit some data while another package

BOX 4.7 (*cont'd*)

appears to fit those data just fine, we encourage you to compare the log-likelihood values of the fit between the two packages and also to inspect how the fitted functions compare to your data (which is good practice anyway).

We regularly get questions from users regarding failed fits and it is almost universally the case that the user simply did not have sufficient data to estimate the parameters they were attempting to estimate. The solution here is to make sure that you have sufficient data to support estimation of the model that you wish to fit. In other words, either get more data or use a fixed value for the parameters regarding which you do not have sufficient information in your data. For example, for the data shown in [Figure B4.7.4](#), fixing the value for the slope allows estimation of a threshold, a standard error for this estimate, and a goodness-of-fit of the resulting model. It should be noted here that if a fixed slope is used in a fit, a parametric bootstrap should be avoided (use a nonparametric bootstrap instead). If you do perform a parametric bootstrap, it is pertinent that you use a value for the slope that is realistic; the value for the SE on the threshold will, in some circumstances, be determined entirely by the value for the slope that you use.

So, we will arrive at different threshold estimates when we repeat an experiment, even when we use identical conditions. We call this sampling error. This is a problem, especially when we are interested in determining whether experimental manipulations affect the parameters of our PF. We can never know for sure whether differences between parameter estimates in different conditions are a result of experimental manipulation or sampling error. However, there are methods available to us to make at least a reasonable guess as to whether or not an experimental manipulation affected the parameters of the PF. Here we will discuss a method that allows us to estimate by how much we can expect our parameter estimates to differ from the true value of the parameter.

So, we would like to gain some information as to the magnitude of the error to which our parameter estimates are subject. We will address this issue by considering a different but similar question. The question we ask ourselves is as follows: given particular values for the true parameters and an infinite number of experiments like ours, all resulting in estimates of the true parameters, what degree of error are these estimates subject to? More generally, given particular values for the true parameters, what are the probability density functions of our parameter values? This is, of course, a hypothetical question because we will never know the true values of the parameters, and we certainly do not wish to repeat our experiment an infinite number of times. Not only is this question hypothetical, but it also seems backward. When we have completed our single experiment and fit our data we will know the parameter estimates and would like to know the true parameter values, but the question assumes knowledge of the true parameters and asks about the parameter estimates. As backward as this question may appear, let us attempt to answer it anyway, and see where that leads us.

The distribution of parameter estimates resulting from a hypothetical infinite number of experiments is known as a “sampling distribution.” In certain circumstances sampling distributions for parameters may be derived analytically. These are typically algebraically simple parameters, such as the mean. As an example, let us consider the sampling distribution of the mean. By the well-known Central Limit Theorem (e.g., Hays, 1994), given a population of scores with mean μ and standard deviation σ , the sampling distribution of the (sample) mean \bar{X} will have a mean ($\mu_{\bar{X}}$) equal to μ and a standard deviation ($\sigma_{\bar{X}}$) equal to σ/\sqrt{N} , where N is the sample size. Moreover, if our sample size N is large enough, the sampling distribution of the mean will closely approximate the normal distribution. By way of example, let us say that we have a population with known mean $\mu = 100$ and standard deviation $\sigma = 20$. If we were to collect an infinite number of samples of size $N = 100$, and for each of these samples calculated the sample mean \bar{X} , the resulting distribution of sample means would have mean $\mu_{\bar{X}}$ equal to 100 and the standard deviation of sample means (the standard error) would be equal to $\sigma_{\bar{X}} = \sigma/\sqrt{N} = 2$. If we use the normal distribution as an approximation to the sampling distribution (which is appropriate given that our $N = 100$), we are now in a position to make some interesting statements. For example, approximately 68% of our sample means would have a value between 98 and 102 (i.e., $\mu_{\bar{X}} \pm \sigma_{\bar{X}}$). This may, of course, be paraphrased as: 68% of sample means would be in error by less than two points. We can also determine that a very large proportion (>0.9999) of sample means would have a value between 92 and 108, which may be paraphrased as follows: almost all sample means would be in error by less than eight points. We may paraphrase even further and state that if we were to take a single sample of size $N = 100$ from this population there is a probability of about 68% that the sample mean will be in error by less than two points and also that the probability that the sample mean will be in error by less than eight points is near unity.

Let us now envision a situation in which we have a large population of scores and we wish to estimate the mean of this population. Somehow we know that the standard deviation of the population is equal to 20. We reason that if we take a sample of $N = 100$ and calculate the sample mean, the probability that this sample mean will be in error by less than two points is approximately 68%. Next, we actually do take a sample of $N = 100$, and we calculate the sample mean, which comes out at, say, 50. It sounds as if we can now argue that there is a 68% probability that our sample mean is in error by less than two points and thus, that the probability that the population mean has a value between 48 and 52 is 68%. We cannot make this argument though, simply because the experiment is over and the sample mean is known, and it is either in error by less than two points or it is not. Similarly, the population mean is not a random variable. That is, it either has a value between 48 and 52 or it does not. Despite the fact that we cannot use the term probability, it seems like a reasonable argument. We present our statement using the term “confidence.” We might say: “We are 68% confident that the population mean has a value between 48 and 52.” Mission accomplished!

Thus, once we have conducted our psychophysical experiment, we would like to know the characteristics of the sampling distributions of our parameters. From this, we could determine by how much we can expect our estimate for, say, the threshold to be in error. To determine the sampling distribution, we would first need to know the population parameters, which of course we do not (if we did, there would be no need for our experiment). This problem also exists when we wish to estimate the mean of a distribution, as we did above. Our

example there assumed knowledge of the population standard deviation. In any practical situation, the population standard deviation will be just as unknown as the population mean we wish to estimate. The solution is to consider the sample to be representative of the population and to use the sample to estimate the population parameters. In the case of the mean we would have to estimate the population standard deviation from our sample (our sampling distribution would then have to be the t-distribution, but that aside). We could do the same for our purposes here. Even though we do not know the true PF's parameters, we have estimates of them based on our sample. For the purposes of finding our sampling distribution we will use these as estimates of the true PFs parameters and derive a sampling distribution for a PF with these parameters.

Our second problem is that even if we assume our population parameters to equal our sample parameters, no one as of yet has analytically derived the sampling distribution of, say, the threshold parameter of a PF as estimated by the maximum likelihood method. However, we may approximate our sampling distribution by simulating our experiment many times using the estimated true PF parameters to define the generating function. That is, we simulate an observer to act according to our estimated true PF. We run this observer through simulations of our experiment many, many times. Each time we derive estimates of the PFs parameters. These estimates then serve as our empirically derived sampling distribution.

An example is probably in order here. We wish to measure the threshold of a human observer on a 2AFC psychophysical task. We use the method of constant stimuli to collect 100 responses at each of seven equally spaced (in log units) stimulus levels: $-3, -2, \dots, 3$. The observed number of correct responses at these stimulus levels are: 55, 55, 66, 75, 91, 94, and 97, respectively. We can use Palamedes (or similar software) to find the best-fitting PF. We wish to estimate the threshold and slope, but we fix the guess rate at 0.5 and we will also assume that the lapse rate equals 0. The maximum likelihood estimates for the threshold and slope parameters are -0.1713 and 0.9620 , respectively. We now wish to obtain the sampling distribution of our parameter estimates. This will allow us to get some idea as to how much error our parameter estimates might be subject to. We imagine an observer whose PF's true parameters are those we have estimated from our observer (i.e., $\log \alpha = -0.1713$, $\beta = 0.9620$). We then simulate this observer as a participant in our experiment many, many times. In the simulations, we use the same stimulus intensity values and the same number of trials at each of the stimulus intensities as we did for our human observer. From the results of each simulated experiment, we estimate the PF's parameters. For all these estimates we know exactly by how much they are in error, because we know the true parameters of the generating PF. Our distributions of these estimates are our empirically derived sampling distributions. The left panels of [Figure 4.13](#) show sampling distributions of the estimates for the threshold and slope based on $B = 40,000$ simulated experiments. The results of all these simulated experiments were generated by a PF with known parameter values ($\log \alpha = -0.1713$, $\beta = 0.9620$). These generating parameters are indicated in the figure by the vertical lines through the histograms.

Let us make a few observations. First, the mean of the sampling distribution for the threshold equals -0.1761 . This is quite close in value to the threshold of the generating PF. This means that, although some estimates were too high and others too low, on average they were approximately on target; our fitting procedure results in threshold estimates that have little bias, at least under the conditions of this experiment. Second, the mean of the

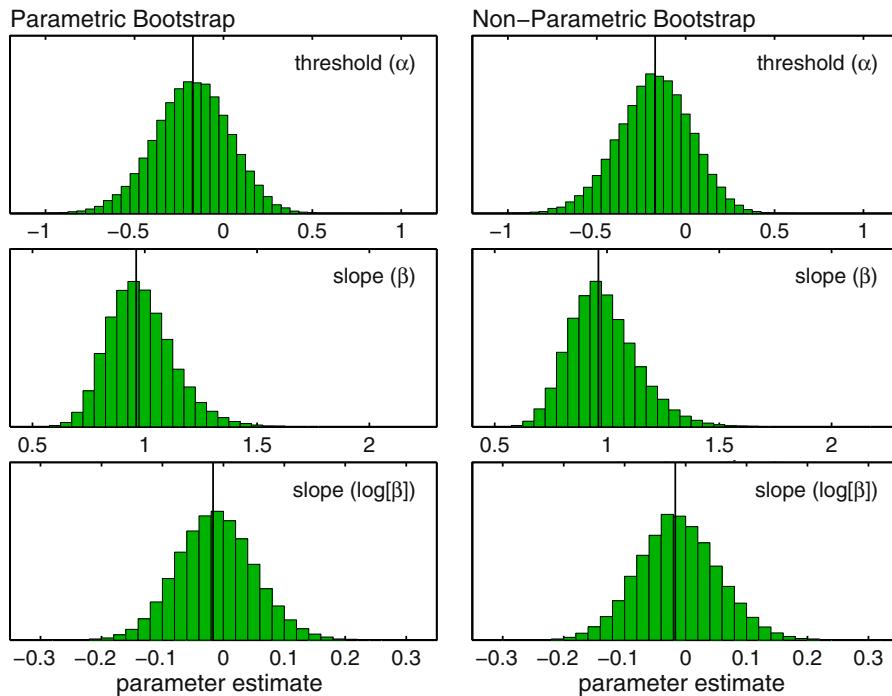


FIGURE 4.13 Empirical sampling distributions of the threshold and slope parameters of a PF. Distributions on the left were obtained using a parametric bootstrap and those on the right using a nonparametric bootstrap. Vertical lines indicate the best-fitting values of the threshold and slope of the results of the human observer, which also are the values used to generate the bootstrap samples.

sampling distribution for the slope equals 0.9835. Again some estimates were too high, some were too low, but now the average, frankly, is a little high, too (compared to generating slope 0.9620). Thus, our estimates for the slope parameter have, on average, overestimated the true slope parameter a bit. We also note that, whereas the sampling distribution of thresholds is symmetrical, that for slopes is positively skewed. In skewed distributions, the median is sometimes used as a measure of central tendency. The median of the sampling distribution for the slope equals 0.9705, closer to the true (i.e., generating) value, but still too high. Another common method to derive a measure of central tendency from an asymmetric distribution is to apply some transformation that makes the distribution (more) symmetric before calculating the measure of central tendency. When we apply a log-transform to the slope estimates, the distribution becomes virtually symmetrical (see bottom row of Figure 4.13). This suggests that a geometric mean might be appropriate as a measure of central tendency. In order to calculate a geometric mean, one calculates the mean of log-transformed values, then takes the antilog of this mean to return its value to the original metric. The geometric mean of the simulated slopes equals 0.9730, again a little higher compared to the generating value.

The bias of our estimates says, of course, nothing about how much any given individual estimate may be in error. For that we need to look at the standard deviation of our sampling

distribution of parameter estimates (i.e., “the standard error of estimate” or SE). The SE is calculated as

$$SE_{\hat{\alpha}} = \sqrt{\sum_{b=1}^B (\hat{\alpha}_b - \bar{\hat{\alpha}})^2 / (B - 1)} \quad (4.15)$$

where B equals the number of simulations (here, $B = 40,000$), $\hat{\alpha}_b$ is the threshold estimate resulting from bootstrap simulation b (actually, since we have used a log-transform of stimulus intensities, in this particular example we should use the log-transformed value of the threshold estimate), and $\bar{\hat{\alpha}}$ is the mean threshold estimate (again, here, we should use the mean of the log-transformed threshold estimates).

The standard error of the threshold (calculated as in Eqn (4.15)) equals 0.2121. Since we note that the shape of the sampling distribution is approximately normal, this would mean that about 68% of threshold estimates deviated from the generating threshold by less than 0.2121 log units. We may assume that the true generating PF of our human observer is much like that used to create these sampling distributions (after all, we modeled our simulated observer after our human observer), and thus the sampling distribution of our human observer’s threshold would be much like that shown in the figure. We can now make statements such as the following: “We can be 68% confident that our human observer’s true threshold has a value in the range -0.1713 ± 0.2121 .”

Calculated analogously to the standard error of the threshold, the standard error of the slope estimate is equal to 0.1461. The shape of this sampling distribution deviates systematically from normal, so we cannot translate this value easily into a confidence interval. Also, as the sampling distribution of the slope is asymmetrical, we might wish to determine two standard errors: one for estimates that were below the generating value and the other for estimates that were above the generating value. The SE for low estimates equals 0.1300 and that for the high estimates is 0.1627. More elegantly, perhaps, is to report the slope itself and its SE in log units (remember that the distribution of $\log(\beta)$ is approximately symmetrical). We would simply calculate the standard error as the standard deviation of the distribution of log-transformed slope values. The standard error of the log-transformed slope calculated thus equals 0.0636. We might then report our estimate of the slope and its SE in this manner: “Our maximum likelihood estimate of $\log(\beta)$ is -0.0168 (i.e., $\log_{10}(0.9620)$) with standard error equal to 0.0636.”

A good case can be made for an alternative calculation of standard errors. In Eqn (4.15), we have used deviations from the sampling distribution’s mean threshold ($\hat{\alpha}_b - \bar{\hat{\alpha}}$) as our basis for the calculation of the SE. However, these values are not truly the errors of estimate. We know the true threshold that generated the sampling distribution, thus we need not estimate it. The true error of estimate of any $\hat{\alpha}_b$ is not $(\hat{\alpha}_b - \bar{\hat{\alpha}})$, rather it is $(\hat{\alpha}_b - \alpha_g)$, where α_g is the threshold of the PF that generated the sampling distribution. If we use instead the equation

$$SE_{\hat{\alpha}} = \sqrt{\sum_{b=1}^B (\hat{\alpha}_b - \alpha_g)^2 / B} \quad (4.16)$$

to calculate the standard error of estimate, we find that the value is nearly identical in value to that obtained using Eqn (4.15). Note that the denominator now is B , since we used the known generating α_g and did not estimate it by $\bar{\hat{\alpha}}$. Whether we use $\bar{\hat{\alpha}}$ or α_g has very little effect on the

obtained standard error of estimate; this is because $\bar{\alpha}$ and α_g are nearly identical in value. However, when we make the same adjustment in the calculation of the standard error of estimate of the slope, we do arrive at somewhat different estimates (0.1477 vs 0.1461). The difference can be attributed to the bias in $\bar{\beta}$ relative to β_g . When we calculate separate estimates for the low estimates and the high estimates using β_g , we arrive at SEs of 0.1199 (at the lower end) and 0.1690 (at the higher end), compared to 0.1300 and 0.1627, respectively, when we use $\bar{\beta}$.

The procedure above is referred to as a “parametric” bootstrap. The sampling distribution was created using a simulated observer characterized by the parameters of the best-fitting PF to our human observer’s data. In other words, we have assumed that our human observer’s true PF is, in this case, the Logistic function. The accuracy of our obtained estimated standard errors relies on this assumption being correct.

We may also perform a “nonparametric” bootstrap procedure, in which we do not summarize our human observer by a parametric description of his or her assumed PF to generate the sampling distribution. Rather, we use the observed proportions correct at each of the stimulus intensities directly to generate the bootstrap simulations of the experiment. For example, here the human observer responded correctly on proportions 0.55, 0.55, 0.66, 0.75, 0.91, 0.94, and 0.97 of trials at each of the stimulus intensities, respectively. We may run our simulations without first summarizing this performance by a PF, as we did above. We may instead use these proportions directly in our simulations. That is, at each of the trials at stimulus intensity 23, our simulated observer will generate a correct response with probability 0.55, etc. The panels on the right-hand side of [Figure 4.13](#) display the sampling distributions of the threshold and slope generated by a nonparametric bootstrapping procedure. The SEs obtained are close in value to those obtained by the parametric procedure (0.2059 vs 0.2121 for the SE in threshold and 0.1530 vs 0.1461 for the SE in slope). The two methods generate very similar results here, because our human observer’s proportions correct are well-described by our fitted PF. When our fitted PF is not a very good fit, the two bootstrapping methods might generate somewhat different results. Thus, it is good practice to perform a goodness-of-fit test of the PF and perform a parametric bootstrap only when the PF fits the data well. When the goodness-of-fit is unacceptable, one should perform a nonparametric bootstrap.

A few more words of caution should be given here. The accuracy of our SEs depends critically on the accuracy of our parameter estimates based on our human observer (e.g., [Efron and Tibshirani, 1993](#); [Kuss et al., 2005](#)). The SEs estimated by the parametric bootstrap are actually those that are associated with the function that we used to generate our bootstrap samples. They are accurate only to the extent that our human observer’s PF corresponds to our estimate of it. We should realize that our estimated slope for the human observer’s PF might be biased. When we use a positively biased estimate as the slope of the generating PF in our bootstrap simulations, the bootstrap simulations will be generated by a less noisy observer than our human observer. This will lead to an underestimation of the SE of the threshold parameter. For this reason we might prefer to perform the nonparametric bootstrap procedure, which does not involve estimating the human observer’s threshold or slope in order to perform the simulations.

On the other hand, when we use an adaptive method during our experiment (Chapter 5), the parametric bootstrap procedure might be preferable. This is because when we use an

adaptive procedure we may have very few observations at any utilized stimulus intensity. Consider a situation in which only one trial was presented at stimulus intensity x . Let us assume that our human observer produced a correct response on this trial. When we use the nonparametric bootstrap procedure in this case, our simulated observer will, in all of the simulations, also produce a correct response to this trial. In the extreme case where any given x was utilized on only one trial (as might happen in some adaptive testing procedures (Chapter 5)), the simulated observer will respond identically to our human observer on every trial in all simulations. In this case the parameter estimates from all simulations will, of course, be identical to the estimates of our human observer. Our SEs of estimate will then be 0.

An issue arises when not all simulations result in a successful fit (see [Box 4.7](#)). In such cases, a standard error of the parameter estimate cannot be calculated. If this happens, there is no solution that is truly elegant even though a few tempting, but inappropriate, ideas might spring to mind. One idea would be to ignore the failed simulations and calculate the standard error across the subset of simulations that did converge. Another tempting, but equally inappropriate, idea would be to generate new simulated datasets to replace those that resulted in failed fits in the original set. One more seemingly innocent, but once again inappropriate, solution would be to try the entire set of, say, 400 simulations again. We could continue to produce sets of 400 simulations until we have a set for which all 400 fits were successful. We would then calculate the standard errors across that complete set. The problem with all these ideas is that the resulting error estimates would not be based on a random sample from the population of possible simulations. Rather, the estimates would be based on a select subset of the population of simulations, namely those that can be fitted successfully.

So, what is one to do when some simulations fail to converge? Generally, fits are more likely to converge when we have fewer free parameters in our model ([Box 4.7](#)). Thus, we could fix one of the free parameters. Another manner in which to increase our chances of having all simulations fit successfully is to gather more responses, because the chances of obtaining a successful fit generally increase with an increasing number of responses. One final solution that requires that all but a very few fits converged successfully is to calculate standard errors across the successful fits only but to acknowledge to our audience (and ourselves) that our error estimate is based on a sample that was not entirely random.

4.3.3.2 Bayesian Criterion

4.3.3.2.1 BAYES' THEOREM

The likelihood associated with assumed values a and b for the threshold and slope, respectively, of the PF is equivalent in value to the probability that a PF with $\alpha = a$ and $\beta = b$ would result in the exact outcome of the experiment as we have already observed it. As discussed above ([Section 4.3.3.1](#)), this likelihood can be interpreted neither as the probability of our exact experimental outcome having occurred nor as the probability that the threshold has value a and the slope has value b . A somewhat similar issue exists in classical ("Fisherian" or "frequentist") Null Hypothesis testing. The p -value that is the inevitable final result of a classical hypothesis test, and which eventually leads us either to reject the Null Hypothesis (if $p < 0.05$ or some other criterion value) or accept it (otherwise) is, as our statistics instructors have stressed to us, not the probability that the Null Hypothesis is true. Rather, the p -value we calculate in a classical hypothesis test corresponds to (something like) the probability that our experimental

results could have occurred by sheer coincidence if, in fact, the Null Hypothesis were true. We can write this probability as $p(D|H)$, where D represents the outcome of our experiment (D stands for “data”) and H is shorthand for “the Null Hypothesis is true.” Of course, in classical testing, we do not consider D to be the exact outcome of our experiment, but rather we consider D to be a range of outcomes (specified before we started collecting our results) that are unlikely to be obtained if H were true. Notwithstanding the tremendous intuitive appeal of the validity of concluding that H is likely false if $p(D|H)$ is small, this conclusion is nevertheless without merit. To conclude that H is unlikely given our experimental results is to make a statement regarding the value of $p(H|D)$; unfortunately, we have a value only for $p(D|H)$. However, we may relate $p(H|D)$ and $p(D|H)$ using Bayes’ Theorem:

$$P(H|D) = \frac{p(H)p(D|H)}{p(H)p(D|H) + p(\bar{H})p(D|\bar{H})} = \frac{p(H)p(D|H)}{p(D)} \quad (4.17)$$

One may think of Eqn (4.17) as expressing the central Bayesian concept that we use our experimental results D as serving to adjust the probability of H as we estimated it before, considering the results of our experiment. As an illustration of this concept let us consider the entirely hypothetical case of Mr J. Doe. As part of his routine annual medical exam, Mr Doe is administered a diagnostic test D that tests for the presence of the rare medical condition H . Test D is known to be highly accurate; whereas the test results will be positive (D^+) for 99% of those individuals afflicted with the condition (H^+), test results will be negative (D^-) for 99% of those individuals not afflicted with the condition (H^-). We may write this as $p(D^+|H^+) = 0.99$ and $p(D^+|H^-) = 0.01$. In other words, test D diagnoses 99% of individuals correctly, whether they are afflicted with H or not.

Unfortunately, Mr Doe’s test results are positive. Applying Fisherian logic leads us to conclude that Mr Doe is afflicted with H . After all, the probability that Mr Doe would test positive under the hypothesis that he is not afflicted with H is quite low: $p(D^+|H^-) = 0.01$.

Mr Doe’s outlook is not as bleak, however, when considered from a Bayesian perspective. The Bayesian perspective considers, besides the test results, another piece of information; if you will recall, medical condition H is rare. Let us assume that a proportion of only 1/10,000 of the population is afflicted with H . We may write this as $p(H^+) = 0.0001$. $p(H^+)$ is known as the “prior probability” of H^+ . That is, prior to learning of Mr Doe’s test results, our best estimate for the probability that Mr Doe was afflicted with H would have been 0.0001. In the Bayesian framework, the positive test result is considered to be merely a second piece of evidence that is used to adjust our prior probability, now also taking into account Mr Doe’s positive test result, to derive the posterior probability ($H^+|D^+$).

According to Bayes’ Theorem

$$\begin{aligned} p(H^+|D^+) &= \frac{p(H^+)p(D^+|H^+)}{p(H^+)p(D^+|H^+) + p(H^-)p(D^+|H^-)} \\ &= \frac{0.0001 \cdot 0.99}{0.0001 \cdot 0.99 + 0.9999 \cdot 0.01} \\ &= \frac{0.000099}{0.000099 + 0.009999} \\ &\approx 0.0098 \end{aligned}$$

In other words, despite Mr Doe's positive test result, the odds are still strongly in favor of Mr Doe not being afflicted with H .

The obtained value for the posterior probability flies in the face of common sense. Indeed, students introduced to Bayesian reasoning by way of the above example often suspect something akin to sleight of hand, even if these students agree with every intermediate step performed. MDs, somewhat disconcertingly, do not do well either when it comes to Bayesian reasoning (e.g., [Hoffrage and Gigerenzer, 1998](#)).

4.3.3.2.2 BAYES' THEOREM APPLIED TO THE LIKELIHOOD $L(a, b | \mathbf{y})$

We might apply Bayes' Theorem to derive the posterior probability density function on our values for a and b . The situation is a little different here, since the likelihood is a function of a and b , which are continuous variables. In practice, however, we discretize our likelihood function. The appropriate formulation of Bayes' Theorem in this case becomes

$$p(a, b | \mathbf{y}) = \frac{L(a, b | \mathbf{y})p(a, b)}{\sum_a \sum_b L(a, b | \mathbf{y})p(a, b)} \quad (4.18)$$

where $L(a, b | \mathbf{y})$ is our likelihood function, as calculated in [Section 4.3.3.1](#), and $p(a, b)$ is the prior distribution. The resulting posterior distribution $p(a, b | \mathbf{y})$ is a probability density function. That is, it allows one to determine the probability that the values of a and b lie within a specified range of values.

What should we use as our prior distribution? The prior distribution should, according to the Bayesian framework, reflect our prior beliefs regarding the values of the threshold and slope of our PF. We might perhaps base our prior beliefs on research preceding ours. We might also base our prior beliefs on informal pilot experiments. Defining a prior is, of course, somewhat of a subjective exercise. For this reason, and as one might imagine, the Bayesian approach is not without its critics.

Before we will argue that the Bayesian approach does not need to be as subjective an exercise as one may have concluded from the above, let us first illustrate [Eqn \(4.18\)](#) by example. Let us imagine that, prior to performing our experiment of which [Figure 4.12](#) shows the likelihood function, we formed beliefs regarding the values of the threshold and slope. Perhaps we did so by considering existing literature on similar experiments, or perhaps we did so based on our own informal pilot experiments. Either way, let us imagine that we judge our prior beliefs to be well-described by the two-dimensional Gaussian

$$p(a, b) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(\log a)^2 + (\log b)^2}{2\sigma^2}\right)$$

with $\sigma = 0.5$. This prior distribution is illustrated in [Figure 4.14](#). Also shown in [Figure 4.14](#) is our likelihood function again as well as the posterior distribution derived by [Eqn \(4.18\)](#). It is clear that the posterior distribution is determined primarily by our choice of the prior and bears little resemblance to the likelihood function that was derived from our experimental data. We should keep in mind, though, that our example is not very typical for two reasons. First, our prior beliefs are quite specific. For example, our prior indicates a belief that the probability that $\log a$ has a value in the interval $(-0.5, 0.5)$ is near unity. Thus, apparently

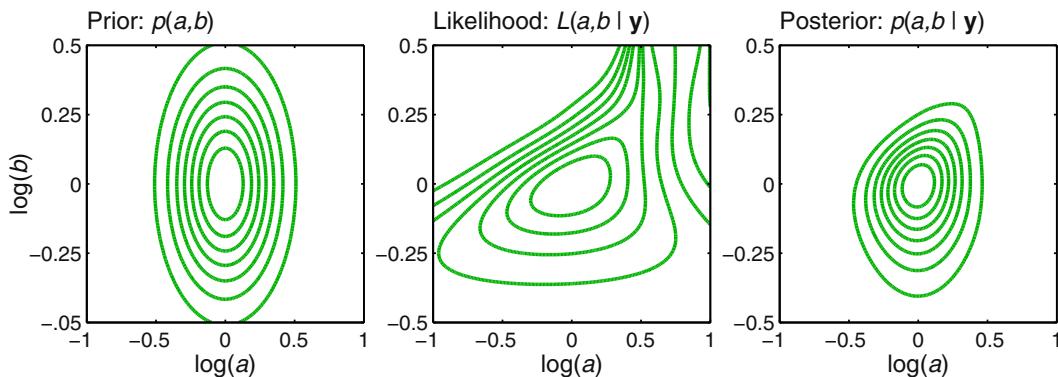


FIGURE 4.14 Contour plots of the prior distribution (left), the Likelihood function (middle), and the posterior distribution (right). The prior distribution reflects the researcher’s beliefs regarding the value of the threshold and slope parameter before the results of the experiment are taken into account, the Likelihood function is based on the results of the experiment only, and the posterior distribution combines the prior and the Likelihood according to Eqn (4.18).

we already knew a considerable amount about the values of α and β before we started our experiment. Second, our experiment was based on a very small number of trials ($N = 20$). Thus, it is not surprising that the results of our very small experiment hardly changed our strong prior beliefs.

For illustrative purposes, let us briefly consider two more priors, each at one of the two extreme ends of specificity. First, let us assume that the exact values of α and β are known with certainty before we conduct our experiment. Thus, our prior will be the unit impulse function located at the known values of α and β . Providing additional evidence by our experimental results will not alter our beliefs, and indeed our posterior will also be the unit impulse function located at the known values of α and β , regardless of what our likelihood function might be. The prior at the other extreme of specificity is the uniform prior. The uniform prior does not favor any values of α and β over any others and is thus consistent with a complete lack of knowledge or belief of what the values of α and β might be before we start our experiment. For this reason, some refer to the uniform prior as the “prior of ignorance.” If our prior is the uniform prior, our posterior distribution will be proportional to our likelihood function, and our parameter estimates will be determined entirely by the results of our experiment.

In practice it is difficult, if not impossible, to derive the continuous likelihood function analytically. Instead, we approximate the likelihood function across a discretized parameter space, which is necessarily of finite extent. Thus, given that we can only consider a limited extent of the parameter space, a strictly uniform prior is not possible. The best we can do is to create a rectangular prior which, in essence, assigns a prior probability of 0 to values of α and β that lie outside our limited parameter space, but which, within the considered parameter space, favor no values of α and β over other values.

We are finally ready to discuss how to derive parameter estimates based on the posterior distribution, and we will do so by example. Figure 4.15 presents the posterior distribution based on the 2AFC experiment we fitted before using a maximum likelihood criterion (Section 4.3.3.1). The experiment consisted of 700 trials, 100 trials at each of seven stimulus

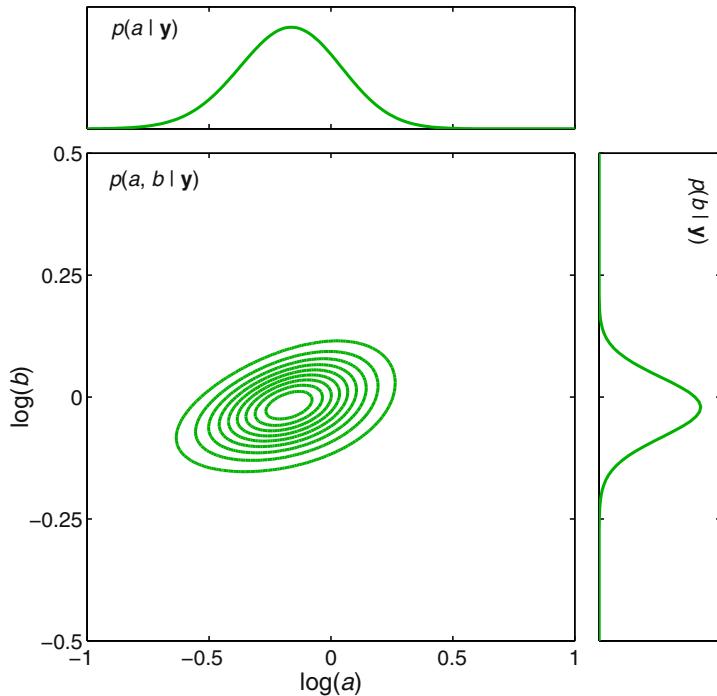


FIGURE 4.15 The posterior distribution.

intensities x , which were equally spaced on a logarithmic scale between $\log(x) = -3$ and $\log(x) = 3$. The number of correct responses (out of the 100 trials) for each of the seven stimulus levels was 55, 55, 66, 75, 91, 94, and 97, respectively. The prior used here was the uniform prior limited to $a \in [-1, 1]$, $b \in [-0.5, 0.5]$. We first calculated the likelihood function across the (discretized) parameter range defined by the prior (see Section 4.3.3.1). We used a guess rate (γ) equal to 0.5 and a lapse rate (λ) equal to 0. Since our prior is uniform and the posterior distribution is, by definition, a probability density function, our calculations simplify to

$$p(a, b | \mathbf{y}) = \frac{L(a, b | \mathbf{y})}{\sum_a \sum_b L(a, b | \mathbf{y})} \quad (4.19)$$

That is, our posterior distribution is simply our likelihood function rescaled such that $\sum_a \sum_b p(a, b | \mathbf{y}) = 1$, which is a quality of any probability density function.

Also shown in Figure 4.15 are the marginal probability densities across a and b individually, which are derived from $p(a, b | \mathbf{y})$ as follows

$$p(a | \mathbf{y}) = \sum_b p(a, b | \mathbf{y}) \quad (4.20)$$

$$p(b | \mathbf{y}) = \sum_a p(a, b | \mathbf{y}) \quad (4.21)$$

Our Bayesian estimator of $\log \alpha$ is the expected value of $\log \alpha$. That is

$$\log \hat{\alpha} = E(\log \alpha) = \sum_a \log a \ p(\log a | \mathbf{y}) \quad (4.22)$$

Similarly,

$$\log \hat{\beta} = E(\log b) = \sum_b \log b \ p(\log b | \mathbf{y}) \quad (4.23)$$

In this example, $\log \hat{\alpha} = -0.1715$ and $\log \hat{\beta} = -0.0225$.

We may note from [Figure 4.15](#) that the parameter space included in the prior was such that it excluded only parameter values associated with extremely small likelihoods (remember that, using a uniform prior, the likelihood function is proportional to the posterior distribution, and thus [Figure 4.15](#) may also be regarded as a contour plot of the likelihood function). As such, our exact choice for the range of values for $\log a$ and $\log b$ to be included in the prior would have a negligible effect on our final parameter estimates. Had we instead limited our prior to, for example, $a \in [-0.5, 0.5]$, $b \in [-0.25, 0.25]$ our likelihood function would have “run off the edge” of our prior (specifically the edge $\log a = -0.5$), and this would have affected our parameter estimates significantly. However, whenever we utilize a uniform prior, which encompasses all but the extremely small likelihoods, the contribution of our subjective prior to our parameter estimates will be negligible.

4.3.3.2.3 ERROR ESTIMATION

The posterior distribution is a probability density function across the parameter space. As such, we may use it to derive probabilities of either parameter having a value within any range of values. For example, we may derive the probability that our (log-transformed) threshold parameter has a value between -0.25 and 0 as

$$p(-0.25 \leq \log \alpha \leq 0) = \sum_{\log a \in [-0.25, 0]} \sum_{\log b} p(\log a, \log b | \mathbf{y}) = 0.4581$$

Due to the discretization of the parameter space, this value will only be approximate. The finer the grid of our parameter space, the more accurate the approximation will be. We may derive the standard errors of estimate of our parameter estimates as the relevant marginal standard deviation of the posterior distributions. For example, the standard error of the threshold estimate is given by

$$SE_{\log \alpha} = \sqrt{\sum_a \sum_b (\log a - \hat{\alpha})^2 p(\log a, \log b | \mathbf{y})} \quad (4.24)$$

The standard error for the slope is determined in an analogous manner. The SE for the estimate of log threshold determined using [Eqn \(4.24\)](#) equals 0.2106 ; that for log slope equals 0.0629 . Note that the standard error on log threshold obtained here corresponds closely to that obtained from bootstrap analysis under the maximum likelihood framework ([Section 4.3.3.1.3](#)). There we estimated SE on log threshold as 0.2121 and 0.2059 using a parametric and a nonparametric bootstrap, respectively. The comparison of SE on slope obtained here (0.0629) to that obtained using a parametric bootstrap analysis (0.0636) is also quite close.

Note that standard errors calculated in this manner are derived from the posterior distribution and as such will be affected not only by the data collected but also by the prior distribution. [Box 4.8](#) outlines how to derive parameter estimates and their standard errors using a Bayesian approach in the Palamedes toolbox.

BOX 4.8

FITTING PSYCHOMETRIC FUNCTIONS USING A BAYESIAN CRITERION IN PALAMEDES

The Palamedes function that can be used to fit a PF using a Bayesian criterion is `PAL_PFBA_Fit`. We will use this function here to fit the same data we fitted in [Box 4.2](#), but this time using a Bayesian criterion. As in [Box 4.2](#) we must first define the stimulus intensities used, the number of trials that were used at each of these levels, and the number of correct responses that were observed at each of the stimulus intensities:

```
>>StimLevels = [.01 .03 .05 .07 .09 .11];
>>NumPos = [59 53 68 83 92 99];
>>OutOfNum = [100 100 100 100 100 100];
```

We also need to specify the parameter space across which the posterior will be computed. It is important that the entire mass of the posterior distribution is effectively contained within the parameter space specified. In case the posterior distribution “runs off” the edges of the parameter space, this will affect the parameter estimates. Thus we will throw a wide net for the free parameters (threshold and slope) while we specify a single value for the fixed parameters (guess and lapse rate):

```
>>grid.alpha = linspace(0,.11,100);
>>grid.beta = linspace(0.3,100); %log-transformed values for beta
>>grid.gamma = 0.5;
>>grid.lambda = 0.02;
```

Note that the values for the slope are defined as log-transformed values for β . In other words, the smallest value for β contained in the parameter space is 1 (10^0), the largest value is 1000 (10^3), and the values in between are logarithmically scaled. We also assign the Logistic function to the variable `PF` as an inline function:

```
>>PF = @PAL_Logistic;
```

We are now ready to call function `PAL_PFBA_Fit`:

```
>>[paramsValues posterior] = PAL_PFBA_Fit(StimLevels, NumPos, ...
OutOfNum, grid, PF);
```

BOX 4.8 (*cont'd*)

The array `paramsValues` has two rows and four columns. The four columns correspond to the threshold, slope, guess rate, and lapse rate, respectively. The first row gives the estimates for these parameters, and the second row gives their standard errors.

```
paramsValues =
0.0584 1.8500 0.5000 0.0200
0.0038 0.0848 0.0000 0.0000
```

Note that the estimate for the threshold parameter corresponds closely to that obtained in [Box 4.2](#), where a maximum likelihood criterion was used. The value for the estimate of the slope parameter is given on a log-transformed scale. Thus the best-fitting value for β on a linear scale equals $10^{1.85} = 70.79$, which also corresponds closely with that obtained in [Box 4.2](#) (71.01). The standard error for the threshold parameter equals 0.0038 (compare to the standard error obtained using a parametric bootstrap in [Box 4.3](#): 0.0040). The standard error for the log-transformed slope parameter equals 0.0848. We may compare this to the standard error obtained in [Box 4.3](#) by determining the slope values corresponding to one standard error above the estimate (i.e., $10^{(1.85+0.0848)} = 86.06$) and one standard error below the estimate (i.e., $10^{(1.85-0.0848)} = 58.24$). Using the bootstrap method in [Box 4.3](#) the values corresponding to one standard error above and one standard error below the estimate were 86.49 ($71.0135 + 15.4723$) and 55.54 ($71.0135 - 15.4723$), respectively.

It is good practice to inspect the posterior distribution to make sure that the mass of the posterior distribution was effectively contained within the parameter space used. The return argument `posterior` in the call contains the posterior distribution. In this case the posterior is two-dimensional and can be easily inspected using, for example, a contour plot:

```
>>contour(posterior)
```

This call produces [Figure B4.8.1](#). Note that the parameter space generously encompasses the posterior distribution. We might rerun the fit while using a parameter space that targets the posterior a bit better:

```
>>grid.alpha = linspace(.03,.08,100);
>>grid.beta = linspace(1.5,2.5,100); %log-transformed values for beta

>>[paramsValues posterior] = PAL_PFBA_Fit(StimLevels, NumPos, ...
    OutOfNum, grid, PF);
```

Note that this parameter space not only throws a net that is not only not as wide as that above but that also has a much finer grain than that above. Nevertheless, the parameter values and their SEs returned match those above to the degree of precision provided. This suggests that both parameter spaces were sufficiently wide to encompass the posterior and used a grain that was sufficiently fine such that the discretization of the parameter space did not play a significant role in the values obtained.

BOX 4.8 (cont'd)

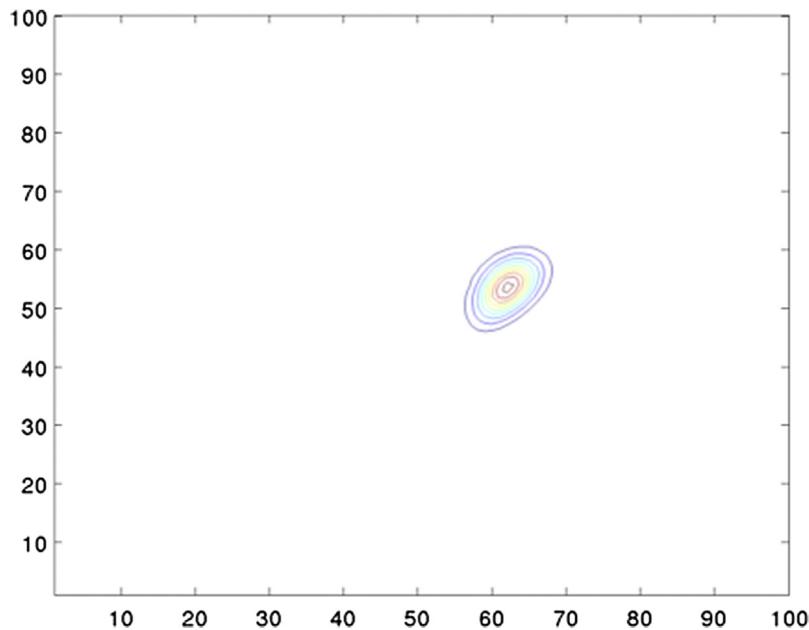


FIGURE B4.8.1 A basic contour plot of the posterior distribution. Note that threshold varies along the *y*-axis, while slope varies along the *x*-axis.

Above, we did not specify a prior distribution and the default uniform prior was used. The function `PAL_PFBA_Fit` allows the user to specify a custom prior. Let's say that based on experience a researcher wants to specify a prior on both the threshold and the slope of the PF. The prior on the threshold is specified as a Gaussian with mean 0.05 and standard deviation 0.03. The prior on the slope is also specified as a Gaussian but has mean 2 and standard deviation 1 (note that these values for the mean and standard deviation of the slope are again defined on log-transformed scale). Here, we create the prior by first defining the priors on the individual parameters, then using the MATLAB function `ndgrid` to specify the full prior as an array of the same size as the parameter space defined in the `grid` structure above:

```
>>prior.alpha = PAL_pdfNormal(grid.alpha,0.05,0.03);
>>prior.beta = PAL_pdfNormal(grid.beta,2,1);
>>prior.gamma = ones(size(grid.gamma));
>>prior.lambda = ones(size(grid.lambda));
>>prior.prior=ndgrid(prior.alpha,prior.beta,prior.gamma,prior.lambda);
>>prior.prior = prior.prior/sum(sum(sum(sum(prior.prior))));%prior must
%sum to 1
```

In order to pass the prior to the fitting routine we use the optional '`prior`' argument followed by the array containing the prior:

BOX 4.8 (*cont'd*)

```
>>[paramsValues posterior] = PAL_PFBA_Fit(StimLevels, ...
    NumPos, OutOfNum, grid, PF,'prior',prior.prior);

>>paramsValues =
0.0582      1.8488      0.5000      0.0200
0.0038      0.0846      0.0000      0.0000
```

The parameter estimates and their SEs are nearly identical to those obtained above since the prior distribution used was rather nonspecific (i.e., near-uniform) and was also nearly centered on the peak in the likelihood function.

Note that any combination of the four parameters can be estimated. Above we estimated the threshold and the slope while fixing the guess rate and the lapse rate. Let us also include the lapse rate as a free parameter:

```
>>grid.lambda = linspace(0,.06,100);

>>[paramsValues posterior] = PAL_PFBA_Fit(StimLevels, NumPos, ...
    OutOfNum, grid, PF);

>>paramsValues =
0.0586      1.8442      0.5000      0.0179
0.0040      0.0907          0      0.0140
```

Since the posterior is now three-dimensional it is not easily visualized in a figure. However, we can easily visualize the marginal posterior distributions for each of the three free parameters:

```
>>posteriorAlpha = squeeze(sum(sum(sum(posterior,4),3),2));
>>posteriorBeta = squeeze(sum(sum(sum(posterior,4),3),1));
>>posteriorLambda = squeeze(sum(sum(sum(posterior,3),2),1));

>>subplot(3,1,1)
>>plot(1:length(posteriorAlpha), posteriorAlpha);
>>subplot(3,1,2)
>>plot(1:length(posteriorBeta), posteriorBeta);
>>subplot(3,1,3)
>>plot(1:length(posteriorLambda), posteriorLambda);
```

Continued

BOX 4.8 (*cont'd*)

This code produces the rather unpolished plots shown in Figure B4.8.2.

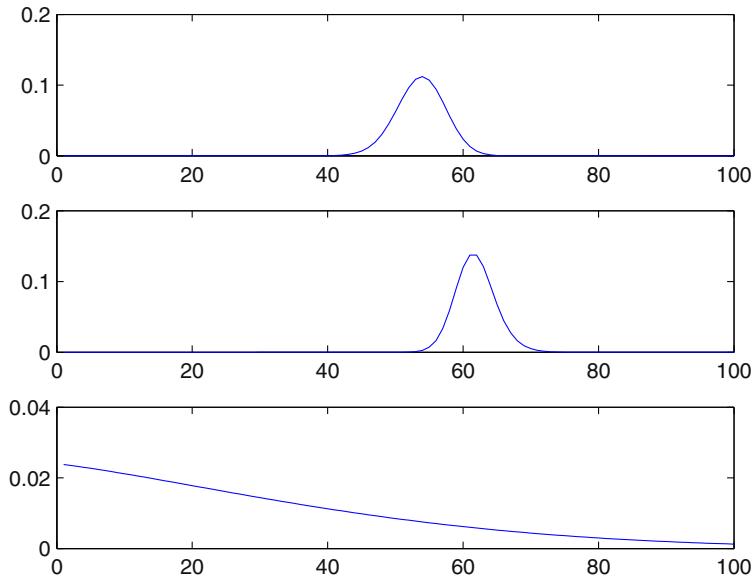


FIGURE B4.8.2 Crude plots of the marginal posterior distributions across values of alpha (top), beta (middle), and lambda (bottom).

FURTHER READING

[Swets \(1961\)](#) provides a very readable discussion of threshold theories. Chapter 6 of this text contains much more information on SDT. Maximum likelihood estimation is a standard technique and is discussed in any introductory statistical text. An excellent text on bootstrap methods is [Efron and Tibshirani \(1993\)](#). Our example of Mr Doe was adapted from many similar examples given in many places, among which is [Cohen \(1994\)](#), which discusses in a very readable manner some Hypothesis testing issues.

EXERCISES

1. A two-alternative forced-choice experiment is conducted in which the log stimulus levels are $-2, -1, 0, 1$, and 2 . 100 trials are presented at each of the stimulus levels. The observer responds correctly on respectively 48, 53, 55, 100, and 100 trials.

- a. Plot the results of this experiment.
 - b. By visual inspection, what do you estimate the 75% correct threshold to be?
 - c. Use PAL_PFML_Fit to fit these data.
 - d. Offer some suggestions to help improve on the design of the experiment.
2. As more trials are included in the computation of the likelihood associated with a parameter value, will this likelihood increase or decrease or does it depend on the outcome of the trial? Why?
 3. It is said sometimes that “extraordinary claims require extraordinary evidence” (Carl Sagan coined the phrase). Explain how this statement relates to Bayes’ Theorem.

References

- Cohen, J., 1994. The earth is round ($p < .05$). *Am. Psychol.* 49, 997–1003.
- van Driel, J., Knapen, T., van Es, D.M., Cohen, M.X., 2014. Interregional alpha-band synchrony supports temporal cross-modal integration. *Neuroimage* 101, 404–415.
- Efron, B., Tibshirani, R.J., 1993. An Introduction to the Bootstrap. Chapman & Hall/CRC, Boca Raton, FL.
- García-Pérez, M.A., Alcalá-Quintana, R., 2007. The transducer model for contrast detection and discrimination: formal relations, implications, and an empirical test. *Spat. Vis.* 20 (1–2), 5–43.
- Green, D.M., Swets, J.A., 1966. Signal Detection Theory and Psychophysics. Wiley, New York, NY.
- Hall, J.L., 1981. Hybrid adaptive procedure for estimations of psychometric functions. *J. Acoust. Soc. Am.* 69 (6), 1763–1769.
- Hays, W.L., 1994. Statistics. Wadsworth Group/Thomson Learning, Belmont, CA.
- Hoel, P.G., Port, S.C., Stone, C.J., 1971. Introduction to Statistical Theory. Houghton Mifflin Company, Boston, MA.
- Hoffrage, U., Gigerenzer, G., 1998. Using natural frequencies to improve diagnostic inferences. *Acad. Med.* 73, 538–540.
- Kuss, M., Jäkel, F., Wichmann, F.A., 2005. Bayesian inference for psychometric functions. *J. Vis.* 5, 478–492.
- Nachmias, J., 1981. On the psychometric function for contrast detection. *Vision Res.* 21, 215–223.
- Nelder, J.A., Mead, R., 1965. A simplex method for function minimization. *Comput. J.* 7, 308–313.
- Prins, N., 2010. Testing hypotheses regarding psychometric functions: robustness to violations of assumptions (abstract). *J. Vis.* 10 (7), 1384.
- Prins, N., 2012. The psychometric function: the lapse rate revisited. *J. Vis.* 12 (6), 25, 1–16.
- Prins, N., Kingdom, F.A.A., 2009. Palamedes: Matlab routines for analyzing psychophysical data. <http://www.palamedestoolbox.org>.
- Quick, R.F., 1974. A vector-magnitude model of contrast detection. *Kybernetik* 16, 65–67.
- Swanson, W.H., Birch, E.E., 1992. Extracting thresholds from noisy psychophysical data. *Percept. Psychophys.* 51 (5), 409–422.
- Swets, J.A., 1961. Is there a sensory threshold? *Science* 134, 168–177.
- Treutwein, B., Strasburger, H., 1999. Fitting the psychometric function. *Percept. Psychophys.* 61 (1), 87–106.
- Wichmann, F.A., Hill, N.J., 2001. The psychometric function: I. Fitting, sampling, and goodness of fit. *Percept. Psychophys.* 63, 1293–1313.

Adaptive Methods*

Frederick A.A. Kingdom¹, Nicolaas Prins²

¹McGill University, Montreal, Quebec, Canada; ²University of Mississippi, Oxford, MS, USA

OUTLINE

5.1 Introduction	119	5.3.2 Quest	132
5.2 Up/Down Methods	120	5.3.3 Termination Criteria and Threshold Estimate	133
5.2.1 Up/Down Method	120	5.3.4 Some Practical Tips	133
5.2.2 Transformed Up/Down Method	122		
5.2.3 Weighted Up/Down Method	122	5.4 The Psi Method and Variations	137
5.2.4 Transformed and Weighted Up/Down Method	123	5.4.1 The Psi Method	137
5.2.5 Termination Criteria and the Threshold Estimate	124	5.4.2 Termination Criteria and the Threshold and Slope Estimates	141
5.2.6 Some Practical Tips	129	5.4.3 Some Practical Tips	144
5.3 “Running Fit” Methods: The Best PEST and Quest	131	5.4.4 Psi-Method Variations	145
5.3.1 The Best PEST	131	Exercises	147
		References	147

5.1 INTRODUCTION

Measuring performance on a psychophysical task can be a time-consuming and tedious exercise. The purpose of adaptive methods is to make life easier for both the observer and the experimenter by increasing the efficiency of the testing procedure. Efficiency increases as the amount of effort required to reach a particular level of precision in the estimate of a parameter, such as the threshold of a psychometric function (PF), decreases. Taylor and

*This chapter was primarily written by Nicolaas Prins.

[Creelman \(1967\)](#) proposed a quantification of efficiency in the form of what they called the “sweat factor,” which is symbolized as K and is calculated as the number of trials multiplied by the variance of the parameter estimate (the variance of the parameter estimate is, of course, simply the standard error squared). Adaptive methods aim to increase efficiency by presenting stimuli at stimulus levels where one might expect to gain the most information about the parameter (or parameters) of interest. Adaptive methods are so termed because they adjust the stimulus level to be used on each trial based on the responses to previous trials.

Many specific adaptive methods have been proposed, and we could not possibly discuss them all here. However, adaptive methods can be grouped roughly into three major categories. This chapter discusses all three categories in turn. [Section 5.2](#) discusses what are commonly referred to as up/down methods. The basic idea behind up/down methods is straightforward. If the observer responds incorrectly to a trial, the stimulus intensity is increased on the next trial, whereas if the observer responds correctly on a trial (or a short series of consecutive trials), the stimulus intensity is decreased on the next trial. In such a procedure, the stimulus level will tend toward a specific proportion correct and oscillate around it once it is reached. While up/down methods work well if one is interested only in the value of the PF’s threshold, they provide little information regarding the slope of the PF.

[Section 5.3](#) discusses adaptive methods that perform a “running fit” on the data. That is, after every response a PF is fit to the responses of all preceding trials. The stimulus intensity to be used on the next trial is that which corresponds to the best estimate of the PF’s threshold, based on all previous trials. As was the case with up/down methods, running fit methods also provide information only about thresholds, not slopes of the PF.

In [Section 5.4](#) we will discuss the “psi method” and some variations on it. The psi method combines ideas from several adaptive methods proposed earlier. The psi method selects stimulus intensities on every trial that maximize the efficiency with which not only the threshold but also the slope of the PF is estimated. Various modifications of the psi method have been proposed and we will discuss some of these. The psi method is arguably the most sophisticated of the adaptive methods.

5.2 UP/DOWN METHODS

5.2.1 Up/Down Method

The up/down method was developed by [Dixon and Mood \(1948\)](#). We will explain the logic using the same example that Dixon and Mood used, which is that of determining the sensitivity of explosive mixtures to shock. Apparently, it was common practice to do this by dropping weights from different heights on specimens of explosive mixtures and noting whether an explosion resulted. The idea is that there will be a critical height which, if exceeded, will result in a mixture exploding, whereas below this height it will not explode. Let us say we drop a weight from a height of 20 feet and no explosion occurs. We now know that the critical height is greater than 20 feet. We could investigate further by increasing the height in steps of, say, one foot. We drop the weight from 21 feet... nothing, 22 feet... still nothing, 23 feet... Kaboom! We now know that the critical height has a value between 22 and 23 feet.

In reality, things are a little bit more complicated, of course, in that no two explosive mixtures are identical, and no two drops and consequent impacts of a weight are identical, either. We are no experts on explosive mixtures, but we imagine that other factors also play a role.

Thus, it would be more appropriate to say that for every drop height there is some probability that it will cause an explosive mixture to explode; the greater the height, the higher the probability that the mixture will explode. We might define the critical height as that height at which a mixture has a probability of, say, 50% of exploding. You will have realized the similarity between the problem of determining such an “explosion threshold” and that of determining, say, a detection threshold in the context of a sensory experiment.

Keeping the above in mind, the most we can conclude for certain from the above experiment is that the probability that a mixture will explode at 23 feet has a value greater than 0, and the probability that it will explode at 22 feet has a value less than 1. In order to get a better idea of what the value of the “explosion threshold” is, we should get more data. From what we have done so far, it seems reasonable to assume that the explosion threshold has a value somewhere around 22 or 23 feet. It would be silly to start dropping weights from a height of 150 feet or one foot at this point. The former is almost certain to result in an explosion, and the latter is almost certain not to result in an explosion.

Dixon and Mood (1948) suggest a very simple rule to decide which height we should drop a weight from on any trial. The rule simply states that if an explosion occurs on a trial, we should decrease the drop height on the next trial. If an explosion does not occur on a trial, we should increase the height on the next trial. In other words, our decision regarding the height to use on any trial is determined by what happened on the previous trial, and for this reason this method is termed an adaptive method. Figure 5.1 shows the results of a simulated experiment that continues the series started above, following the simple up/down rule. Any trials that resulted in an explosion are indicated by the star-shaped, filled symbols, and trials that did not are indicated by the open circles. The height corresponding to the 50% threshold that was used in the simulations is indicated in the figure by the broken line. We will discuss how to derive an explosion threshold estimate from these data later (Section 5.2.5), but for now we note that almost all trials (16 of 17 trials, or 94%) where the drop height was 23 feet resulted in an explosion, and that only seven of the 23 trials (30%) where the drop height was 22 feet resulted in an explosion. Thus, it seems reasonable to assume that the explosion threshold has a value somewhere between 22 and 23 feet.

Dixon and Mood’s up/down method targets the point on the PF at which either of two responses is equally likely to occur. This makes it particularly useful when one is interested

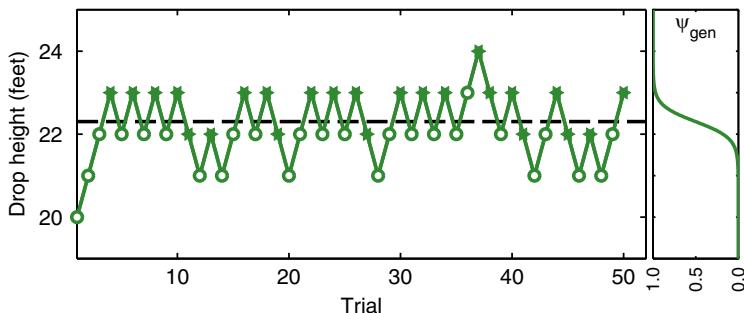


FIGURE 5.1 Simulated run of the up/down method of Dixon and Mood (1948). The figure follows the example described in the text. Weights are dropped from various heights (ordinate) on explosive mixtures. If an explosion occurs (filled symbols) the drop height is reduced by 1 foot on the next trial, and if no explosion occurs (open circular symbols), the drop height is increased by 1 foot. The targeted height (22.3 feet) is indicated by the broken line. Responses were generated by a Logistic function with $\alpha = 22.3$, $\beta = 5$, $\gamma = 0$, and $\lambda = 0$. The generating function (ψ_{gen}) is shown on the right.

in determining the point of subjective equality in an appearance-based task. For example, in the Muller–Lyer illusion (see Chapter 2) one could use the up/down method to find the ratio of line lengths at which the observer is equally likely to respond with either line when asked to indicate which of the lines appears to be longer.

5.2.2 Transformed Up/Down Method

As mentioned, Dixon and Mood's up/down method targets the stimulus intensity at which either of two possible responses is equally likely to occur. In many experimental situations this will be no good. In a 2AFC task, for example, 50% correct corresponds to chance performance. In such situations, we could use [Wetherill and Levitt's \(1965\)](#) "transformed" up/down method. In the transformed up/down method the decision to decrease stimulus intensity is based on a few preceding trials, rather than the very last single trial. For example, we could adopt a rule that increases stimulus intensity after every incorrect response, as before, but decreases stimulus intensity only after two consecutive correct responses have been observed since the last change in stimulus intensity. Such a rule is commonly referred to as a 1 up/2 down rule. Wetherill and Levitt make the argument that the 1 up/2 down rule targets 70.71% correct. Another commonly used rule is similar to the 1 up/2 down except that stimulus intensity is decreased only after three consecutive correct responses have been observed since the last change in stimulus intensity. This 1 up/3 down rule targets 79.37% correct. A simulated experimental run using the 1 up/2 down rule is shown in [Figure 5.2\(a\)](#). Correct responses are indicated by the filled symbols and incorrect responses are shown by the open symbols. Note that the 1 up/2 down rule came into effect only after the first incorrect response was observed. Before this point, a 1 up/1 down rule was employed. In the run shown, the simulated observer responded correctly on all of the first five trials. This was because the run started out at a stimulus intensity that was well above the targeted threshold and possibly a bit of luck. Either way, had we adopted the 1 up/2 down rule from the start, it would have likely taken many more trials to reach stimulus intensities around threshold levels. The strategy to adopt a 1 up/1 down rule until a first reversal of direction is needed was suggested by Wetherill and Levitt in order to avoid presenting many trials at intensities that are far above threshold at the start of the run.

5.2.3 Weighted Up/Down Method

Another possibility is to adopt a "weighted" up/down method ([Kaernbach, 1991](#)) in which a 1 up/1 down rule is used, but the steps up are not equal in size to the steps down. Kaernbach argues that the rule targets a probability correct equal to

$$\psi_{\text{target}} = \frac{\Delta^+}{\Delta^+ + \Delta^-} \quad (5.1a)$$

where Δ^+ and Δ^- are the sizes of the steps up and steps down, respectively, and ψ_{target} is the targeted proportion correct. A little algebra reveals

$$\frac{\Delta^-}{\Delta^+} = \frac{1 - \psi_{\text{target}}}{\psi_{\text{target}}} \quad (5.1b)$$

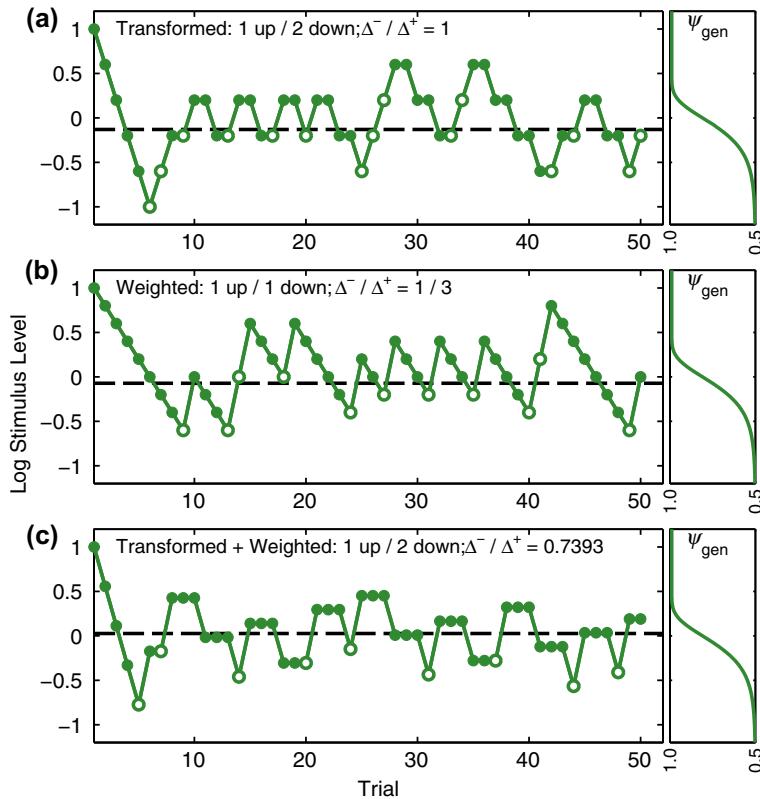


FIGURE 5.2 Examples of simulated staircases following a transformed up/down rule (a); a weighted up/down rule (b); and a transformed and weighted up/down rule (c). Correct responses are indicated by the filled symbols, and incorrect responses are indicated by open symbols. Stimulus levels corresponding to the targeted percent correct values are indicated by the broken lines (note that the different procedures target different performance levels). In all example runs the responses were generated by a Gumbel function with $\alpha = 0$, $\beta = 2$, $\gamma = 0.5$, and $\lambda = 0.01$. The generating PF (ψ_{gen}) is shown to the right of each graph. Δ^+ : size of step up; Δ^- : size of step down (see Section 5.2.3).

Let's say you wish to target 75% correct performance. Using a value of 0.75 for ψ_{target} in Eqn (5.1b) gives

$$\frac{\Delta^-}{\Delta^+} = \frac{1 - 0.75}{0.75} = \frac{1}{3}$$

Figure 5.2(b) shows a simulated run of 50 trials using a 1 up/1 down rule and a ratio of stepsizes equal to 1/3.

5.2.4 Transformed and Weighted Up/Down Method

In the “transformed and weighted up/down method” (García-Pérez, 1998), steps up and steps down are, as in the weighted method, not of equal size. Also, stimulus intensity is

decreased only after a set number of consecutive incorrect responses, as in the transformed up/down method. The proportion correct targeted by the transformed and weighted up/down method is given as

$$\psi_{\text{target}} = \left(\frac{\Delta^+}{\Delta^+ + \Delta^-} \right)^{\frac{1}{D}} \quad (5.2)$$

where Δ^+ , Δ^- , and ψ_{target} are as before, and D is the number of consecutive correct responses after which a step down is to be made. Dixon and Mood's up/down method, the weighted up/down method, and the transformed up/down method are, of course, also covered by Eqn (5.2), as they are all particular cases of the transformed and weighted up/down method. Figure 5.2(c) shows an example run of 50 trials using a transformed and weighted up/down method. Box 5.1 explains how to use Palamedes (Prins and Kingdom, 2009) to set up and use an up/down adaptive testing run.

5.2.5 Termination Criteria and the Threshold Estimate

Several methods are in use to estimate a threshold after a run of trials has been completed. Most commonly, researchers will terminate a run after a specific number of reversals of direction have occurred (García-Pérez, 1998). The threshold estimate is consequently calculated as the average stimulus intensity across the last few trials on which a reversal occurred in the run. For example, a run may be terminated after 10 reversals have taken place, and the threshold estimate is calculated as the average stimulus intensity across the last eight trials on which a reversal occurred. Less frequently, the run is terminated after a specified number of trials have occurred, and the threshold is calculated as the average stimulus intensity across the last of so many trials.

Yet another strategy is to adopt what Hall (1981) has termed a "hybrid adaptive procedure" in which an up/down method is used to select stimulus intensities, after which a threshold estimate is derived by fitting a PF to all the data collected using, for example, a maximum likelihood criterion (see Chapter 4). This strategy has the obvious disadvantage that fitting the data with a PF requires us to assume a shape of the PF as well as values of some of its parameters (such as the guess rate, lapse rate, and perhaps the slope). The up/down methods themselves are nonparametric; they do not assume a particular shape of the underlying PF, other than that it is a monotonic function of stimulus intensity.

After we have determined the thresholds for the individual runs we face another decision. Typically, we would want to use more than one run, and thus we end up with several threshold estimates that should be combined into a single estimate. The obvious, and indeed most common, solution is to average the individual threshold estimates. The standard deviation of threshold estimates may serve the function of standard error of estimate. The hybrid adaptive procedure allows us to combine trials across the different runs before we fit them with a single PF. When we do use the hybrid procedure, we should keep in mind that by the nature of the up/down procedure trials will be concentrated around a single point on the PF. Such data do not lend themselves well to the estimation of the slope of the PF. Combining all data and fitting them with a single PF will also allow us to determine the reliability of our estimate by performing a bootstrap analysis or by using the standard deviation of the parameter's posterior distribution (see Chapter 4).

BOX 5.1

UP/DOWN METHODS IN PALAMEDES

The core function associated with the up/down methods in Palamedes is `PAL_AMUD_updateUD`. The function `PAL_AMUD_updateUD` is called after every trial with two arguments: a structure which we call `UD` (although you may give it a different name) and a scalar which indicates whether the observer gave a correct (1) or an incorrect (0) response on the trial. The structure `UD` stores things such as the stimulus intensity used on each trial, the response of the observer, etc., and this information is updated after every trial when the function `PAL_AMUD_updateUD` is called. The `UD` structure also stores such things as the up/down rule to be used, stepsizes to be used, the stimulus value to be used on the first trial, etc.

Before trials can begin, the function `PAL_AMUD_setupUD` must be called in order to create the `UD` structure and initialize some of its entries. Let's first create the `UD` structure using the default values and inspect it:

```
>>UD = PAL_AMUD_setupUD;
>>UD
UD =

```

up:	1
down:	3
stepSizeUp:	0.0100
stepSizeDown:	0.0100
stopCriterion:	'reversals'
stopRule:	32
startValue:	0
xMax:	Inf
xMin:	-Inf
truncate:	'yes'
response:	[]
stop:	0
u:	0
d:	0
direction:	[]
reversal:	0
xCurrent:	0
x:	0
xStaircase:	[]

The `up` and `down` fields indicate which up/down rule should be used. By default the values are set to 1 and 3, respectively, such that a 1 up/3 down rule will be used. Of course, the default values can be changed to suit your needs. You can pass optional arguments to `PAL_AMUD_setupUD` to make such changes. These arguments come in pairs. The first argument of each pair indicates which option should be changed, and the second argument of the pair indicates the new value. For example, let's say you would like to use a 1 up/2 down rule instead of the default 1 up/3 down rule. You would change the `UD` field down to 2 by giving the following command:

```
>>UD = PAL_AMUD_setupUD(UD, 'down', 2);
```

BOX 5.1 (*cont'd*)

In this call we also passed the existing `UD` structure as the first argument. When an existing `UD` structure is passed to `PAL_AMUD_setupUD` as the first argument, the existing structure `UD` will be updated according to the additional arguments. It is also possible to create an entirely new structure `UD` using optional arguments to override the default values. For example, the call

```
>>UD = PAL_AMUD_setupUD('down', 2);
```

creates a new structure `UD` but with the field `down` set to 2 instead of the default value 3. It is important to realize that when the latter syntax is used, a new structure is created and any previous changes made to `UD` will be undone. Before we worry about the other options, let us demonstrate how to use the function `PAL_AMUD_updateUD` to update the stimulus level on each trial according to the chosen up/down rule. As we go along, we will explain some of the other options.

Imagine we are measuring a contrast threshold for an observer and we would like to use a 1 up/3 down rule. We vary contrast amplitudes on a logarithmic scale and wish to use stepsizes equal to 0.05 log units for steps up as well as steps down. We first create a new `UD` structure with the above options.

```
>>UD = PAL_AMUD_setupUD('up', 1, 'down', 3, 'stepsizeup', 0.05, ...
    'stepsizedown', 0.05);
```

We also wish the procedure to terminate after, say, 50 trials have occurred. We can change the relevant settings in the existing `UD` structure by calling the function again and passing it the existing structure as the first argument followed by the other settings we wish to change. We set the value of `stopCriterion` to the string 'trials' to indicate that we wish to terminate the run after a set number of trials (the default setting is to terminate after a set number of reversals of direction have occurred). Because we wish the run to terminate after 50 trials, we set the value of `stopRule` to 50.

```
>>UD = PAL_AMUD_setupUD(UD, 'stopcriterion', 'trials', ...
    'stoprule', 50);
```

We should also indicate what contrast amplitude should be used on the first trial (or accept the default value of 0):

```
>>UD = PAL_AMUD_setupUD(UD, 'startvalue', 0.3);
```

Note that all changes to the default entries could have also been made in a single call and that the case of the string arguments will be ignored.

We are now ready to present our first stimulus to our observer. The contrast amplitude that we should use on any trial is given in `UD`'s field `xCurrent`. Currently, the value of `UD.xCurrent` is 0.3. This is, after all, the value that we submitted as '`startvalue`'. After we present our stimulus at the amplitude given in `UD.xCurrent` we collect a response from the observer. We create a variable `response` and assign it the value 1 in case the response was correct or the value 0 in case the response was incorrect.

BOX 5.1 (*cont'd*)

```
>>response = 1;
```

Now we call the function `PAL_AMUD_updateUD`, passing it the structure `UD` and the value of `response`

```
>>UD = PAL_AMUD_updateUD(UD, response);
```

`PAL_AMUD_updateUD` makes the appropriate changes to `UD` and returns the updated version. You will have noted that the above call assigns the returned structure to `UD`. In effect, the updated `UD` will replace the old `UD`. The value of `UD.xCurrent` has now been updated according to the staircase rules and should be used as the contrast amplitude to be used on the next trial.

```
>>UD.xCurrent
ans =
0.2500
```

The process then repeats itself: we present the next trial at the new value of `UD.xCurrent`, collect a response from the observer, and call `PAL_AMUD_updateUD` again. When the criterion number of trials has been reached, the `stop` field of the `UD` structure will be set to 1, and this will be our signal to exit the testing procedure. The program that controls your experiment would contain a trial loop such as this

```
while ~UD.stop
    %Present trial here at stimulus intensity UD.xCurrent
    %and collect response (1: correct [more generally: too high],
    %0: incorrect)
    UD = PAL_AMUD_updateUD(UD, response); %update UD structure
end
```

The `UD` structure maintains a record of stimulus intensities and responses for all trials. The stimulus amplitudes of all trials are stored in `UD.x`, and the responses are stored in `UD.response`. The file `PAL_AMUD_Demo` demonstrates how to use the functions in the Palamedes toolbox to implement the 1 up/3 down staircase discussed here. It simulates responses of a hypothetical observer who acts according to a Gumbel function. The program will produce a plot such as those in [Figure 5.2](#). A correct response to a trial is indicated by a filled circle and an incorrect response by an open circle. Note again that at the start of the series, the procedure decreases the stimulus intensity after every correct response. It is only after the first reversal of direction occurs that the 1 up/3 down rule goes into effect.

The field `UD.reversal` contains a 0 for each trial on which a reversal did not occur and the count of the reversals for trials on which a reversal did occur. For example, in the run shown in [Figure 5.2\(a\)](#), the first reversal took place due to an incorrect response on trial 6 and the second reversal took place following the second of two consecutive correct responses on trial 11. Thus, the field `UD.reversal` will start: [0 0 0 0 0 1 0 0 0 0 2].

Continued

BOX 5.1 (*cont'd*)

The function `PAL_AMUD_analyzeUD` will calculate the mean of either a specified number of reversals or a specified number of trials. By default, it will calculate the average of all but the first two reversal points. Usage of the function is as follows:

```
>>Mean = PAL_AMUD_analyzeUD(UD);
```

where `UD` is the result of a run of an up/down adaptive procedure. You may override the default and have the mean calculated across a specific number of reversals. For example, in order to calculate the mean across the last five reversals use

```
>>Mean = PAL_AMUD_analyzeUD(UD, 'reversals', 5);
```

You may also calculate the mean across the last of so many trials. For example, in order to calculate the mean across the last 10 trials use

```
>>Mean = PAL_AMUD_analyzeUD(UD, 'trials', 10);
```

Keep in mind that all data are stored in the `UD` structure, and you may use them as you please. For example, you could use `PAL_PFML_Fit` (Chapter 4) to fit a PF to the data using a maximum likelihood criterion. For example, to fit a threshold value to the data assuming the shape of a Logistic function, a slope of 20, a guess rate of 0.5, and a lapse rate of 0.01 use

```
>>searchGrid.alpha = [0.5:.01:.5];
>>searchGrid.beta = 20;
>>searchGrid.gamma = 0.5;
>>searchGrid.lambda = 0.01;
>>params = PAL_PFML_Fit(UD.x, UD.response, ones(1, ...
    length(UD.x)), searchGrid, [1 0 0 0], @PAL_Logistic);
```

Below we list all of the options for the Palamedes up/down routines that may be changed using the function `PAL_AMUD_setupUD` with an explanation as to their function. Default values are those shown in curly brackets {}.

`Up` positive integer scalar {1}

Number of consecutive incorrect responses after which stimulus intensity should be increased.

`Down` positive integer scalar {3}

Number of consecutive correct responses after which stimulus intensity should be decreased.

`stepSizeUp` positive scalar {0.01}

`Size of step up`

`stepSizeDown` positive scalar {0.01}

`Size of step down`

`stopCriterion` 'trials' | {'reversals'}

When set to 'trials', the staircase will terminate after the number of trials set in `stopRule`. When set to 'reversals', the staircase will terminate after the number of reversals set in `stopRule`.

BOX 5.1 (*cont'd*)

stopRule see stopCriterion {32}
 startValue scalar {0}
 Stimulus intensity to be used on first trial.
 xMax scalar {Inf}
 Maximum stimulus intensity to be assigned to UD.xCurrent.
 xMin scalar {-Inf}
 Minimum stimulus intensity to be assigned to UD.xCurrent.
 truncate {'yes'} | 'no'

When set to 'yes', the up/down rule will be applied to stimulus intensities as limited by xMax and xMin. When set to 'no', the up/down rule will be applied to stimulus intensities untruncated by xMax and xMin (but stimulus intensities assigned to UD.xCurrent will be truncated by xMax and xMin).

5.2.6 Some Practical Tips

Using a large number of simulated up/down staircases, García-Pérez (1998) has investigated the behavior of up/down staircases systematically. Somewhat surprisingly perhaps, the staircases converged reliably on the proportion correct given by Eqn (5.2) only when specific ratios of the up and down stepsizes (Δ^-/Δ^+) were used. These ratios are listed in Table 5.1. At other ratios, the proportion correct on which the staircases converged depended greatly on the ratio of the stepsize to the spread of the PF (Chapter 4 Section 4.3.2.8). For certain stepsize ratios and up/down rule combinations the staircases converged on proportions correct nowhere near those given by Eqn (5.2). Note that the run shown in Figure 5.2(c) uses the suggested stepsize ratio for the 1 up/3 down rule employed.

According to García-Pérez (1998), large stepsizes should be used, with steps up having a value between $\sigma/2$ and σ , where σ is the spread of the underlying PF using $\delta = 0.01$ (Chapter 4 Section 4.3.2.8). Of course, the spread of the underlying PF will not be known, but we can

TABLE 5.1 Ratios of down stepsize and up stepsize Δ^-/Δ^+ that will reliably converge on the targeted ψ values given by Eqn (5.2); these values are suggested by García-Pérez (1998) and are based on a large number of simulated runs.

Rule	Δ^-/Δ^+	Targeted ψ (%)
1 up/1 down	0.2845	77.85
1 up/2 down	0.5488	80.35
1 up/3 down	0.7393	83.15
1 up/4 down	0.8415	85.84

generate a rough estimate based on intuition or previous research. Large stepsizes produce reversals more quickly and allow for a faster return to stimulus intensities near the targeted threshold after, for example, a series of lucky responses. The use of large stepsizes also ensures that near-threshold levels are reached early in the run. As a result, we may determine the threshold by averaging stimulus intensities across all but the first few reversals.

Stepsizes should be defined in whatever metric appears appropriate. When we started our discussion of up/down methods we used the example of explosive mixtures. We defined the stepsize in terms of drop height (in feet). Consequently, the procedure used steps that were equal in terms of drop height in feet. Perhaps it would have made more sense to use stepsizes that corresponded to equal changes in the speed with which the weights hit the explosive mixture. If so, we should simply define our stepsizes in terms of speed at impact. In the context of psychophysical measurements, stepsizes should be defined in physical units that would correspond to linear units in the internal representation of the stimulus dimension. Our choice should thus be guided by what we believe the transducer function to be (Chapter 4 Section 4.2.3.3). Our choice will ordinarily be between defining stepsizes on a linear scale or on a logarithmic scale. If we wish steps to be constant on a logarithmic scale, we should define our stimulus intensities and stepsizes as such. The final threshold should be calculated as the arithmetic mean calculated across the reversal values in whatever scale our stepsizes were defined in. For example, if we defined our stimulus intensities and stepsizes on a logarithmic scale, we should calculate the arithmetic mean of the reversal values in logarithmic terms, which is equivalent to the geometric mean of the values on a linear scale.

Note that sometimes it may be necessary to set a minimum or maximum stimulus value that may be used in an up/down procedure. You should avoid this if you can, but sometimes you have no other choice. For example, physical limitations of experimental equipment may dictate an upper and/or lower limit on the stimulus intensities that can be used. When negative values for the stimulus intensity are nonsensical (for example in the case of brightness), we should set the minimum value to be used at 0. Another manner in which negative values can be avoided is to define stepsizes on a logarithmic scale.

An issue is raised when possible stimulus values are constrained to a range of values. Suppose you are measuring a contrast threshold. Your stimulus intensities and stepsizes are defined as Michelson contrast on a linear scale. You use a 1 up/1 down with $\Delta^+ = 0.1$ and $\Delta^- = 0.05$. Having defined your stimulus intensity in terms of Michelson contrast you set the minimum stimulus intensity to 0. As it happens your observer has had a few consecutive lucky responses and the stimulus intensity on trial t (let's call it x_t) equals 0. In case your observer responds correctly on trial t , the up/down rule states that stimulus intensity on trial $t + 1$ should be $x_{t+1} = x_t - \Delta^- = 0 - 0.05 = -0.05$. However, having defined the minimum stimulus intensity as 0, the stimulus intensity used will actually be set to 0.

Imagine the observer now produces an incorrect response on trial $t + 1$. Should we make our step up relative to what the intensity should have been on trial $t + 1$ ($x_{t+2} = -0.05 + \Delta^+ = 0.05$) or relative to what it actually was ($x_{t+2} = 0 + \Delta^+ = 0.1$)? Somewhat counterintuitively, perhaps, the former strategy has been shown to produce better results (García-Pérez, 1998).

By the very nature of the up/down procedures, strong trial-to-trial dependencies within a single staircase exist. Observers are very good at discovering rules such as the following: "A series of consecutive lucky guesses is followed by one or more trials on which I also feel like I

am guessing. This continues until I give a few incorrect responses.” Some evidence even suggests that humans can discover such simple contingency rules implicitly and begin to act accordingly before the rules are consciously known (Bechara et al., 1997). In order to avoid trial-to-trial dependencies and observer strategies which are based on trial-to-trial contingencies, it is a good idea to alternate trials randomly between a few interleaved up/down staircases.

5.3 “RUNNING FIT” METHODS: THE BEST PEST AND QUEST

The methods that we will describe here perform a running fit of the results. The idea was first proposed by Hall (1968) at a meeting of the Acoustical Society of America. After every trial a PF is fit to all the data collected so far. The fitted PF then serves to select a stimulus intensity for the upcoming trial. After each trial, the fit is updated based on the new response, and the process repeats itself.

5.3.1 The Best PEST

The first running fit method to be proposed in detail was the “best PEST” (Pentland, 1980). The best PEST assumes a specific form of the PF and estimates only the threshold parameter of the PF. Values for the other parameters (slope, guess rate, and lapse rate) need to be assumed. After each trial, the likelihood function (Chapter 4) is calculated based on the responses to all previous trials. The likelihood function is defined across a range of possible threshold values believed to include the observer’s threshold value. After each trial a value for the threshold parameter is estimated using a maximum likelihood criterion. The stimulus intensity to be used on the next trial corresponds to the threshold estimate determined from all previous trials.

A simulated example run of the best PEST in a 2AFC procedure is shown in Figure 5.3(a). As can be seen from the figure, a run controlled by the best PEST has some similarities to runs controlled by the up/down procedures discussed in Section 5.2. Foremost, the best PEST decreases stimulus intensity after a correct response and increases stimulus intensity after an incorrect response. Keep in mind, though, that in the case of the best PEST this is an emergent property; it is not a rule that is explicitly incorporated in the procedure. Unlike in the up/down procedures of Section 5.2, however, stepsizes in the best PEST are not of fixed size. Generally, stepsizes tend to decrease as the run proceeds. This makes sense when one considers that the relative contribution of each additional trial to the overall fit becomes smaller and smaller, given that the fit is based on all of the preceding trials.

Note that the first stepsize is exceptionally large. As a matter of fact, the size of the first step is bound only by the interval of stimulus values defined by the experimenter. The maximum likelihood estimate of the threshold after a single trial will always be positive infinity (if the response was incorrect) or negative infinity (if the response was correct), and thus the stimulus amplitude on the second trial will always correspond to the highest or lowest value of the interval across which the likelihood function is considered. In the example shown in the figure, the first trial resulted in a correct response and the second trial is presented at the lowest stimulus intensity in the interval across which the likelihood function

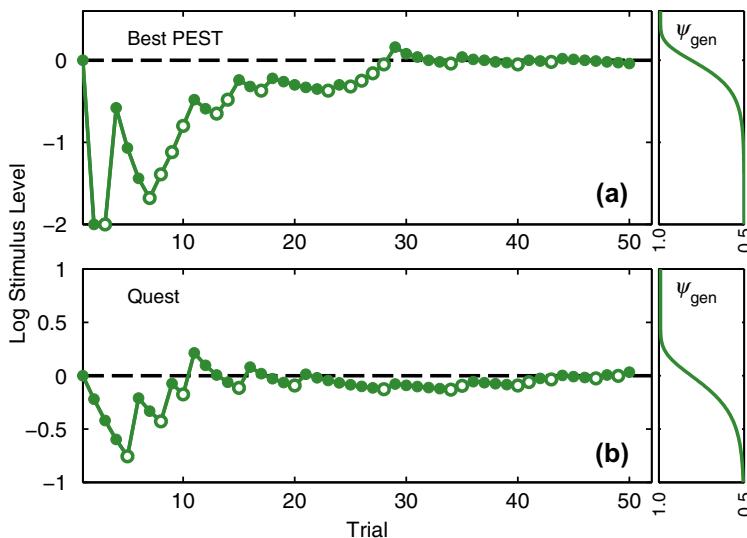


FIGURE 5.3 Examples of a simulated best PEST staircase (a) and a Quest staircase (b). Correct responses are indicated by the filled symbols, and incorrect responses are indicated by open symbols. Stimulus levels corresponding to the threshold of the generating PFs are indicated by the broken lines. In both example runs the responses were generated by a Gumbel function with $\alpha = 0$, $\beta = 2$, $\gamma = 0.5$, and $\lambda = 0.01$. The generating PF (ψ_{gen}) is shown to the right of each graph.

is calculated. The response on the second trial is also correct and, judging by the generating PF on the right of the figure, this should be considered mostly a result of luck. As a result, the third trial is also presented at the same extremely low stimulus intensity. The response to the third trial was incorrect, and as a result the fourth trial is presented at a much higher stimulus intensity, although still at a stimulus intensity where we would expect near-chance level performance. The following few trials once again appear to be the result of some luck. As a result, it takes a while for the best PEST to reach stimulus intensities near threshold in this particular run. It should be pointed out that this particular run was selected because it displayed this behavior. However, although it may not be typical, this behavior is certainly not uncommon using the best PEST.

5.3.2 Quest

Quest (Watson and Pelli, 1983) is essentially a Bayesian version of the best PEST. To refresh your memory, Bayes' Theorem (Chapter 4) can be used to combine the results of an experiment (in the form of the likelihood function) with preexisting knowledge or beliefs regarding the value of the threshold parameter (in the form of a prior probability distribution) to derive the posterior probability distribution across possible values of the threshold parameter. One may think of this procedure as the new data merely serving to adjust our preexisting knowledge or beliefs regarding the value of the threshold parameter. A best-fitting estimate of the threshold parameter is then derived from the posterior distribution using the method outlined in Chapter 4. Thus, before a Quest run can start, the researcher needs to postulate a

prior probability distribution that reflects the researcher’s belief about the value of the threshold. In [Figure 5.3\(b\)](#), we show an example Quest run in which the prior distribution was a Gaussian distribution with mean 0 and standard deviation 1. The simulated observer in the Quest run of [Figure 5.3](#) had identical properties to the simulated observer in the best PEST run in the figure. As can be seen by comparing the best PEST and Quest run, the prior has the effect of curbing the excessive stepsizes that were observed at the beginning of the best PEST run. The prior could thus be considered to act as somewhat of a guide to the selection of stimulus intensities. This is especially true at the onset of the run when stimulus selection is primarily determined by the prior. As data collection proceeds, the prior will start to play a smaller and smaller role relative to the contribution of the data collected.

In the example run of Quest shown in [Figure 5.3](#), the mode of the posterior distribution was used as the threshold estimate, as proposed by Watson and Pelli in the original Quest procedure. [King-Smith et al. \(1994\)](#) show that using the mean of the posterior distribution, rather than its mode, leads to more efficiently obtained parameter estimates that are also less biased. [Alcalá-Quintana and García-Pérez \(2004\)](#) further recommend the use of a uniform prior (i.e., one that favors no particular threshold value over another). Remember that when a uniform prior is used, the posterior distribution will be proportional to the likelihood function (see Chapter 4). Thus, when the prior is uniform and we use the mode of the posterior distribution as our estimate of the threshold, Quest is equivalent to the best PEST. [Box 5.2](#) explains how to use Palamedes to set up and use a running fit testing run.

5.3.3 Termination Criteria and Threshold Estimate

Most commonly, a session is terminated after a specific number of trials. Alternatively, one can terminate a session after a specific number of reversals have occurred. The threshold estimate is, of course, updated for every trial, and the final threshold estimate is simply the estimate that was derived following the final response. In case we have used a nonuniform prior, we may opt to ignore it in our final threshold estimate. Remember from Chapter 4 that the posterior distribution is proportional to the product of the prior distribution and the likelihood function. In other words, if we divide the prior out of the posterior distribution, the result is proportional to our likelihood function. Since choosing a prior is a bit of a subjective exercise, some researchers opt to derive the final threshold estimate from the (recovered) likelihood function. As with the up/down methods we may also use a hybrid approach in which we use a running fit method to guide stimulus selection, but we derive our final threshold estimate and its standard error after we combine trials across several sessions.

5.3.4 Some Practical Tips

The running fit methods described here are “parametric methods.” What this means is that they assume that our observer responds according to a specific form of PF with a specific value for its slope, guess rate, and lapse rate. We need to specify all these assumptions, and the method is optimal and accurate only insofar as these assumptions are true. This was not the case for the up/down methods of [Section 5.2](#). There, we do not have to assume anything about the shape of the PF (other than that it is monotonic). The running fit methods

BOX 5.2**RUNNING FIT METHODS IN PALAMEDES**

The routines in Palamedes that manage a running fit adaptive method are `PAL_AMRF_setupRF` and `PAL_AMRF_updateRF`. The general usage of these functions is analogous to the functions `PAL_AMUD_setupUD` and `PAL_AMUD_updateUD`, described in [Box 5.1](#). We first create a structure `RF` using `PAL_AMRF_setupRF`.

```
>>RF = PAL_AMRF_setupRF;
```

`RF` is a structure which is similar to the structure `UD` in [Box 5.1](#).

```
RF =
```

```
priorAlphaRange: [1x401 double]
    prior: [1x401 double]
    pdf: [1x401 double]
    mode: 0
    mean: 1.9082e-17
    sd: 1.1576
modeUniformPrior: []
meanUniformPrior: []
sdUniformPrior: []
response: []
stopCriterion: 'trials'
stopRule: 50
    stop: 0
    PF: @PAL_Gumbel
    beta: 2
    gamma: 0.5000
    lambda: 0.0200
    xMin: -Inf
    xMax: Inf
direction: []
reversal: 0
meanmode: 'mean'
xCurrent: 1.9082e-17
    x: []
xStaircase: 1.9082e-17
```

The value of the `mean` field is calculated as the expected value of the prior distribution and differs (very slightly) from 0 due to rounding error only. As a result, `xCurrent` and `xStaircase` also differ slightly from 0. Changing the settings to suit your needs is done in a manner similar to changing the values in the `UD` structure in [Section 5.2](#). Let's say we wish to specify the prior to be something different from the uniform prior (which is the default). We must specify the

BOX 5.2 (*cont'd*)

range and resolution of values of possible thresholds to be included in the prior (or accept the default range and resolution: `-2:.01:2`) and define a prior distribution across that range:

```
>>alphas = -3:.01:3;
>>prior = PAL_pdfNormal(alphas, 0, 1);
```

The above call to `PAL_pdfNormal` returns the normal probability densities at the values in `alphas` using a mean equal to 0 and standard deviation equal to 1. Next, we update the relevant fields in RF:

```
>>RF = PAL_AMRF_setupRF(RF, 'priorAlphaRange', alphas, ...
    'prior', prior);
```

The RF options and their default values are given below.

During the testing session, the function `PAL_AMRF_updateRF` updates the posterior distribution after each trial and keeps a record of the stimulus intensities, responses, etc. In the code that controls our experiment we would have a loop:

```
while ~RF.stop
    amplitude = RF.xCurrent; % Note that other value may be used
    %Present trial here at stimulus intensity 'amplitude'
    %and collect response (1: correct, 0: incorrect)
    RF = PAL_AMRF_updateRF(RF, amplitude, response); %update RF
end
```

Note that we also pass the stimulus intensity (`amplitude`) to `PAL_AMRF_updateRF` (we did not do this with the up/down routines). This is because we are entirely free to ignore the value suggested by the procedure and present the stimulus at some other intensity. As such, we need to tell the procedure what stimulus intensity we actually used.

The RF structure stores the stimulus intensities that were actually used on each trial in the field `RF.x` and the corresponding responses in the field `RF.response`. The stimulus intensities that were suggested by the procedure on each trial are stored in the field `RF.xStaircase`. The final estimates of the threshold in the RF structure are `RF.mode` (the mode of the posterior distribution) and `RF.mean` (the mean of the posterior distribution). `RF.sd` contains the standard deviation of the posterior distribution. This standard deviation may serve as the standard error of estimate of the threshold if the threshold is estimated by the mean of the posterior distribution. The entries `RF.modeUniformPrior`, `RF.meanUniformPrior`, and `RF.sdUniformPrior` are analogous, except that they ignore the prior distribution provided by the researcher and instead use a uniform prior. Below we list the options for the Palamedes running fit routines that may be changed using the function `PAL_AMRF_setupRF`. Default values are those shown in curly brackets {}.

Continued

BOX 5.2 (*cont'd*)

```

PriorAlphaRange  vector {[−2:.01:2]}
Vector containing values of threshold to be considered in fit.
prior           vector {uniform across priorAlphaRange}
Prior distribution.
beta            positive scalar {2}
Slope parameter of PF to be fitted.
gamma           scalar in range [0–1] {.5}
Guess rate to be used in fits.
lambda          scalar in range [0–1] {.02}
Lapse rate to be used in fits.
PF              inline function {@PAL_Gumbel}
Form of PF to be used in fit. Refer to Chapter 4 for other possible functions.
stopCriterion   {'trials'} | 'reversals'
When set to 'trials', the staircase will terminate after the number of trials set in stopRule.
When set to 'reversals', the staircase will terminate after the number of reversals set in stopRule.
stopRule         positive integer {50}
see stopCriterion
xMin            scalar {-Inf}
Minimum stimulus intensity to be assigned to RF.xCurrent. If set to empty array, no minimum will be applied.
xMax            scalar {Inf}
Maximum stimulus intensity to be assigned to RF.xCurrent. If set to empty array, no maximum will be applied.
meanmode        {'mean'} | 'mode'
Indicates whether the mean or the mode of the posterior distribution should be assigned to RF.xCurrent.

```

appear a bit awkward perhaps, because we pretend to know all about the observer's PF except for the value of the threshold. As it turns out, though, the procedures are relatively robust when inaccurate assumptions regarding the PF are used. Nevertheless, we should use our best efforts to have our assumptions reflect the true state of the world as accurately as possible. We might base our guesses on our experience with similar experimental conditions, or we could perform some pilot experiments first.

The value we use for the slope affects the stepsizes that the running fit methods use. When a value for the slope is used that is much too high, the methods will use very small stepsizes. As a result, the method becomes sluggish, in that it will have a relatively hard time recovering from a series of lucky responses at very low stimulus intensities, for example. We might also consider allowing for some lapses to occur by setting the lapse parameter to a small value, such as 0.02 (which is the default). The risk you run by setting the lapse rate to 0 is

that when a lapse does occur at a high stimulus intensity, it will be followed by a series of trials at intensities well above threshold.

It is not necessary to present the stimulus on any trial at the intensity suggested by the running fit method. For example, you might prefer to present a stimulus at an intensity that is a bit higher than the threshold intensity in order to avoid frustration on the part of the observer. Presenting a stimulus at a high intensity every once in a while will remind the observer what to look for and might act as a confidence booster. You might also target a few distinct points along the PF so as to generate data that are suitable to estimate the threshold as well as the slope of the PF (although we recommend you to keep reading and use the psi method if your goal is to estimate the slope of the PF as well as the threshold).

Here, as with the up/down methods, it is a good idea to intertwine a few staircases randomly to avoid trial-to-trial dependencies. It is always possible to combine observations from the different staircases later and fit all simultaneously with a single PF, as in Hall's (1981) hybrid procedure ([Section 5.2.5](#)).

The choice of the prior to use deserves some attention. Remember that even if you choose to use a uniform prior, it is in practice not truly uniform, as it will be defined across a finite range of values. In other words, threshold values within the finite range of values are given equal likelihoods, but values outside of that range are assigned a likelihood of 0. It is important to not let your threshold estimates be affected significantly by your choice of the range of the prior. For example, if you use the mode of the posterior distribution, make sure that the value of the mode is in the range of values included within the prior and not at either boundary of the prior. When you use the mean of the posterior distribution as your threshold estimate, make sure that the posterior distribution is (effectively) contained entirely within the range of values in the prior, at least when your final estimate of the threshold is made. In case the posterior distribution is chopped off abruptly by the limits of the prior, your choice of these limits will have a significant effect on the threshold estimate.

In case observers are tested in multiple sessions in the same experimental conditions, we advise the use of the posterior distribution resulting from the previous session as the prior for a new session. In a sense, the staircase will proceed from session to session as if data were collected in a single session. The MATLAB® file that demonstrates the RF routines in Palamedes (`PAL_AMRF_Demo`) shows how to accomplish this. Caution should be exercised when there is reason to suspect that sensitivity varies from session to session, for example due to learning or fatigue. In such situations it might be best to start consecutive sessions with identical priors.

5.4 THE PSI METHOD AND VARIATIONS

5.4.1 The Psi Method

The psi method ([Kontsevich and Tyler, 1999](#)) is a sophisticated method that selects stimulus amplitudes so as to result in efficient estimation of both the threshold and the slope parameter of a PF. In many ways, it is similar to the Quest procedure. After each response, the psi method updates a posterior distribution, but now the posterior distribution is defined not only across possible threshold values but also across possible values of the slope parameter. We have discussed such posterior distributions in Chapter 4, and examples are shown in

Figures 4.14 and 4.15. As such, the psi method is similar to King-Smith and Rose's (1997) modified ZEST method that defined the posterior distribution across possible values of the threshold and the slope parameter as well. In the modified ZEST method, estimates for both the threshold and slope parameters are continuously updated. Stimuli are placed at intensities corresponding to specific probabilities of a correct response of the current estimate of the PF. The psi method, however, selects the stimulus intensity for the upcoming trial that minimizes the expected entropy in the posterior distribution after that trial. The use of entropy as the metric in which to define the amount of information gained from a trial was proposed by Pelli (1987), and was used by Cobo-Lewis (1997) to optimize information gained regarding which of a discrete set of categories observers belong to.

The psi method combines some quite complex issues. In order to break it down a bit, consider the following analogy. Imagine you are in a casino and you face a choice between two rather simple games: "Pick a Queen" and "Grab a Spade." In Pick a Queen you draw a card from a standard deck randomly. If you draw a queen, you win the game and receive \$26, but if you draw something other than a queen, you lose the game and pay \$3. In the game Grab a Spade, you pick a random card from a regular deck of cards and if it is a spade you win the game and receive \$20, but if it is not a spade you lose the game and pay \$8. Which game should you pick? One way to decide is to figure the expected monetary gain of each game. In the game Pick a Queen there is a 1/13 chance that you will indeed pick a queen, consequently win the game, and gain \$26. However, there is a 12/13 chance that you lose the game and pay \$3. The expected value of your monetary gain (x) in dollars is

$$E(x) = \frac{1}{13} \times 26 + \frac{12}{13} \times (-3) = -\frac{10}{13} \approx -0.77$$

You can think of the expected gain as your average gain per game if you were to play this game an infinite number of times. Note that your expected gain is negative. You are, after all, in a casino and casinos only offer games for which your expected monetary gain is negative. In a similar fashion, you can figure the expected monetary gain in the game Grab a Spade:

$$E(x) = \frac{1}{4} \times 20 + \frac{3}{4} \times (-8) = -1$$

The game Pick a Queen has a higher expected monetary gain (albeit still negative) so you choose to play Pick a Queen.

The strategy utilized by the psi method to decide which stimulus intensity to use on any trial is very similar. Where we are faced with a choice between two games to play, the psi method is faced with a choice between various stimulus levels to be used on the next trial. And where we select the game that maximizes our expected monetary gain, the psi method selects the stimulus intensity that minimizes the expected "entropy" in the posterior distribution.

The term entropy is used here as it is defined in the context of information theory (i.e., so-called Shannon entropy). Entropy in this context is a measure of uncertainty. A simple example will demonstrate the concept. Imagine a game in which a card is randomly drawn from a standard deck of cards, and you are to determine the suit of the card. On any given draw there is, of course, a probability equal to 1/4 that a heart is drawn, 1/4 that a spade is

drawn, etc. We can express the degree of uncertainty with regard to the suit of a randomly drawn card by the entropy H :

$$H = - \sum_i p_i \log_2 p_i \quad (5.3)$$

where i enumerates the four possible suits and p_i stands for the probability that the card is of suit i . So, in the above scenario the entropy is

$$H = - \left(\frac{1}{4} \log_2 \left(\frac{1}{4} \right) + \frac{1}{4} \log_2 \left(\frac{1}{4} \right) + \frac{1}{4} \log_2 \left(\frac{1}{4} \right) + \frac{1}{4} \log_2 \left(\frac{1}{4} \right) \right) = 2$$

Since we used the logarithm with base 2 to calculate the entropy, the unit of measurements is "bits." As such, you can think of the entropy as the number of (smart) yes/no questions that stand between the current situation and certainty (i.e., knowing for sure which suit the card is from). Imagine that you get to ask a yes/no question regarding the suit of the card. Let's say you ask this question: "Is the color of the suit red?" Whether the answer is "yes" or "no," it will reduce the uncertainty by half. For example, let's say the answer is "yes." We now know that the card is either a heart or a diamond with equal probability. The entropy becomes:

$$H = - \left(\frac{1}{2} \log_2 \left(\frac{1}{2} \right) + \frac{1}{2} \log_2 \left(\frac{1}{2} \right) + 0 \log_2(0) + 0 \log_2(0) \right) = 1$$

(Note that we have defined $0 \log_2(0)$ to equal 0, as $\lim_{p \downarrow 0} p \log(p) = 0$). In words, the answer to the question reduced the uncertainty from 2 bits to 1 bit. We need one more question to attain certainty. For example, in case we ask the following: "Is the card a heart?" and the answer is "no," we know for certain that the card must be a diamond. Numerically, entropy would indeed be reduced by another bit to equal 0:

$$H = -(1 \log_2(1) + 0 \log_2(0) + 0 \log_2(0) + 0 \log_2(0)) = 0$$

reflecting the absence of uncertainty altogether.

In the context of the psi method, entropy measures the uncertainty associated with the values of the threshold and slope of the PF. The posterior distribution gives us the probability density function across combinations of threshold (a) and slope (b) values. Let us call this $p(a,b)$ here. In practice the posterior is defined across discrete values of a and b in which case we calculate the entropy in the posterior distribution as

$$H = - \sum_a \sum_b p(a,b) \log_2 p(a,b) \quad (5.4)$$

A low value for the entropy in the posterior distribution corresponds to a high degree of precision of the parameter estimates. It might be beneficial here to consider two extremes. The entropy H would attain its maximum value if $p(a,b)$ is identical for all combinations of a and b ; in other words, when you have no information to indicate that any combination of $p(a,b)$ is more likely than any other. This situation is comparable to the situation above in which a random card is drawn from a deck and you have not yet had an opportunity to ask any questions. In this situation $p(Heart) = p(Diamond) = p(Club) = p(Spade) = 1/4$. At the other

extreme, the entropy H would attain its minimum value (0) when $p(a,b)$ is equal to 1 for one particular combination of values for a and b (and of course would then equal 0 for all other combinations of a and b). In other words, we would then be certain which values of a and b are the correct values for the threshold and slope. This situation is comparable to the situation above in which we know (after collecting information by asking questions) that the card is, say, of the suit heart. That is, $p(\text{Heart}) = 1$, $p(\text{Diamond}) = p(\text{Club}) = p(\text{Spade}) = 0$. Consider now Figure 4.15 in Chapter 4. It shows a posterior distribution across possible threshold (a) and slope (b) values after some responses have been collected (and a prior has been applied). This represents a situation in between the two extremes just discussed. That is, some combinations of a and b are now much more likely than others, but some uncertainty regarding the values of a and b remains. Conceptually, the smaller the “blob” in a contour plot of the posterior distribution is, the lower the entropy will be.

To recap, the psi method considers a range of possible stimulus intensities to use on the next trial (compare: games to play) and for each calculates the entropy (compare: monetary outcome) that would result from both a correct response and an incorrect response. It also considers the probabilities of obtaining a correct and incorrect response (compare: probability of win or loss of game). From these, the psi method calculates the expected entropy (compare: expected monetary gain) associated with each of the possible choices of stimulus intensity. It then selects that stimulus intensity associated with the lowest entropy.

Note that stimulus intensity is a continuous variable such that, in theory, the psi method has the choice between an infinite number of values. In practice, the psi method chooses a stimulus intensity from a relatively large number of discrete stimulus intensities in a specified range. The principle is the same as compared to choosing one of the two card games to play, however. That is, the expected entropy is calculated for all possible discrete stimulus intensities and the psi method selects that intensity that will lead to the highest expected entropy.

You may have noticed that we have thus far omitted one last piece of the puzzle. We stated above that the psi method calculates the expected entropy associated with a stimulus intensity. However, in order to do so the psi method must supply the probability of a correct response for that stimulus intensity. These probabilities are given by the PF that relates these probabilities to the stimulus intensity. However, we do not know the characteristics of this PF. As a matter of fact, the reason we are using the psi method is to find it! Note that we did not face this problem above when deciding which game to play. There we knew from the outset the probability of a win for each game. The probability of winning Pick a Queen is $1/13$ and that of winning Grab a Spade is $1/4$. The situation the psi method faces is a bit more daunting. It is as if we were to choose between Pick a Queen or Grab a Spade not knowing how many queens or spades are in the deck. However, you might imagine that as we start playing the games and witness the outcomes of the draws, we start to get an idea as to the makeup of the deck of cards, and we use this to adjust our choice of game. This is similar to what the psi method does and is of course what makes the psi method an adaptive method. Specifically, after each trial the psi method finds the best-fitting PF to the responses collected on all of the previous trials in the staircase. It does this based on the continuously updated posterior distribution using a Bayesian criterion (Chapter 4). The probability of a correct response on the next trial at each of the stimulus intensities under consideration is then determined from the best-fitting PF.

An example run in which stimulus intensity was guided by the psi method is shown in [Figure 5.4](#). The same plotting conventions are used here as in [Figures 5.2 and 5.3](#), except that the threshold estimates on each trial (black line) are also shown. A couple of observations should be made. First, it is apparent that, at the start of the run, the psi method selects stimulus intensities that are at or near the current threshold estimate. This is, of course, the best placement rule to determine the value of the threshold. However, in order to gain information regarding the slope of the PF, measurements need to be made at multiple points along the PF. Indeed, as the run proceeds, the psi method also starts to select stimulus intensities well above and below the running threshold estimate.

5.4.2 Termination Criteria and the Threshold and Slope Estimates

When the psi method is used, it would make no sense to use the number of reversals as a termination criterion. In the up/down and running fit methods, reversals occur mainly when the stimulus amplitude has crossed the targeted threshold. As such, the number of reversals is closely tied to the degree of accuracy with which the threshold can be determined. In the psi method, there is no such close relationship. From [Figure 5.4](#), the pattern with which stimulus amplitudes are selected and reversals occur in the psi method appears much more haphazard as compared to the up/down and running fit methods. For example, the stimulus amplitude may be increased after a correct response (e.g., after the correct response on trial 41 in [Figure 5.4](#)). Thus, when the psi method is used, a run is terminated after a certain number of trials have occurred.

Most naturally, one would use the posterior distribution created during a run to derive an estimate of the threshold and the slope and their standard errors. This would be done by the methods described in Chapter 4. However, as with other adaptive methods, one is free to collect data using the psi method but consequently use any method to derive one's final parameter estimates. One specific suggestion might be to divide out the prior before we make our final parameter estimates, for the same reasons we discussed in the context of the Quest procedure. [Box 5.3](#) explains how to use the Palamedes toolbox to set up a psi method adaptive run.

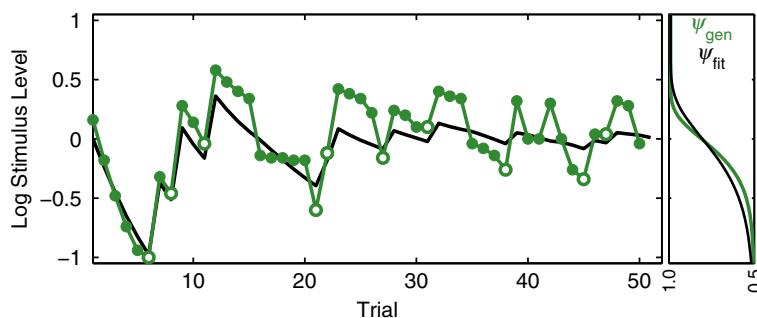


FIGURE 5.4 Example of a simulated psi method staircase. Correct responses are indicated by the filled symbols, and incorrect responses are indicated by open symbols. The black line displays the running estimate of the threshold based on the posterior distribution. The responses were generated by a Gumbel function with $\alpha = 0$, $\beta = 2$, $\gamma = 0.5$, and $\lambda = 0.01$. The generating function (ψ_{gen}) is shown on the right in green, and the function fitted by the psi method (ψ_{fit}) after completion of the 50 trials is shown on the right in black.

BOX 5.3**THE PSI METHOD IN PALAMEDES**

The functions that implement the psi method in the Palamedes toolbox are used in a fashion very similar to those that implement the up/down methods and the running fit methods. We will first demonstrate usage in the context of the original psi method, in which a fixed guess and lapse rate are used. We then demonstrate how to set up the routines in order to run a session that uses the psi-marginal method (Section 5.4.4).

In order to run the original psi method, we first set up a structure `PM` by calling `PAL_AMPM_setupPM`:

```
>>PM = PAL_AMPM_setupPM;
```

In case we wish to change some of the options, we can of course do so (see below for a listing of options). Let's say we wish the testing to terminate after 50 trials, we would like the threshold values to be included in the posterior distribution to be $[-2:1:2]$, we would like the psi method to consider stimulus intensities from the vector $[-1:2:1]$, and we wish to accept the default values for the other options. We can change the options in the `PM` structure by calling `PAL_AMPM_setupPM`:

```
>>PM = PAL_AMPM_setupPM(PM, 'priorAlphaRange', [-2:1:2], ...
    'stimRange', [-1:2:1], 'numtrials', 50);
```

Note that the first argument to the function in the above call is the previously created structure `PM`. This is optional, but the function will behave differently, depending on whether we provide an existing structure `PM` or not. In case we do supply an existing structure, the function will retain any previous changes we had applied to this structure. In case we do not, the function will create a new structure from scratch and overwrite any changes to an existing structure.

The program that controls the experiment would contain a loop such as this:

```
while ~PM.stop
    %Present trial here at stimulus intensity PM.xCurrent
    %and collect response (1: correct, 0: incorrect)
    PM = PAL_AMPM_updatePM(PM, response); %update PM structure
end
```

Note that on each trial the stimulus has to be presented at the intensity indicated in the entry in the field `PM.xCurrent`. However, this field will always contain a value that is in the `PM.stimRange` vector, which is under our control. Thus, if (for whatever reason) we can (or wish to) present stimuli only at intensities $-2, -1, -0.5, 0, 1/3$, and π , we need to make that clear to the psi method beforehand. We do that by defining the vector `PM.stimRange` accordingly:

```
>>PM = PAL_AMPM_setupPM('stimRange', [-2 -1 -.5 0 1/3 pi]);
```

The psi method will now only suggest values that we can actually present.

Once the run completes, the Bayesian estimates of the PF's parameters are given in the `PM` structure in the fields `PM.threshold`, `PM.slope`, `PM.guess`, and `PM.lapse`. Standard errors of parameters are given in `PM.seThreshold`, etc. These are in fact all vectors that list the estimates

BOX 5.3 (*cont'd*)

derived after each trial in the run. All the estimates are also derived while ignoring any custom prior provided by the user and these estimates are stored in `PM.thresholdUniformPrior`, etc.

The Palamedes file `PAL_AMPM_Demo` demonstrates use of the psi method. While the demonstration program runs, it will display the posterior distribution after every trial. For the original psi method, the posterior distribution across threshold and slope values is shown in panel (b) in the figure created by `PAL_AMPM_Demo`. Panels (c) and (d) display the posterior distribution across the threshold and slope values, respectively. You will note that as the session proceeds, and more information is obtained, the posterior distribution will get narrower with respect to both the threshold and slope values.

In order to demonstrate usage of the psi-marginal method (Section 5.4.4) let us imagine a 4AFC task. Let us further imagine that we are only interested in the value of the threshold. However, we do not wish to assume a fixed value for the slope. There is also a specific concern that the lapse rate might be high and may have a large influence on the threshold estimate. Thus, we wish to maintain a posterior distribution across the threshold, slope, and lapse rate values, but we wish to optimize the stimulus placement only with regard to the value of the threshold.

Let us first create a new structure `PM`:

```
>>PM = PAL_AMPM_setupPM;
```

Next, we'll define the dimensions of the prior distribution:

```
>>PM = PAL_AMPM_setupPM(PM, 'priorAlphaRange', [-1:.1:1], ...
    'priorBetaRange', [-1:.1:1], 'priorGammaRange', 0.25, ...
    'priorLambdaRange', [0:.01:.4]);
```

Note that the values we supply for the slopes are logarithmic values (using base 10). Thus, using the values we supplied, the lowest slope value (β in the equation of the PF, Chapter 4) included the posterior equals 0.1 (i.e., 10^{-1}), and the highest equals 10 (10^1). Next, we will define the stimulus values to be used and specify that we wish to run 250 trials:

```
>>PM = PAL_AMPM_setupPM(PM, 'stimRange', [-1:.1:.4 1], 'numTrials', 250);
```

Note that we included one stimulus value that is very high relative to the others. It is a good idea to include a very high-intensity stimulus when the lapse rate is included in the prior distribution. Stimuli presented at this intensity can serve to present “free trials,” which provide a lot of information regarding the value of the lapse rate (see Box 4.6). Finally, we specify that the method should marginalize the slope and lapse rate, since we are not concerned with their value per se.

```
>>PM = PAL_AMPM_setupPM(PM, 'marginalize', 'slope', 'marginalize', 'lapse');
```

The effect this will have on the behavior of the method is that it will address the value of the marginalized parameters only if doing so maximizes the information gain regarding the value of the free parameter(s) that are not marginalized (here: the threshold).

BOX 5.3 (*cont'd*)

In the program that controls the experiment we would again use the function `PAL_AMPM_updatePM` to update the posterior after each trial and find the stimulus intensity to be used on the next trial. The demo program `PAL_AMPM_Demo` is interactive and can be set up to demonstrate the psi-marginal method. Note that when the lapse rate is included in the prior distribution, the method has a tendency to present consecutive series of “free trials.” This can be avoided by temporarily using a fixed lapse rate (as the original psi method does). This will generally result in stimulus placements that are not free trials, although it may occasionally use free trials early in a given run. `PAL_AMPM_Demo` demonstrates how to use this feature.

Below we list the options for the Palamedes psi method routines that may be changed using the function `PAL_AMPM_setupPM`. Default values are those shown in curly brackets {}.

`priorAlphaRange` scalar or vector {[−2:.05:2]}

Values of threshold to be considered in posterior distribution.

`priorBetaRange` scalar or vector {[−1:.05:1]}

Log-transformed values of slope to be considered in posterior distribution.

`priorGammaRange` scalar or vector {.5}

Values of guess rate to be considered in posterior distribution.

`priorLambdaRange` scalar or vector {.02}

Values of lapse rate to be considered in posterior distribution.

`stimRange` vector {[−1:.1:1]}

stimulus values to be considered on each trial.

`Prior` array {uniform across all of the free parameters}

Prior distribution.

`PF` inline function {@PAL_Gumbel}

Form of PF to be assumed by psi method. Refer to Chapter 4 for other possible functions.

`numTrials` positive integer {50}

Length of run in terms of number of trials.

`gammaEQlambda` logical {false (0)}

When `gammaEQlambda` is set to true, gamma and lambda are both assumed to result from lapses, and one value will be estimated for both. Use `priorLambdaRange` to set the range of values to be considered; `priorGammaRange` will be ignored.

`marginalize` vector {[]} or string

Identifies which of the PF’s parameters should be marginalized. May be set using strings ('threshold', 'slope', 'guess', 'lapse') as above or as a vector identifying parameters to be marginalized (1: threshold, 2: slope, 3: guess rate, 4: lapse rate).

5.4.3 Some Practical Tips

Many of the practical tips we gave for the running fit methods apply to the psi method for the same reasons. Here one would also want to allow for lapses to occur by setting the lapse rate to a small nonzero value, such as 0.02. One should also define the prior distribution

across ranges of threshold and slope values that are wide enough to accommodate the entire posterior distribution, at least at the time when we derive our final parameter estimates, so as to not let our estimates be determined in large part by the ranges of values we happened to have included in the prior.

The psi method is quite taxing on the RAM memory of your computer. Also, updating the posterior distribution and calculating the expected entropy will involve quite a few calculations on every trial. What this means in practical terms is that you need to find an acceptable balance between the resolution of your posterior distribution and possible stimulus values on the one hand and the time you will allow to perform the necessary computations between trials and the amount of RAM memory you have available on the other.

5.4.4 Psi-Method Variations

The core idea behind the psi method is to select stimulus intensities such that the expected information gain regarding the value of parameters is maximized. This principle can be easily extended to the optimization of any parameter. One obvious extension to the psi method is to define the posterior distribution across any combination of the four parameters. For example, in a “yes/no” task, the value of the “false alarm rate” is unknown. The false alarm rate would correspond to what we have referred to heretofore as the guess rate. Thus, in the context of a yes/no task one could define the posterior distribution across values of the threshold, the slope, and the guess rate. Similarly, we can include the lapse rate in the posterior distribution.

However, the method would then seek to optimize the estimation of these parameters as well as the parameters of primary interest. Generally, however, we are not interested in the value of the guess and/or lapse rate parameter per se. They are aptly referred to as “nuisance parameters” because even though we do not care about their value, uncertainty regarding their value adds to the uncertainty regarding the values of the parameters that are of primary concern (the threshold and the slope). We made this argument in Box 4.6 using Figure B4.6.1. It shows the posterior distribution across the threshold, slope, and lapse rate values based on stimulus placements typical of that obtained under the original psi method. To illustrate the issue, let us contrast two scenarios. The first scenario is very hypothetical. In this scenario we know, without any uncertainty, what the value of the lapse rate is. Let’s say we know the lapse rate is equal to 0.03. In effect, what this would mean is that our uncertainty regarding the values of the threshold and slope is given by the posterior distribution contained in the single slice that corresponds to a lapse rate of 0.03. In the other, more realistic scenario, we do not know what the value of the lapse rate is. Simply put, we do not know which slice in Figure B4.6.1 is the correct posterior distribution across threshold and slope. Since the location of the posterior across threshold and slope varies systematically across the different possible values of the lapse rate (i.e., the posterior is slanted in threshold \times slope \times lapse rate space), the uncertainty regarding the lapse rate contributes to the uncertainty regarding the threshold and slope values. Specifically, not knowing the value of the lapse rate means that our uncertainty regarding the threshold and slope is given by the posterior in which the lapse rate has been marginalized. This marginal distribution is obtained simply by summing across lapse rates and is shown on the floor of the parameter space in Figure B4.6.1. In a sense, this marginal distribution gives the posterior distribution across threshold and slope

values, taking into account that we do not know the value of the lapse rate. By comparing this marginal posterior distribution to the posterior that corresponds to a known lapse rate of 0.03 (or any other value) it is clear that the uncertainty in threshold and slope values is much greater in the marginal distribution. Thus, the uncertainty regarding the value of the lapse rate contributes to our uncertainty regarding the threshold and slope values. We could of course simply include the lapse rate in the posterior that the psi method attempts to reduce the uncertainty in. The method will then have the explicit goal of reducing the spread of the posterior along all three axes in Figure B4.6.1. Reducing the vertical spread (i.e., gathering information regarding the value of the lapse rate) will also contribute to reducing the uncertainty regarding the threshold and slope values. A problem with this strategy, however, is that while reducing the vertical spread may be the optimal way on a given trial to reduce the overall uncertainty in the posterior distribution across the threshold, slope, and lapse rates, it may not be the optimal way to reduce uncertainty regarding the threshold and slope.

One of us (Prins, 2013) has proposed the “psi-marginal method,” which maintains a posterior distribution that includes the lapse rate but selects stimulus placements that will minimize the uncertainty (as measured by entropy) in the distribution from which the nuisance parameters have been marginalized. With reference to Figure B4.6.1, the method would maintain the entire posterior distribution but would select stimulus intensities that minimize the expected uncertainty in the marginal distribution shown on the floor of the parameter space. The method will then have the option to reduce uncertainty regarding the value of the lapse rate but will only do so if this is the best strategy to reduce uncertainty regarding the threshold and slope.

The psi-marginal method is the first adaptive method to deal with nuisance parameters in an adaptive manner without optimizing the estimation of the nuisance parameters themselves. Note that the psi-marginal method is extremely flexible. Essentially, for each of the four parameters of a PF we can specify whether the method should treat it as a parameter of which the values are known (as the original psi method does with the guess and lapse rates), should treat it as a parameter whose estimation should be optimized (as the original psi method does with the threshold and slope), or should treat it as a nuisance parameter of unknown value whose estimation should only proceed in order to subserve optimization of the parameters of interest. The obvious application is to include the lapse rate as a nuisance parameter, but we can for example also choose to optimize estimation of the threshold only, while treating the slope as well as the lapse rate as nuisance parameters. This is somewhat similar to using Quest but without the need to assume a specific value for the slope and lapse rate. To give one more example, perhaps we are exclusively interested in the lapse rate, in which case we can treat the threshold and slope as nuisance parameters! Box 5.3 demonstrates the usage of the Palamedes routines that implement the psi-marginal method.

Other extensions of the psi method have been proposed. For example, Lesmes et al. (2010) reparameterized thresholds of multiple PFs, collectively describing contrast detection performance across a range of spatial frequencies into four parameters that summarize an observer’s contrast sensitivity function (CSF). The four parameters are the CSF’s peak sensitivity, the peak spatial frequency, the bandwidth, and the truncation level at low spatial frequencies. The concept of reparameterization is discussed in some detail in Chapter 9. Having reparameterized the thresholds allows for the creation of a posterior distribution across the four parameters of the CSF, which means that the optimization principle behind the

psi method can now be used to optimize estimation of the four CSF parameters. This qCSF (or “quick CSF”) method selects stimuli from a two-dimensional stimulus space: the method selects a spatial frequency to be tested as well as a stimulus contrast. [Kujala and Lukka \(2006\)](#) also utilized reparameterization of thresholds in order to optimize parameter estimation across a multidimensional stimulus space. Note that the idea behind the psi-marginal method may be extended to these (and other) methods. For example, if one is interested only in the CSF’s peak spatial frequency, the method can be given the explicit goal of reducing the entropy in the posterior distribution that is defined across that parameter only by marginalizing the other three parameters of the CSF.

EXERCISES

1. By using a program similar to `PAL_AMUD_Demo`, try out what happens when the stepsizes are much smaller or greater than suggested by the text.
2. By using a program similar to `PAL_AMRF_Demo`, try out what happens when the assumed value for the slope differs significantly from the true “generating” slope. Use slopes that are much too high and much too low.
3. By using a program similar to `PAL_AMRF_Demo`, try out what happens when the assumed value for the lapse rate equals 0 but lapses do in fact occur. How does this affect the value for the threshold and slope parameter estimates? What if the assumed value for the lapse rate does not equal 0 (say it equals 0.01), but lapses in fact do not occur?
4. Somebody draws a card from a standard deck. You are to guess what suit it is. Before you have to make a guess, you get to ask one yes/no question. Show that the question “Is the color of the suit red?” results in a lower expected entropy as compared to the question “Is the suit hearts?”
5. Repeat question 3 in the context of the psi method.

References

- Alcalá-Quintana, R., García-Pérez, M.A., 2004. The role of parametric assumptions in adaptive Bayesian estimation. *Psychol. Methods* 9, 250–271.
- Bechara, A., Damasio, H., Tranel, D., Damasio, A.R., 1997. Deciding advantageously before knowing the advantageous strategy. *Science* 275, 1293–1295.
- Cobo-Lewis, A.B., 1997. An adaptive psychophysical method for subject classification. *Percept. Psychophys.* 59, 989–1003.
- Dixon, W.J., Mood, A.M., 1948. A method for obtaining and analyzing sensitivity data. *J. Am. Stat. Ass.* 43, 109–126.
- García-Pérez, M.A., 1998. Forced-choice staircases with fixed stepsizes: asymptotic and smallsample properties. *Vision Res.* 38, 1861–1881.
- Hall, J.L., 1968. Maximum-likelihood sequential procedure for estimation of psychometric functions [abstract]. *J. Acoust. Soc. Am.* 44, 370.
- Hall, J.L., 1981. Hybrid adaptive procedure for estimation of psychometric functions. *J. Acoust. Soc. Am.* 69, 1763–1769.
- Kaernbach, C., 1991. Simple adaptive testing with the weighted up/down method. *Percept. Psychophys.* 49, 227–229.
- King-Smith, P.E., Grigsby, S.S., Vingrys, A.J., Benes, S.C., Supowitz, A., 1994. Efficient and unbiased modifications of the QUEST threshold method: theory, simulations, experimental evaluation, and practical implementation. *Vision Res.* 34, 885–912.

- King-Smith, P.E., Rose, D., 1997. Principles of an adaptive method for measuring the slope of the psychometric function. *Vision Res.* 37, 1595–1604.
- Kontsevich, L.L., Tyler, C.W., 1999. Bayesian adaptive estimation of psychometric slope and threshold. *Vision Res.* 39, 2729–2737.
- Kujala, J.V., Lukka, T.J., 2006. Bayesian adaptive estimation: the next dimension. *J. Math. Psychol.* 50, 369–389.
- Lesmes, L.A., Lu, Z.L., Baek, J., Albright, T.D., 2010. Bayesian adaptive estimation of the contrast sensitivity function: the quick CSF method. *J. Vis.* 10, 17.
- Pelli, D.G., 1987. The ideal psychometric procedure. *Invest. Ophthalmol. Vis. Sci.* 28 (Suppl.), 366.
- Pentland, A., 1980. Maximum likelihood estimation: the best PEST. *Percept. Psychophys.* 28, 377–379.
- Prins, N., 2013. The psi-marginal adaptive method: how to give nuisance parameters the attention they deserve (no more, no less). *J. Vis.* 13 (7), 3.
- Prins, N., Kingdom, F.A.A., 2009. Palamedes: Matlab routines for analyzing psychophysical data. <http://www.palamedestoolbox.org>.
- Taylor, M.M., Creelman, C.D., 1967. PEST: efficient estimates on probability functions. *J. Acoust. Soc. Am.* 41, 782–787.
- Watson, A.B., Pelli, D.G., 1983. QUEST: a Bayesian adaptive psychometric method. *Percept. Psychophys.* 33, 113–120.
- Wetherill, G.B., Levitt, H., 1965. Sequential estimation of points on a psychometric function. *The Br. J. Math. Stat. Psychol.* 18, 1–10.

Signal Detection Measures*

Frederick A.A. Kingdom¹, Nicolaas Prins²

¹McGill University, Montreal, Quebec, Canada; ²University of Mississippi, Oxford, MS, USA

OUTLINE

6.1 Introduction	150	6.2.10 Comparing Pcs from d's Across Different Tasks	166
6.1.1 What is Signal Detection Theory (SDT)?	150	6.2.11 Modeling Psychometric Functions with SDT	166
6.1.2 A Recap on Some Terminology: N, m and, M	150	6.3 Section B: Theory	171
6.1.3 Why Measure d'?	151	6.3.1 Relationship Between z-Scores and Probabilities	171
6.2 Section A: Practice	153	6.3.2 Calculation of d' for M-AFC	172
6.2.1 Basic Assumptions	153	6.3.3 Calculation of d' and Measures of Bias for 1AFC Tasks	175
6.2.2 Converting Pc to d' for Standard M-AFC Tasks	153	6.3.4 Calculation of d' for Unbiased and Biased 2AFC Tasks	178
6.2.3 Measuring d' for 1AFC Tasks	154	6.3.5 Calculation of d' for Same-Different Tasks	180
6.2.4 Performing a Rating Scale Experiment with 1AFC	158	6.3.6 Calculation of d' for Match-to-Sample Tasks	185
6.2.5 Measuring d' for 2AFC Tasks with Observer Bias	160	6.3.7 Calculation of d' for M-AFC Oddity Tasks	185
6.2.6 Measuring d' for Same-Different Tasks	162	Further Reading	187
6.2.7 Measuring d' for Match-to-Sample Tasks	164	Exercises	187
6.2.8 Measuring d' for M-AFC Oddity Tasks	165	References	188
6.2.9 Estimating Pc_{max} with Observer Bias	165		

*This chapter was primarily written by Frederick Kingdom.

6.1 INTRODUCTION

6.1.1 What is Signal Detection Theory (SDT)?

In performance-based psychophysical tasks there are many situations in which proportion correct P_c is not a valid measure of performance. In this chapter we will examine some of these situations and describe a popular alternative measure termed d' ("dprime"). d' is a measure derived from a branch of psychophysics known as Signal Detection Theory (SDT). In Chapter 4 (Section 4.3.1.2), SDT was introduced as one of the models of how perceptual decisions were made in a forced-choice task. The SDT model attempted to explain the shape of the psychometric function that related P_c to stimulus magnitude. It was argued that the presence of internal noise, or uncertainty, led to stimuli being represented in the brain not by a single point along a sensory continuum but as a random sample drawn from a distribution with a mean and a variance. SDT is therefore a theory of how observers make perceptual decisions, given that the stimuli are represented stochastically (or probabilistically) inside the brain. The aim of this chapter is to discuss why d' is a useful measure of performance, to show how one converts conventional measures of performance such as P_c into d' , and to provide the theory behind those conversions.

SDT is a large topic and it is impossible to do it justice in a single chapter of an introductory book on psychophysics. There are a number of excellent books and articles on SDT (see Further Reading at the end of the chapter) that cover a much wider range of material and provide a more in-depth treatment than is possible here. In particular, Macmillan and Creelman's (2005) comprehensive treatment of SDT is strongly recommended as an accompaniment to this chapter. Although our chapter is modest by comparison to textbooks specializing in SDT, it nevertheless aims to do more than just scratch the surface of the topic. Section A is intended to familiarize readers with the basic concepts of SDT and provide the practical tools necessary for converting psychophysical measurements into d' s, and vice versa, without needing to understand the underlying theory. The theory is provided in Section B, and we have attempted to make it as accessible as possible.

6.1.2 A Recap on Some Terminology: N, m and, M

As elsewhere in this book, we use the term "forced-choice" for any task in which observers are required on each trial to make a forced-choice response, irrespective of the number of stimuli or stimulus alternatives presented in the trial. We denote the number of stimuli presented during a trial as N . The number of response choices available to the observer is denoted by m , and as we saw in Chapter 4 the value of m determines the expected chance performance or guessing rate of the task, given by $1/m$. Remember that m and N are not always the same. For example, in a yes/no task N is 1 because only one of two stimulus states is presented per trial, target-present or target-absent, but m is 2 because there are two choices of response—"yes" or "no." In a same-different task N can be 2 or 4, depending on whether the Same and Different pairs are presented on separate trials or together in the same trial. When the pairs are presented on separate trials the task for the observer is to respond "same" or "different," whereas when presented together in the same trial the task is to respond "1" or "2," depending on the interval that contains the Different (or Same) pair. However, for both varieties of same-different task, m is 2.

For the purposes of SDT the third parameter described in Chapter 2, M , is especially important. M is the “number of stimulus alternatives presented per trial.” In the $N=4$ same-different task described above, M is 2, as two stimulus alternatives are presented per trial—the Same pair and the Different pair. Although m is also 2 for this task, m and M are not always the same. For example, with the yes/no task m is 2 but M is 1, since although there are two available response choices, there is only one stimulus alternative per trial: either target-present or target-absent. As we stated in Chapter 2, forced-choice tasks in this book are prefixed with the value of M . Table 6.1 summarizes the relationship between N , m , and M for the main varieties of tasks discussed in this chapter.

One could argue that because this chapter deals exclusively with forced-choice tasks, prefixing every task with the acronym AFC is unnecessary because it is redundant, and that AFC should be used only sparingly, as in standard SDT texts (e.g., Macmillan and Creelman, 2005). However, in a book dealing with psychophysical procedures that are not all forced-choice, the acronym helps to make explicit the ones that are.

Recall, also, that for most tasks the stimulus alternatives can be presented in spatial or temporal order, and hence be denoted as AFC or IFC. AFC, however, is the generic acronym, so we have adopted this as the default and use IFC only when referring to temporally ordered stimuli. From the point of view of SDT, however, the two acronyms are interchangeable.

6.1.3 Why Measure d' ?

Suppose we wanted to compare the results of two texture-segregation experiments, one that employed a standard 4AFC task and the other a standard 2AFC task. The stimuli employed in texture segregation experiments typically consist of a “target” texture embedded in a “background” texture. The target and background differ in some textural

TABLE 6.1 Relationship between M , N , and m for the psychophysical tasks discussed in the chapter

Task	Acronym prefixed by number of stimulus alternatives per trial M	Number of stimuli per trial N	Number of response options per trial m
Yes/no	1AFC	1	2
Symmetric single-alternative	1AFC	1	2
Standard two alternative forced-choice	2AFC	2	2
Single alternative same-different	1AFC	2	2
Two alternative same-different	2AFC	4	2
Two alternative match-to-sample	2AFC	3	2
Three alternative oddity	3AFC	3	3
Standard M -alternative forced-choice	M -AFC	M	M
M -alternative match-to-sample	M -AFC	$M+1$	M
M -alternative oddity	M -AFC	M	M

property, such as the average orientation or average size of the texture elements. In the popular 4AFC version, the target is embedded in one of four quadrants of the background texture, and on each trial the observer selects the quadrant containing the target. In the 2AFC version, the embedded target is typically positioned on either side of the fixation point. In the 4AFC task proportion correct (P_c) would be expected to range from 0.25 to 1, since the guessing rate $1/m$ is 0.25. For the 2AFC version on the other hand, P_c would be expected to range from 0.5 to 1, since the guessing rate is 0.5. Yet, presumably, the underlying sensory mechanisms involved in segregating the target from the background are the same for both the 2AFC and 4AFC tasks, especially if the target locations are arranged equidistant from fixation in order to stimulate equivalent visual mechanisms. Thus, any differences in performance between the two tasks, which will be most evident when performance is close-to-chance, are unlikely to be due to differences in the observer's sensitivity to the stimuli, but instead due to differences in the uncertainty of the target location. Put another way, with four possible target locations the observer is more likely to make a mistake than with two target locations, all else being equal. And the more possible target locations, the more mistakes the observer will likely make. One reason for using d' is that it can remove, or take into account, the effects of target location uncertainty, providing a measure of performance that is procedure-free. In other words, for M -AFC tasks, d' may equate performance across M . We stress "may" because it is ultimately an empirical question, not a foregone conclusion, as to whether d' does equate performance across M , and there are some situations where it has been shown not to do so (e.g., Yeshurun et al., 2008). Figure 6.1 shows hypothetical P_c data for a 2, 4, and 8AFC task, together with the same data converted into d' . In this case, converting to d' brings the data into alignment.

Although the most popular value of M in forced-choice tasks is 2 (either 2AFC or 2IFC), M can be much higher. For example, one of the authors once conducted an experiment in which observers were required to choose a column of pixels with the highest average intensity from 256 columns—see Kingdom et al. (1987) for details. So M for this task was 256!

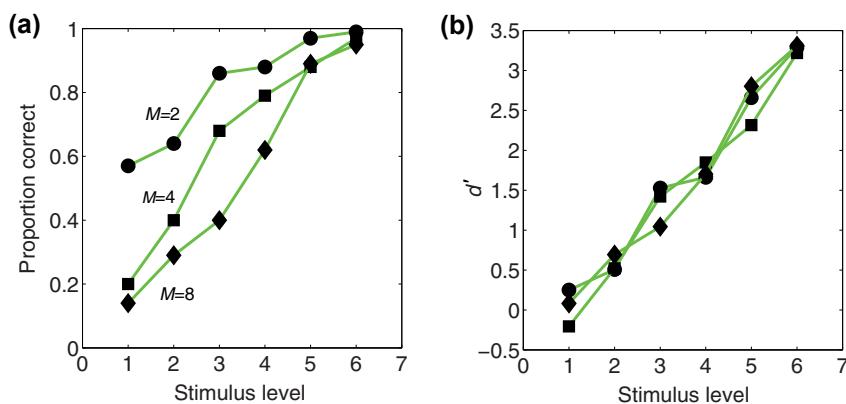


FIGURE 6.1 (a) Hypothetical P_c (proportion correct) data for an $M = 2$, $M = 4$, and $M = 8$ forced-choice task. (b) The same data plotted as d' .

A second, and for many investigators the most important, reason for using d' is that certain types of psychophysical tasks are prone to the effects of observer bias. In Chapters 2 and 3 we noted that in the yes/no task observers tend to be biased toward responding "yes" or "no," irrespective of their underlying sensitivity to the target. As we shall see, the greater the bias, the smaller the expected P_c , making P_c an invalid measure of sensitivity. d' is designed to take into account the effects of bias, in other words to provide a relatively "bias-free" measure of performance.

A third use of d' is that it forms the basis for estimating certain important sensory parameters. In this chapter we shall see how, given certain assumptions, d' provides the basis for estimating how the internal sensory representation of a stimulus dimension grows with its magnitude—the so-called transducer function. In the following chapter dealing with summation we show how d' forms the basis of a number of formulations that are useful for testing how multiple stimuli combine to reach detection threshold.

6.2 SECTION A: PRACTICE

6.2.1 Basic Assumptions

There are two important assumptions underlying all the analyses described in this chapter. The first concerns the stimuli. We assume that all the stimulus alternatives are presented the same number of times in each experiment. Thus, in a yes/no task we assume that there are as many target-present as target-absent trials, and in a same-different task as many Same pairs as Different pairs. The second assumption is that the observer's internal noise variance is the same across stimulus alternatives. This is the "default" assumption in SDT analyses. The only analysis described below that does not make the equal-noise-variance assumption is the one dealing with receiver-operating-characteristic (ROC) curves (Section 6.2.4).

6.2.2 Converting P_c to d' for Standard M-AFC Tasks

We begin with the standard M -AFC task, where M is any value greater than 1. For this class of task the three variables, N , M , and m , are equal. In an M -AFC task, one of the alternatives on each trial contains the target, while the remaining $M-1$ alternatives contain no target. If the observer selects the alternative containing the target his or her response is scored "correct," otherwise "incorrect," and P_c is calculated as the proportion of trials in which the observer is scored correct.

How are d' , P_c , and M related in a standard forced-choice task? As d' increases, so does P_c , so a larger P_c implies a larger d' . But how do these quantities vary with M ? First, d' is 0 whenever performance is at chance, which for the standard M -AFC task is a P_c of $1/M$. Thus d' is 0 for a P_c of 0.5 when $M = 2$, and d' is 0 for a P_c of 0.333 when $M = 3$, etc. Any P_c below these values will produce a negative d' . Second, for a given P_c , d' increases with M . Why? The higher the value of M the greater the chance that one of the $M-1$ nontarget intervals will produce the biggest signal and hence be incorrectly selected as the target. Thus for a given d' , P_c will decrease with increasing M . By the same logic, d' will increase with M for a given P_c .

BOX 6.1**SIGNAL DETECTION THEORY WITH PALAMEDES**

Palamedes contains a number of routines for performing signal detection computations. To understand the convention for the names of the routines, consider the following example: `PAL_SDT_2AFCmatchSample_DiffMod_PCToDP`. The routine is identifiable as part of the SDT package by the prefix `PAL_SDT`. The term `2AFCmatchSample` identifies the task as match-to-sample, and the number of forced-choice alternatives as 2. The generic acronym `AFC` is used in all routines. The term `DiffMod` specifies the particular model for the computation; in this case it means a differencing model. By “model” we mean the particular strategy that the observer is assumed to adopt when performing the task. This is only relevant to those tasks for which there is more than one possible strategy. Finally, `PCToDP` specifies the actual computation, in this case the conversion of P_c to d' . This last term is invariably of the form `XtoY`, where `X` is the principle input argument(s) and `Y` the principle output argument(s). However, the routine may require additional input arguments and may produce additional outputs. The abbreviations used for the `XtoY` term are `PC` for P_c , `DP` for d' , `SL` for stimulus level (or stimulus intensity or amplitude), `PHF` for proportion hits and proportion false alarms, and `PH` for proportion hits. If the required input to a routine is either `PC` or `DP`, the data can be entered as a scalar, vector, or matrix. If the input is `PHF` the data must be an $m = 2$ matrix with a minimum of one row. One column is for the proportion of hits and the other is for the corresponding proportion of false alarms.

Although tables exist for converting P_c to d' for a range of M (Elliot, 1964; Macmillan and Creelman, 2005) the Palamedes routines, introduced in [Box 6.1](#) and described for standard M-AFC tasks in [Box 6.2](#), work for any value of M and are simple to implement.

6.2.3 Measuring d' for 1AFC Tasks

As discussed in Chapters 2 and 3, we use the abbreviation 1AFC for tasks where a single stimulus is presented on a trial and the observer makes a forced-choice decision about it. We also noted that 1AFC tasks that are not symmetric, such as the yes/no task, are particularly prone to bias. Remember that with the yes/no task, the observer is required to indicate on each trial whether the target stimulus is present or absent. If the observer adopts a loose criterion, there will be a bias toward responding “yes,” whereas a strict criterion will lead to a bias toward responding “no.” Both types of bias may occur irrespective of how sensitive the observer is to the stimulus. For this reason, SDT computes d' for 1AFC tasks differently from tasks that are assumed to be bias-free. Rather than use P_c , the responses from a 1AFC task are divided into two groups: the target-present trials in which the observer responds “yes”, i.e., correctly, and the target-absent trials in which the subject responds “yes” i.e., incorrectly. The former responses are commonly termed “hits” and denoted by pH while the latter responses

BOX 6.2**STANDARD AFC ROUTINES**

The routines for standard M -AFC tasks assume that the observer is not biased toward responding to any one alternative/interval more than any other. There are two routines for standard 2AFC tasks: `PAL_SDT_2AFC_DPttoPC` for converting d' 's to P_{cs} and its inverse `PAL_SDT_2AFC_PCToDP`. To convert an array of P_{cs} to d' 's, type and execute the following:

```
>> DP = PAL_SDT_2AFC_PCToDP([0.55 0.65 0.75 0.85 0.95])
```

The array returned is

```
DP =
0.1777 0.5449 0.9539 1.4657 2.3262.
```

Now try the inverse routine with your own choice of d' inputs.

The corresponding routines for an M -AFC task, where M can take on any number greater than 1, are `PAL_SDT_MAFC_DPttoPC` and `PAL_SDT_MAFC_PCToDP`. These routines takes two arguments: the first is the array of the measure to be converted (d' or P_c) and the second is the value of M . Try filling a vector named `PC` with an array of P_{cs} as follows:

```
>> PC = [.3:.1:9];
```

To convert the array to d' 's for, say, a 3AFC task, type and execute

```
>> DP = PAL_SDT_MAFC_PCToDP(PC,3)
```

The array returned is

```
DP =
-0.1207 0.2288 0.8852 1.6524 2.2302.
```

Note that the first value in the array is negative. d' is 0 when performance is at chance, which for the standard 3AFC task is 0.33, so any P_c below 0.33 will produce a negative d' . Try repeating the above with $M = 4$. Note that the first value is now positive, since chance level for a 4AFC task is 0.25. If one sets M to 2 (chance = 0.5) the first two d' 's are negative. One can see from these examples that increasing M for a given P_c increases d' . This is because as M increases so too does the chance that one of the nontarget intervals/locations will contain a signal that is by chance greater than that in the target interval/location. In other words, for a given P_c , observer sensitivity is computed to be higher if the task has a large compared to a small M .

Try also converting an array of d' 's to P_{cs} using `PAL_SDT_MAFC_DPttoPC`. Note that increasing M for a given d' this time decreases P_c because the more possible target intervals/locations, the more likely one of them, by chance, will contain a signal greater than that in the target interval/location.

are termed “false alarms” and denoted by pF . The overall P_c is given by $[pH + (1 - pF)]/2$; the term $1 - pF$ gives the proportion of target-absent trials in which the observer responds “no,” i.e., correctly. The two measures, pH and pF , can be used not only to calculate d' but also the bias toward responding “yes” or “no.” **Box 6.3** shows how to calculate d' and two measures of bias, $\ln\beta$ and C , using Palamedes. The theory behind the calculations is explained in [Section B](#).

The measures of bias, or “criteria,” range from negative to positive, with negative values indicating a bias toward “yes” and positive values a bias toward “no.” **Table 6.2** gives pH and pF for a d' of 2 for various levels of the criterion measure C . Note that as C increases (loose to strict criterion) both pH and pF decreases. **Figure 6.2(a)** shows the relationship between pH and pF as a function of C for three values of d' . As one travels along each of the curves from left to right, C decreases, resulting in an increase in both pH and pF . The relationship between pH and pF is known as a receiver operating characteristic, or

TABLE 6.2 Proportion hits pH and proportion false alarms pF for various levels of the criterion measure C , for a d' of 2

C	pH	pF
-1	0.98	0.5
-0.5	0.93	0.31
0	0.84	0.16
0.5	0.69	0.067
1	0.5	0.023

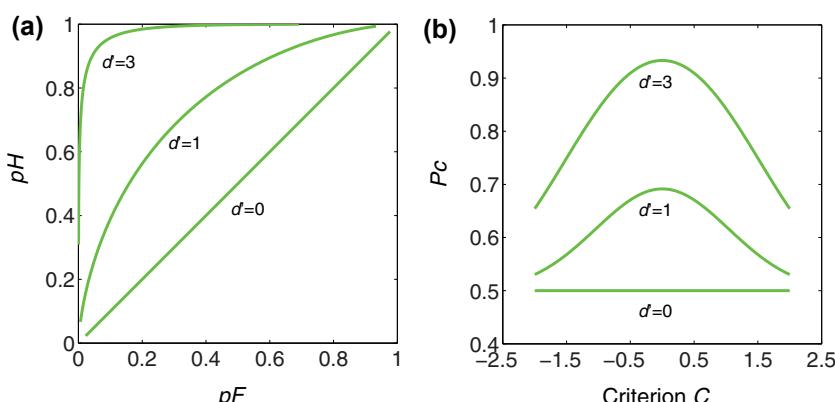


FIGURE 6.2 (a) Hypothetical ROCs for three d' s. Note that pH (proportion of hits) is plotted against pF (proportion of false alarms). As the criterion C decreases, one moves along each curve from left to right. (b) The effect of changing C on P_c (proportion correct) for the same d' s as in (a). Note that all three curves peak when $C = 0$.

ROC. The ROC in [Figure 6.2\(a\)](#) is hypothetical. However, ROCs can be generated from experiments in which the responses are not binary options such as “yes” or “no,” but ratings, for example 1 to 5, of how confident the observer is that the target is present. Measuring ROC curves for 1AFC tasks is the topic of the next section.

[Figure 6.2\(b\)](#) reveals why P_c is not a good measure of performance when there is observer bias. Assuming that the “true” observer’s sensitivity is given by d' , one can see that P_c varies considerably with criterion C. Only if there is no bias ($C = 0$) is P_c a valid measure of performance. A zero-bias assumption may sometimes be reasonable with “symmetric” 1AFC tasks, such as the 1AFC orientation discrimination experiment discussed in Chapter 3. However, some researchers argue that even with symmetric 1AFC tasks the data should be analyzed under the presumption that bias might have occurred. Then, if it turns out there is no bias nothing is lost, but if bias is found to have occurred it is taken into account—a win-win situation!

How, then, do we convert the responses from the symmetric 1AFC orientation discrimination experiment into pH and pF ? The answer is to classify the responses in a way analogous to that of the yes/no experiment. This means classifying a “left-oblique” response as a “hit” when the stimulus is left-oblique and as a “false alarm” when the stimulus is right-oblique. pH is then the proportion of “left-oblique” responses for the left-oblique stimuli, and pF is the proportion of “left-oblique” responses for the right-oblique stimuli. Note that the two measures pH and pF defined in this way are sufficient to describe all the responses in the experiment, i.e., including the “right-oblique” responses. The proportion of times the observer responds “right-oblique” is $1 - pH$ for the left-oblique stimulus trials and $1 - pF$ for the right-oblique stimulus trials. Note also that as with yes/no, overall P_c is given by $[pH + (1 - pF)]/2$. Thus, if the observer in the orientation discrimination experiment is biased toward responding “left-oblique,” both pH and pF , as defined above, will tend to be relatively high. By comparing the two measures in the same way as with the yes/no task, any bias can be taken into account and a valid measure of sensitivity calculated.

The Palamedes routines for converting pH and pF to d' and to measures of bias, and vice versa, are described in [Boxes 6.3 and 6.4](#).

BOX 6.3

1 - AFC ROUTINES

The Palamedes routine that converts pH and pF to d' , as well as to two measures of bias, is `PAL_SDT_1AFC_PHFtoDP`. The input argument can either be a prenamed $m = 2$ matrix of pH and pF values or the raw values themselves. There are four output arguments: d' , two measures of bias termed C and $\ln\beta$, and overall P_c .

Suppose we want to input just a single pair of raw pH and pF values. Type and execute the following, and remember to place the square brackets around the two values so that they are entered as a matrix:

```
>> [dp C lnB Pc] = PAL_SDT_1AFC_PHFtoDP([0.6 0.1])
```

Continued

BOX 6.3 (*cont'd*)

The output should be

```
dp =
1.5349
C =
0.5141
lnB =
0.7891
Pc =
0.7500
```

Criterion C can range from negative to positive, with negative values indicating a bias toward "yes" and positive values a bias toward "no." The criterion measure $\ln\beta$ shows the same pattern as C . The positive values of $C = 0.51$ and $\ln\beta = 0.78$ in the above example are indicative of a relatively strict criterion, in other words a bias toward responding "no."

To explore the relationship between d' , bias, pH , and pF , one can also use the reverse routine `PAL_SDT_1AFC_DPToPHF`. For example, create a vector named `dprime` filled with a 5×5 array of 2s and a vector named `criterion` with values $-1, -0.5, 0, 0.5$, and 1 . Then type and execute

```
>> pHF = PAL_SDT_1AFC_DPToPHF(dprime, criterion)
```

The output should be

```
pHF =
0.9772 0.5000
0.9332 0.3085
0.8413 0.1587
0.6915 0.0668
0.5000 0.0228
```

The first column gives pH and the second pF . Note that as C increases (loose to strict criterion) both the number of hits and the number of false alarms decreases.

6.2.4 Performing a Rating Scale Experiment with 1AFC

The previous section outlined how in theory an ROC curve can be generated from d' with various levels of criterion C . However, in practice we will want to estimate d' from data, and for this we will need to measure both pH and pF for various C , plot the results, and estimate d' . How do we do this? One popular method is the rating scale experiment. In a 1AFC rating scale experiment the observer is presented with a stimulus on each trial, with a 50% chance of

BOX 6.4

1AFC DEMONSTRATION SCRIPT

The script `PAL_SDT_1AFC_PHFtoDP_Demo` demonstrates how the routines for 1AFC tasks can be incorporated into a program that generates a more user-friendly output of d' and criterion measures. When executed, the program prompts you as follows:

```
Enter a matrix of proportion Hits and False Alarms
```

You must enter arrays of raw values. An example input matrix of pH and pF values would be:

```
[0.6 0.2; 0.7 0.2; 0.8 0.2]
```

The output should be:

pH	pF	Dprime	propCorr	Crit C	lnBeta
0.6000	0.2000	1.0950	0.7000	0.2941	0.3221
0.7000	0.2000	1.3660	0.7500	0.1586	0.2167
0.8000	0.2000	1.6832	0.8000	-0.0000	-0.0000

The inverse script `PAL_SDT_1AFC_DPtоНPHF_Demo` works similarly. You are prompted for two vectors of numbers. Try entering the following values, then execute:

```
Enter a vector of Dprime values [1 2 3]
```

```
Enter a vector of Criterion C values [0.5 0.5 0.5]
```

The output should be:

dprime	critC	pH	pF	pCorr
1.0000	0.5000	0.5000	0.1587	0.6707
2.0000	0.5000	0.6915	0.0668	0.8123
3.0000	0.5000	0.8413	0.0228	0.9093

it containing a signal, and is required to give a rating as to how certain they are that they perceived a signal. To illustrate how to generate an ROC curve from just such an experiment we have used rating-scale data from [McNicol \(2004\)](#). McNicol provides many useful details about the choices of signal and noise trials, the number and type of ratings used, etc. for this type of experiment.

In the experiment there are a total of 288 signal and 288 noise trials. On each trial the subject indicates their confidence in perceiving the signal, using a rating of 1–5, as follows: 1 = high certainty signal; 2 = fairly sure there was signal; 3 = equally likely to be signal or no signal; 4 = fairly sure there was no signal; and 5 = high certainty there was no signal. With the data collected, the first thing is to separate the signal S and noise N trials into

two groups, then for each group add up the number of trials producing each rating of response. The S and N rows in [Table 6.3](#) show the results. The next step is to convert these numbers into cumulative values. Why? Suppose we had performed the same 1AFC task but observers had been instructed to adopt the strictest of criteria, i.e., they would only respond “yes” when very certain a signal was present. This would have resulted in 159 hits and 2 false alarms. Suppose instead that the observer had adopted the single criterion corresponding to a rating of 2. Now the total number of hits would be $159 + 41 = 200$ and the total number of false alarms $2 + 3 = 5$. By the same argument, a criterion of 3 would result in $159 + 41 + 19 = 219$ hits and $2 + 3 + 21 = 26$ false alarms, and so on. These values are shown in the rows H_{cum} and F_{cum} (cum stands for cumulative). The final step is to convert H_{cum} and F_{cum} into proportions of hits pH and false alarms pF —these are the last two rows of [Table 6.3](#). We are now in a position to plot our ROC curve!

One can see from [Table 6.3](#) that instead of having just one pair of pH and pF values for deriving d' we now have four. We can now find the value of d' that best fits those values, using a maximum likelihood criterion. The round symbols in [Figure 6.3](#) show the proportion of hits pH plotted against the proportion of false alarms pF for the data in [Table 6.3](#). The continuous green curve is the best-fitting curve according to a maximum likelihood criterion, and the estimated value of d' is 1.76 (see legend). The corresponding value of R shown in the legend gives the ratio of signal-to-noise standard deviations, an important parameter that can be estimated from an ROC curve. The figure shows two other sets of data and corresponding estimates of d' and R . On the right the data have been replotted in z or standard deviation units, based on the relation $d' = z(pH) - z(pF)$ (the theoretical basis of this relation is explained in [Section B](#)). Note that when plotted in z units the data tend to fall along straight lines. [Box 6.5](#) describes the Palamedes demo script that produces [Figure 6.3](#).

6.2.5 Measuring d' for 2AFC Tasks with Observer Bias

Although the inherent symmetry of 2AFC tasks makes them less susceptible to bias than the yes/no task, biases can still occur. In this instance, the bias takes the form of responding

TABLE 6.3 Data from a 1AFC rating scale experiment

Rating	1	2	3	4	5	Total
S	159	41	19	37	32	288
N	2	3	21	80	182	288
H_{cum}	159	200	219	256	288	
F_{cum}	2	5	26	106	288	
pH	0.55	0.69	0.76	0.89	1.0	
pF	0.01	0.02	0.09	0.37	1.0	

S = signal-present trials with “yes” responses; N = signal-absent trials with “yes” responses; Cum = cumulative; H = Hits; F = false alarms; pH = proportion hits; pF = proportion false alarms.

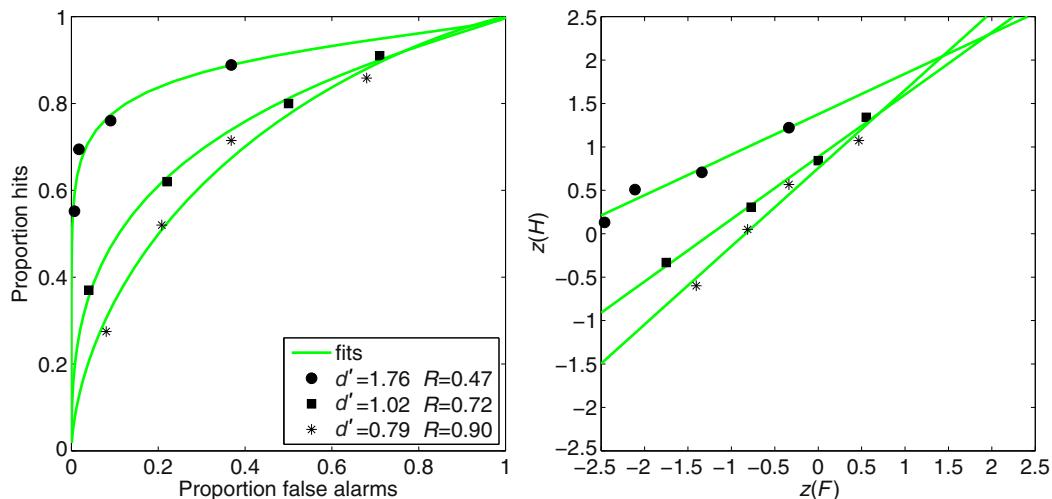


FIGURE 6.3 ROC curves and analyses for 1AFC rating-scale experiments. Left: proportion hits plotted against proportion false alarms. Right: data converted in z units. See text for details.

BOX 6.5

RATING SCALE DATA DEMONSTRATION SCRIPT

The demonstration script `PAL_SDT_ROCML_Demo` outputs Figure 6.3. When you run the script you will be prompted to enter the numbers of simulations for the standard errors and goodness-of-fits: 400 is a recommended number for both measures.

more to one alternative/interval than the other, and as with other types of bias if it occurs P_c becomes an invalid measure of sensitivity. As with symmetric 1AFC tasks, some researchers prefer not to hedge their bets with 2AFC and analyze the data on the presumption that bias might have occurred.

To take into account bias in 2AFC tasks, the observer's responses need to be classified as hits and false alarms, as with the symmetric 1AFC task. Let the response be "1" or "2," depending on the alternative perceived to contain the target. A "1" response is then designated as a "hit" if the target is present in the first alternative and a "false alarm" if present in the second alternative. Thus pH is calculated as the proportion of "1" responses for first-alternative targets and pF the proportion of "1" responses for second-alternative targets. Note that, as with 1AFC tasks, the measures pH and pF defined in this way are sufficient to describe fully the pattern of responses. Thus $1 - pH$ is the proportion of "2" responses for first-alternative targets and $1 - pF$ the proportion of "2" responses for second-alternative targets. Note also that, as with 1AFC tasks, overall P_c is given by $[pH + (1 - pF)]/2$. The Palamedes routines for biased standard 2AFC tasks are described in Box 6.6.

BOX 6.6**ROUTINES FOR BIASED STANDARD 2AFC TASKS**

Palamedes has two routines for a 2AFC task when the input arguments are proportion hits pH and proportion false-alarm pF : `PAL_SDT_2AFC_DPtоСPHF` and `PAL_SDT_2AFC_PHFtoDP`. The input and output arguments correspond to those for the 1AFC routines. Remember that one can also use `PAL_SDT_2AFC_PCToDP` and `PAL_SDT_2AFC_DPtоСPC` for 2AFC tasks but only if one is assuming that the observer is unbiased.

What is the expected relationship between performance in a 1AFC and 2AFC task? One can use `PAL_SDT_1AFC_PHFtoDP` and `PAL_SDT_2AFC_PHFtoDP` to find out. Try the following. Input the same pair of pH and pF values and the same value of the criterion to both routines. Take the ratio of the resulting 1AFC to 2AFC d' s. The result should be $\sqrt{2}$. The $\sqrt{2}$ relationship between d' s for 1AFC and 2AFC is often emphasized in expositions of SDT, but one must be careful with its interpretation. It is tempting to suppose that if one performed a 1AFC task and a 2AFC task using the same stimulus magnitudes, the computed d' s would have a ratio of $\sqrt{2}$. In fact, the d' s would likely be very similar. Remember that d' is a measure of sensitivity that is ostensibly independent of the method used to obtain it—although be reminded of the cautionary note from [Yeshurun et al. \(2008\)](#). The likely difference between the two tasks will be in P_c , not d' . As [Figure 6.4](#) demonstrates, the same d' predicts different P_c s for 1AFC and 2AFC. Put another way, observers will typically find a 1AFC task more difficult than a 2AFC task for the same stimulus magnitudes. This is because there is more information in each 2AFC compared to 1AFC trial.

6.2.6 Measuring d' for Same-Different Tasks

In Chapters 2 and 3 we described the class of psychophysical task termed “same-different.” One reason for using same-different tasks is that the observer is not required to know the basis on which the discriminands differ. There are two main varieties of same-different task. In the 1AFC version only one pair, Same or Different, is presented on a trial, and the observer has to decide “same” or “different.” Each pair of stimuli can be presented either together on the display or in temporal order. In the 2AFC version the Same and Different pairs are both presented in the same trial (either together on the display or in temporal order), and the observer chooses the alternative/interval containing the Different (or the Same) pair. The 2AFC same-different task is probably the more popular of the two versions for vision experiments, because it is less prone to bias.

For same-different tasks where only one pair, Same or Different, is presented in a trial, [Macmillan and Creelman \(2005\)](#) argue that observers typically adopt one of two strategies: the “independent observation” or “differencing” strategy (Macmillan and Creelman refer to the 1AFC same-different task as 2IAX or AX, the first two symbols of the first abbreviation denoting the fact that the two stimuli are presented in different temporal intervals).

Suppose that in each experimental session there are only two stimuli: S_1 and S_2 . On each trial the observer is presented with one of four possible combinations: $\langle S_1S_1 \rangle$, $\langle S_2S_2 \rangle$, $\langle S_1S_2 \rangle$, or $\langle S_2S_1 \rangle$. Macmillan and Creelman argue that the most likely strategy in this scenario is that the observer assesses the likelihood that each stimulus in a pair is either S_1 or S_2 . The decision “different” is made when the joint likelihood of the pair being S_1 and S_2 exceeds the observer’s criterion. This is the independent observation strategy.

The differencing strategy for same-different is less optimal, but under certain circumstances more likely to be adopted. As with the strategy assumed for the 2AFC same-different task described above, the decision rule is based on the perceived difference between the two stimuli in each pair. The observer responds “different” when the absolute perceived difference between the two stimuli exceeds the criterion. According to Macmillan and Creelman, the differencing strategy is more likely to be adopted when many different stimuli are presented during a session, termed a “roving” experiment. For example, suppose that one wished to compare the detectability of four types of color manipulation applied to images of natural scenes. Suppose the four manipulations are shifts in the average color of the scene toward either red, green, blue, or yellow. On each trial observers are presented either with two identical natural scenes (the Same pair) or two versions of the same scene but with the color in one scene shifted towards one of the colors (the Different pair). It would be difficult for observers to independently assess the likelihood that each member of a pair had been subject to a particular color shift, because there are four possible color shifts. The more likely strategy in this situation is that observers assess the difference in color between the images in each pair and base their decision accordingly. **Box 6.7** describes the Palamedes routines for same-different tasks.

BOX 6.7

ROUTINES FOR SAME-DIFFERENT TASKS

The Palamedes routines for the 2AFC same-different task are `PAL_SDT_2AFCsameDiff_DPttoPC` and `PAL_SDT_2AFCsameDiff_PCToDP`. Both routines assume an unbiased observer who adopts the strategy of selecting the pair with the greater (or smaller) absolute perceived difference. The routines implement the equations in [Macmillan et al. \(1977\)](#) for a “4IAX” same-different task, where 4IAX denotes that the four stimuli are presented in temporal order, the typical scenario in an auditory experiment. Both of the Palamedes routines take a single argument (d' or P_c) and output a single argument (P_c or d'). The input arguments may be scalars, vectors, or matrices.

The Palamedes routines for the 1AFC same-different task assuming an independent observation model are `PAL_SDT_1AFCsameDiff_IndMod_PHFtoDP` and `PAL_SDT_1AFCsameDiff_IndMod_DPttoPHF`. The first routine takes two arguments, an $m = 2$ matrix of pHs and pFs , and outputs two arguments: d' and criterion C . The second routine performs the reverse operation. For example, try inputting the same matrix of pHs and pFs as for the basic 1AFC task described earlier, i.e.,

```
PHF = [0.6 0.2; 0.7 0.2; 0.8 0.2]
```

Continued

BOX 6.7 (*cont'd*)

then type and execute

```
>>[dp C] = PAL_SDT_1AFCsameDiff_IndMod_PHFtoDP(PHF)
```

The output should be three d' values and three criterion C values. Compare these values with those obtained using `PAL_SDT_1AFC_PHFtoDP`. Try also computing d' 's for a bias-free version of the same-different task by setting pF equal to $1 - pH$. You will see that the resulting d' 's under the independent observation model are the same as those for the 2AFC same-different task ($Pc = pH$), which assumes a differencing strategy.

For the 1AFC same-different task assuming a differencing model the routines are `PAL_SDT_1AFCsameDiff_DiffMod_PHFtoDP` and `PAL_SDT_1AFCsameDiff_DiffMod_DPtoPHF`. They are implemented in the same way as the routines for the independent observer model. However, they return a different measure of bias termed k (Macmillan and Creelman, 2005). Unlike C , k is not 0 when the observer is unbiased.

Ask yourself the following. For a given pH and pF , would you expect d' to be larger or smaller for the differencing compared to the independent observer model? Try various pH and pF combinations to test your predictions.

6.2.7 Measuring d' for Match-to-Sample Tasks

In a match-to-sample task, the observer is presented with a “Sample” stimulus followed by two or more “Match” stimuli, one of which is the same as the Sample—the one the observer must choose. Match-to-sample tasks are popular in animal research, research into children’s perception, and studies of cognitive vision (see Chapter 3). As with the same-different task, one advantage of match-to-sample over standard M -AFC is that the observer need not know the basis on which the discriminands differ. The minimum number of stimuli per trial in a match-to-sample task is three (one Sample; two Match), and this is the most popular design. With two Match stimuli the task is 2AFC, according to our naming system; Macmillan and Creelman (2005) refer to the task as ABX.

Macmillan and Creelman argue that for the ABX task, observers may adopt either an independent observation or a differencing strategy, the latter more likely in “roving” experiments where a variety of stimulus pairs are presented within a session. The independent observation strategy is analogous to that for the same-different task. The differencing strategy implies that the observer selects the Match that is perceived to be least different from the Sample.

Given that observers might be biased toward choosing one Match alternative over the other, it is recommended to base the calculation of d' on pH and pF rather than Pc , unless

BOX 6.8**ROUTINES FOR MATCH-TO-SAMPLE TASKS**

Palamedes provides eight routines for the 2AFC match-to-sample task:

```
PAL_SDT_2AFCmatchSample_DiffMod_PCToDP  
PAL_SDT_2AFCmatchSample_DiffMod_DPToPC  
PAL_SDT_2AFCmatchSample_DiffMod_PHFtoDP  
PAL_SDT_2AFCmatchSample_DiffMod_DPToPHF  
PAL_SDT_2AFCmatchSample_IndMod_PCToDP  
PAL_SDT_2AFCmatchSample_IndMod_DPToPC  
PAL_SDT_2AFCmatchSample_IndMod_PHFtoDP  
PAL_SDT_2AFCmatchSample_IndMod_DPToPHF
```

The routines use the same input and output arguments as the same-different routines. Given that observers might be biased toward choosing one match alternative over the other, it is recommended to use the routines that take pH and pF rather than Pc as arguments, unless there is a good reason to assume the observer is unbiased.

there is good reason to assume the observer is unbiased. The relevant Palamedes routines are described in [Box 6.8](#).

6.2.8 Measuring d' for M-AFC Oddity Tasks

In an oddity task, often termed an “odd-man-out” task, the observer is presented with an array of stimuli, all the same bar one, and chooses the one that is different—the “oddity.” As with the same-different and match-to-sample tasks, the observer in an oddity task does not need to know the basis upon which the stimuli differ. Probably the most popular form of the oddity task is the one using the minimum possible number of stimuli, 3, termed by some the “triangular” method. However, the principle extends to any M . One likely strategy in an oddity task is that observers select the alternative that is most different from the mean of all the alternatives, another instance of the differencing strategy. The independent observer strategy for the oddity task is to select as the target the alternative most likely to be either one or the other of the two types of stimulus presented. [Box 6.9](#) describes the Palamedes routines for oddity tasks.

6.2.9 Estimating Pc_{max} with Observer Bias

As we argued earlier, Pc is not a valid measure of performance for any of the procedures described so far if there is a significant amount of observer bias. However, even when there is bias, it is possible to obtain an estimate of the Pc that one would expect if the observer were

BOX 6.9**ROUTINES FOR ODDITY TASKS**

Palamedes provides the following routines for oddity tasks:

```
PAL_SDT_3AFCoddity_IndMod_DPttoPC
PAL_SDT_3AFCoddity_IndMod_PCToDP
PAL_SDT_MAFCoddity_IndMod_DPttoPC
PAL_SDT_MAFCoddity_IndMod_PCToDP
PAL_SDT_MAFCoddity_DiffMod_DPttoPC
PAL_SDT_MAFCoddity_DiffMod_PCToDP
```

The 3AFC routines only require the array of measures to be converted, whereas the M -AFC routines require the additional parameter M as input. The M -AFC routines are relatively slow, and the $_PCToDP$ especially slow, because they are implemented by Monte Carlo simulation using a very large number of trials. Results using the M -AFC differencing model routines can be compared to those provided by [Craven \(1992\)](#).

not biased. This is termed $P_{c_{max}}$ (or $P_{c_{unb}}$), the max subscript capturing the fact that P_c reaches a theoretical maximum when there is no bias (e.g., see [Figure 6.2\(b\)](#) for the 1AFC task). One can think of $P_{c_{max}}$ as an unbiased estimate of P_c . To obtain $P_{c_{max}}$ one first computes d' and C from pH and pF and then reverses the operation, this time setting C to 0. $P_{c_{max}}$ is then equal to pH . [Box 6.10](#) describes how to use the Palamedes routines for estimating $P_{c_{max}}$.

6.2.10 Comparing P_c s from d' s Across Different Tasks

[Figure 6.4](#) compares P_c s as a function of d' for four tasks modeled under SDT. Palamedes scripts are described in [Box 6.11](#).

6.2.11 Modeling Psychometric Functions with SDT

The above applications of SDT deal for the most part with individual values of d' and P_c . What then of whole psychometric functions of P_c ? In principle one could convert each value of P_c from the psychometric function into a d' and then proceed with whatever analysis one wished. That would be fine. However, one can do more than this: one can “fit” an SDT model to a psychometric function and estimate certain parameters from the fit that turn out to be very useful. Some of the uses of these parameters will be described in the next chapter that deals with the summation of multiple stimuli. For now we will establish the principle. To understand how SDT models can be fit to psychometric functions it is first necessary to consider the relationship between d' and the intensity of the stimulus. To recap, d' is a measure of the internal strength of a stimulus expressed in units of the internal noise standard

BOX 6.10

ESTIMATING $P_{c_{\max}}$

Estimating $P_{c_{\max}}$ is straightforward with Palamedes, provided one has available a measure of the criterion that is 0 when the observer is unbiased, as with the routines that compute the criterion measure C . To obtain $P_{c_{\max}}$ one inputs pH and pF into the relevant routine (i.e., one ending in `_PHFtoDP`) to obtain d' and a measure of C and then use the reverse routine (the same routine ending in `_DPToPHF`) to convert back to pH and pF , using as the input argument a 0 value for C . $P_{c_{\max}}$ is then equal to the output pH .

Take the following example. Suppose you want to estimate $P_{c_{\max}}$ for a 2AFC match-to-sample task assuming a differencing strategy. Let $pH = 0.8$ and $pF = 0.6$. One can glean from these values that the observer is biased toward the alternative for which a correct response has been classified as a “hit,” since the number of false alarms exceeds $1 - 0.8$ i.e., 0.2. Recall also that P_c is given by $[pH + (1 - pF)]/2$, which for this example is 0.6. If the values of pH and pF are input to `PAL_SDT_2AFCmatchSample_DiffMod_PHFtoDP`, the routine returns a d' of 1.2137 and a criterion C of -0.5475. If one now inputs the same d' to `PAL_SDT_2AFCmatchSample_DiffMod_PHFtoDP`, but with C set to 0, the outputs are $pH = 0.6157$ and $pF = 0.3843$. Thus, $P_{c_{\max}}$ is 0.6157. $P_{c_{\max}}$ is only slightly higher than the actual P_c because the bias in this example is not particularly strong.

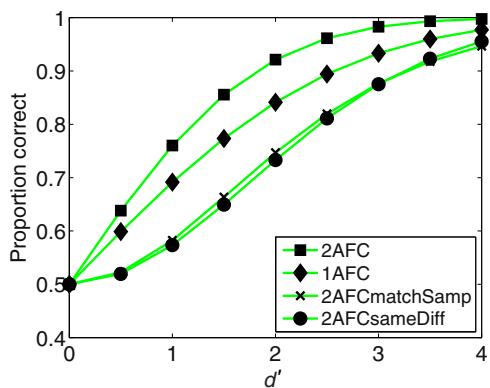


FIGURE 6.4 Comparison of P_c s as a function of d' for various tasks. The match-to-sample and same-different tasks assume a differencing strategy.

deviation and therefore carries no information about the units of the stimulus itself. In other words, d' is agnostic to whether the stimulus is expressed in units of contrast, depth, velocity, etc. It follows that there must be a scaling factor that relates every d' to the unit of stimulus intensity it embodies. However, there is more to the relationship between d' and stimulus intensity than a simple scaling factor. As stimulus intensity grows, so does d' , but the relationship between the two, the transducer function, is not always linear; in fact, more often

BOX 6.11**COMPARING d' AND P_c ACROSS DIFFERENT TASKS**

Two scripts are provided by Palamedes that demonstrate the differences between the computed d 's and P_c s for a variety of tasks: `PAL_SDT_DPtоСCcomparison_Demo` and `PAL_SDT_PCToDPcomparison_Demo`. The tasks compared are 1AFC, standard 2AFC, 2AFC same-different, and 2AFC match-to-sample. The standard 2AFC, same-different, and match-to-sample tasks assume a differencing strategy, and all tasks assume an unbiased observer.

Therefore, for the 1AFC tasks, criterion C is set to 0 to produce an optimal P_c . The scripts prompt you either for d 's or P_c s. Try the first program:

```
>>PAL_SDT_DPtоСCcomparison_Demo

Enter a vector of Dprime values and enter

[0:.5:4]
```

The output should look like this:

-----Proportion correct-----				
dprime	1AFC	2AFC	2AFCsameDiff	2AFCmatchSamp
0	0.5000	0.5000	0.5000	0.5000
0.5000	0.5987	0.6382	0.5195	0.5223
1.0000	0.6915	0.7602	0.5733	0.5825
1.5000	0.7734	0.8556	0.6495	0.6635
2.0000	0.8413	0.9214	0.7330	0.7468
2.5000	0.8944	0.9615	0.8110	0.8196
3.0000	0.9332	0.9831	0.8753	0.8765
3.5000	0.9599	0.9933	0.9231	0.9178
4.0000	0.9772	0.9977	0.9555	0.9467

and a graph similar to [Figure 6.4](#) will be plotted.

than not it will be nonlinear. For example, there is abundant evidence that for many stimuli around detection threshold, e.g., for contrast, the transducer function is an accelerating function of stimulus intensity ([Tanner and Swets, 1954](#); [Legge and Foley, 1980](#); [Heeger, 1991](#); [Meese and Summers, 2009](#); [Meese and Summers, 2012](#)). These considerations lead to the following formulation that describes the relationship between stimulus intensity x and d' :

$$d' = (gx)^\tau \quad (6.1)$$

where g is a scaling factor and τ (pronounced “tau”) is the exponent on the transducer function. The idea of incorporating a separate parameter g for the scaling of stimulus intensity when modeling SDT was suggested to us by Alex Baldwin (the scientist not the actor!).

Once we have identified the appropriate SDT function for the task, we can define a psychometric function of the form:

$$P_c = F_{SDT}(x; g, \tau, M) \quad (6.2)$$

where $F_{SDT}(x; g, \tau, M)$ is the SDT function in question, with input parameters g , τ , and for some functions M , the number of alternatives in the forced-choice task. Palamedes routines for converting x to P_c and back, given g and τ , are described in Box 6.12.

Figure 6.5 shows an example function fitted to hypothetical data from a standard 2AFC task. The function has been fitted using a maximum likelihood criterion and the two parameters estimated from the fit are g and τ . Table 6.4 shows these estimates together with their bootstrap errors. The data have also been fitted with a Weibull function (see Chapter 4), and the table provides the Weibull estimates of threshold α (approximately 0.8 proportion correct level) and slope β . On the right of the figure is shown the relationship between d' and

BOX 6.12

CONVERTING STIMULUS INTENSITY x TO P_c AND BACK, GIVEN g AND τ

Palamedes has two routines that convert stimulus intensity x to proportion correct P_c , assuming an SDT model, stimulus scaling factor g , and transducer exponent τ . These are `PAL_SDT_SLtoPC` and its inverse `PAL_SDT_PCToSL`. The letters `SL` in the routine name stand for stimulus level (i.e., x). Taking the first of these, the routine takes as input x (which can be a scalar, vector, or matrix), g , τ , the SDT routine that models the task, and M , the number of alternatives in the forced-choice task. For some SDT routines M is implicit, so for these M should be set to empty, i.e., `[]`.

In the following example P_c is calculated for a standard 3AFC task from a set of x (1...5), with $g = 0.5$ and $\tau = 1.5$. Type and execute

```
>> Pc = PAL_SDT_SLtoPC([1 2 3 4 5],0.5,1.5,@PAL_SDT_MAFc_DPtOpc,3)

Pc =
0.4377 0.6337 0.8371 0.9586 0.9950
```

The following example performs the inverse operation, this time for a 3AFC oddity task under the independent observation model, for a single P_c of 0.75 and for the same values of g and τ . Type and execute

```
>> x = PAL_SDT_PCToSL(0.75,0.5,1.5,@PAL_SDT_3AFCoddity_IndMod_PCToDP,[])

x =
3.5245
```

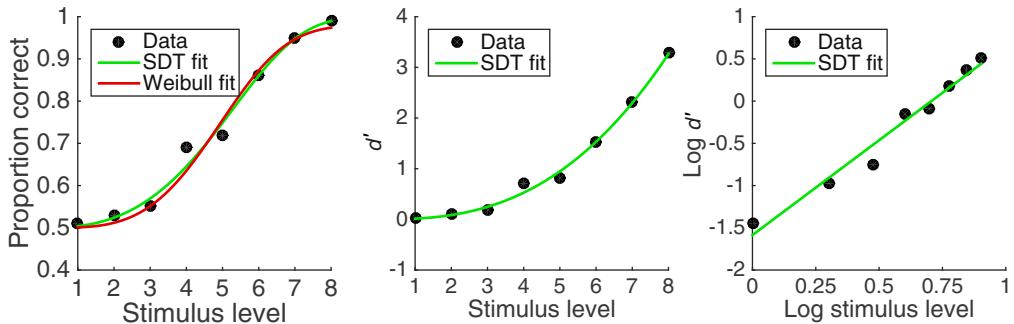


FIGURE 6.5 Left: proportion correct as a function of stimulus level for a standard 2AFC task. The continuous green curve is the fitted SDT model; the red curve is the Weibull fit. Middle: d' plotted against stimulus level on a linear–linear plot, revealing directly the accelerating nonlinearity. Right: same data converted into logarithmic units—the slope of the straight line fit is an estimate of τ .

TABLE 6.4 Parameter estimates for the data in Figure 6.5. The SDT estimates are of the scaling factor g and transducer exponent τ . The direct fit (left columns) corresponds to the fit shown in the middle graph, while the straight line fit (middle columns) corresponds to the fit in the graph on the right. The Weibull fit estimates of α (approximately 0.8 proportion correct level) and slope β are also given (right columns). SES (standard errors) were obtained using bootstrap analysis. The p values are goodness-of-fit measures based on the likelihood ratio test, with the higher the p value the better the fit.

	SDT model direct fit		SDT model log fit		Weibull model	
Parameter	g	τ	g	τ	α	β
Mean	0.196	2.63	0.197	2.25	5.38	3.73
SE	0.0084	0.38			0.20	0.74
p value		0.97				0.74

stimulus intensity x . Note that the function is accelerating; this is precisely what is captured by $\tau > 1$. As one might expect, there is an intimate relationship between τ and β ; as a rough approximation, $\beta = 1.2 \tau$ (Strasburger, 2001; May and Solomon, 2013).

If one takes logarithms of both sides of Eqn (6.1) one obtains

$$\log d' = \tau \log x + \tau \log g \quad (6.3)$$

The last term is a constant so in theory the plot of $\log d'$ against $\log x$ will be a straight line with a slope of τ and an intercept of $\tau \log g$. This suggests an alternative method of estimating τ and g : convert each value of P_c from the psychometric function into a d' , plot $\log d'$ against $\log x$, and estimate the slope and intercept from the best-fitting straight line to the plot.

The graph on the right of Figure 6.5 shows the result. Table 6.4 gives the estimates of g and τ from both the “direct fit” (left graph) and the “log fit” (right graph) methods as well as estimates of Weibull threshold α and slope β . Box 6.13 describes the Palamedes demo script that produced Figure 6.5.

BOX 6.13**FITTING SDT PSYCHOMETRIC FUNCTIONS**

The script `PAL_SDT_PF_Demo` demonstrates the Palamedes routines for fitting data with SDT model psychometric functions and outputs a figure similar to [Figure 6.5](#). The script prompts the user for the number of simulations required for the bootstrap errors and goodness-of-fit tests. The script outputs the parameter estimates similar to those shown in [Table 6.4](#). The user can change the SDT model that is fitted to the data within the script.

6.3 SECTION B: THEORY**6.3.1 Relationship Between z -Scores and Probabilities**

To understand the theory behind the various calculations of d' we need to begin with some basics, and the basic that underpins all of SDT is the relationship between z -values and probabilities. [Figure 6.6](#) shows a “standardized” normal probability distribution—a normal distribution in which the abscissa is given in units of standard deviation, or z units. The ordinate in the graph is termed “probability density,” denoted by ϕ . Probability density values are not actual probabilities, but relative likelihoods, specifically derivatives or rates of change of probabilities. Thus, in order to convert z units, or rather intervals between z units, into probabilities, one has to integrate the values under the curve between pairs of z -values. If one integrates the curve between $-\infty$ and some value of z , the result is a value from a distribution termed the cumulative normal, denoted by Φ . The total area under the standard normal distribution is by definition unity, so the cumulative normal ranges from 0–1. In short the cumulative normal gives the probability that a random variable from a standardized normal distribution is less than or equal to z .

The equation for the standardized normal distribution is

$$\phi(z) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{z^2}{2}\right) \quad (6.4)$$

and for the cumulative normal

$$\Phi(z) = 0.5 + 0.5 \operatorname{erf}\left(\frac{z}{\sqrt{2}}\right) \quad (6.5)$$

where erf stands for the “error function”—this is the function that performs the integration. The two values of 0.5 in the equation convert the range to 0–1. The inverse of the cumulative normal converts Φ to a z -value and is given by

$$z(\Phi) = \sqrt{2} \operatorname{erfinv}(2\Phi - 1) \quad (6.6)$$

BOX 6.14***z*-SCORES AND *p*-VALUES**

Palamedes contains two routines, `PAL_ZtoP` and `PAL_PtoZ`, which implement Eqns (6.5) and (6.6), respectively. Apart from converting *z*-scores to *p*-values and vice versa, they can also be used for some situations to calculate d' values. The following example converts a vector with three proportion corrects into a vector of d' 's for an unbiased 2AFC task:

```
>>DP = PAL_PtoZ([0.5 0.7 0.8].*sqrt(2))
```

The output should be

```
DP =
0.5450 2.3245 NaN
```

Given that *z*-values are symmetric around 0, we can state two simple relationships:

$$1 - \Phi(z) = \Phi(-z) \quad (6.7)$$

and

$$-z(\Phi) = z(1 - \Phi) \quad (6.8)$$

Box 6.14 describes the Palamedes routines for converting *z*-scores to *p*-values and back again.

6.3.2 Calculation of d' for M-AFC

We begin by describing the theory behind the computation of d' for a standard *M*-AFC task, where *M* can be any value greater than 1 and where $M = N = m$. It is worth remembering that the calculations described in this section are based on two assumptions: the first that the observer is unbiased and the second that the internal responses to all the stimulus alternatives are normally distributed and of equal variance.

Figure 6.7 shows two standardized normal distributions. One represents the distribution of sensory magnitudes or internal responses to a “blank” interval or location, i.e., one without a target and denoted by “noise alone” or *N*. The other represents the distribution of sensory magnitudes to the interval/location containing the target, typically denoted in the SDT literature as “signal-plus-noise” or *S + N*, but here just *S*. Note, however, that *N* versus *S* is not the only scenario for which the present analysis is applicable. The principle also extends to the situation in which one interval/location contains stimulus *S*₁ while the remaining intervals/locations contain stimulus *S*₂.

Representing the sensory magnitudes of *N* and *S* as probability distributions means that on any trial the actual sensory magnitudes will be random samples from those distributions.

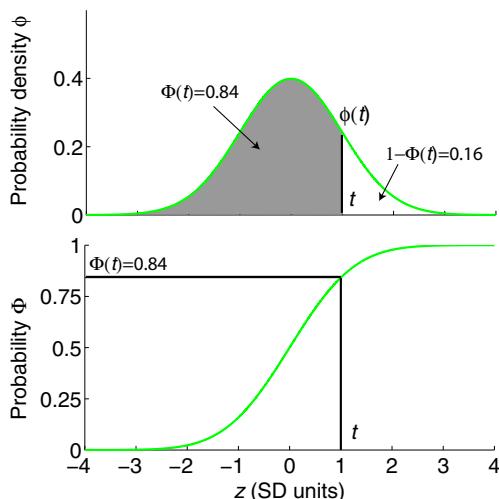


FIGURE 6.6 Relationship between z , probability density ϕ , and cumulative probability Φ . Top: standardized normal distribution. Bottom: the integral of the standardized normal distribution, or cumulative probability distribution. The value of z at point t is 1 in both graphs. In the top graph the height of the distribution at t is denoted by $\phi(t)$ and the area under the curve to the left of t (shown in gray), which has a value of 0.84, is denoted by $\Phi(t)$. The white area to the right of t , defined as $1 - \Phi(t)$ has a value 0.16. In the bottom graph $\Phi(t)$ is now a point on the ordinate.

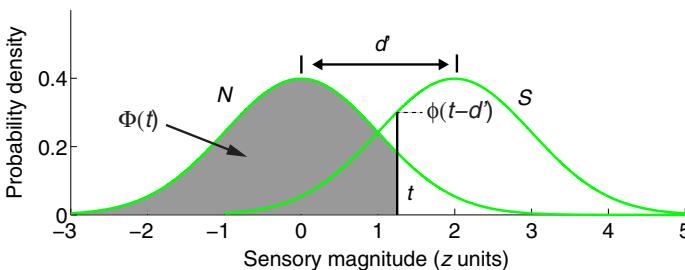


FIGURE 6.7 The parameters that form the basis of the calculation of P_c from d' . N = noise; S = signal-plus-noise; t is a random variable; $\Phi(t)$ is the gray area under the noise distribution; and $\phi(t - d')$ is the height of the signal distribution at t . See text for further details.

The relative probabilities of particular samples are given by the heights of the distributions at the sample points. The aim of the observer in the standard forced-choice task is to identify on each trial the alternative containing the target. Let us assume that the observer adopts what is intuitively the optimum strategy: select the alternative with the biggest signal. Try to imagine a strategy that would result in better performance. There isn't one. The rule employed by the observer for selecting the target is termed the "decision rule" and this decision rule is termed the MAX rule. The question then becomes: How well will the observer do, as measured by P_c , when adopting the MAX rule? If we make the two assumptions stated above, then the computation of d' turns out to be reasonably straightforward.

One can glean from [Figure 6.7](#) that when there is little overlap between the N and S distributions the observer will perform better than when there is a lot of overlap. The reason for this is that, as the N and S distributions draw closer together, there is an increasing likelihood that a sample drawn randomly from the N distribution will be greater in magnitude than a sample drawn randomly from the S distribution. Each time this happens the observer will make a mistake of adopting the MAX decision rule. If there were no overlap at all between the two distributions, the observer would never make an incorrect decision using this rule, while on the other hand if the distributions were superimposed the observer would perform at chance. Thus, the degree of overlap between the two distributions is the critical determinant of performance. And because the overlap is governed by two factors, first the separation of the two distributions and second their spread, or σ , one can see that the measure d' , which is the separation between the distributions expressed in units of σ , captures the discriminability of N and S . But how do we calculate the expected P_c , given d' and M ?

Suppose that on a given trial the target stimulus has a sensory magnitude given by t in the figure. Remember that t is a random sample, meaning that t will vary between trials, and the relative probability of t is the height of the distribution at t . The probability that t will be greater than a random sample from just one noise (N) location is given by the gray area to the left of t under the noise distribution. This is simply $\Phi(t)$, since we have (arbitrarily) centered the noise distribution at zero. However, we do not just wish to know the probability that t will be greater than a random sample from just one noise location, but from $M - 1$ noise locations. In other words, we want to know the probability that t will be greater than a random sample from noise location number one *and* noise location number two *and* noise location three *and* four and so on, up to $M - 1$. The “*and*” term here implies a joint probability, and if we assume that the samples from the different noise locations are independent, this is obtained simply by multiplying the individual probabilities. Since we are multiplying the same thing over again we simply raise the probability to the power of $M - 1$ and hence obtain $\Phi(t)^{M-1}$. However, this still only gives us the probability that one specific random sample from the signal distribution, t will be greater than all random samples from all $M - 1$ noise locations. To obtain the probability that a random sample t will be greater than random samples from $M - 1$ noise locations, which gives us our P_c , we need to integrate the above result across all possible values of t . We do this by multiplying $\Phi(t)^{M-1}$ by the height, or relative likelihood of t , which is given by $\phi(t - d')$ (the S distribution is offset from 0 by d'), and integrate over all values of t . Hence we have

$$P_c = \int_{-\infty}^{\infty} \phi(t - d') \Phi(t)^{M-1} dt \quad (6.9)$$

([Green and Swets, 1974](#); [Wickens, 2002](#)).

Normally, however, we want to convert a P_c into a d' not the other way round. How do we do this? [Equation \(6.9\)](#) is not easily invertible, which is a problem that arises with many SDT formula. In these situations the solution is to employ an iterative search method. Practical details are given in [Box 6.2](#).

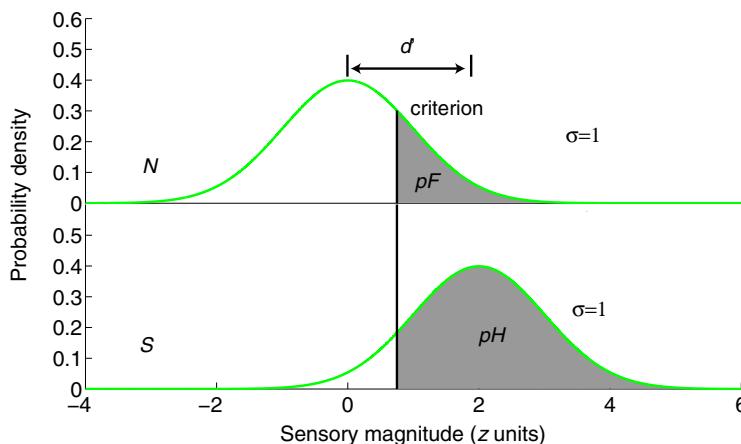


FIGURE 6.8 Distributions of sensory magnitude in response to both noise N and signal S in a 1AFC yes/no task. The vertical black line shows the position of the observer's criterion. Sensory magnitudes to the right of this line result in a "yes" response, while those to the left result in a "no" response. pH is the proportion of "hits," or correct "yes" responses, and pF is the proportion of "false alarms," i.e., incorrect "yes" responses. pH and pF are given by the gray areas to the right of the criterion line.

6.3.3 Calculation of d' and Measures of Bias for 1AFC Tasks

6.3.3.1 d' for 1AFC

Let us now consider the 1AFC task known as yes/no, a task that is particularly prone to bias. Adopting the same scheme for representing the distributions of sensory magnitudes as in the previous section for the standard M -AFC task, the situation is illustrated in Figure 6.8. This time, the N and S distributions are shown separately as the stimuli they represent are presented on separate trials. The gray areas to the right of the vertical criterion line represent sensory magnitudes that the observer deems large enough to warrant a "yes" response. Sensory magnitudes to the left of this line are "no" responses. The gray area in the lower S distribution gives the probability of target-present trials resulting in a "yes" response, i.e., the probability of hits or pH . The gray area in the upper N distribution gives the probability of "yes" responses in target-absent trials, i.e., the proportion of false alarms or pF . If we denote the position of the criterion line on the abscissa as c (see below), then one can see that

$$\begin{aligned} pF &= 1 - \Phi(c) \\ \text{or } pF &= \Phi(-c) \end{aligned}$$

and

$$\begin{aligned} pH &= 1 - \Phi(c - d') \\ \text{or } pH &= \Phi(-c + d') \end{aligned}$$

Converting pF and pH to z -values one obtains

$$z(pF) = -c$$

and

$$z(pH) = -c + d'$$

Combining these two equations and solving for d' gives

$$d' = z(pH) - z(pF) \quad (6.10)$$

6.3.3.2 Criterion C for 1AFC

In Figure 6.8 it can be seen that the criterion is measurable in z units, with a high z -value implying a strict criterion (few hits but few false alarms) and a low z -value implying a loose criterion (many hits but many false alarms). However, the actual criterion z -value depends on where the zero z -value is positioned, so a convention is needed to ensure that the criterion measure is comparable across conditions. The convention is to place the zero point midway between the N and S distributions, as shown in Figure 6.9.

With $z = 0$ centered midway between the two distributions, the criterion, denoted by C , is positioned in the noise distribution at

$$z(1 - pF) - d'/2$$

and in the signal-plus-noise distribution at

$$z(1 - pH) + d'/2$$

However, since $z(p) = z(1 - p)$, the two expressions can be rewritten as $-z(pF) - d'/2$ and $-z(pH) + d'/2$. Thus, the position of C can be defined in two ways:

$$C = -z(pF) + d'/2$$

and

$$C = -z(pH) - d'/2$$

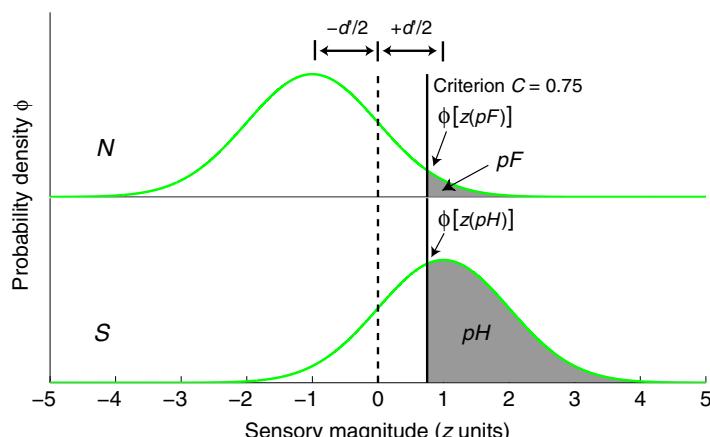


FIGURE 6.9 Method for calculating criterion C . Note that $z = 0$ is centered midway between the N and S distributions. See text for further details.

Adding the two equations together gives

$$C = -[z(pH) + z(pF)]/2 \quad (6.11)$$

(Macmillan and Creelman, 2005). Thus, criterion C can be estimated by converting pH and pF into z -values and then combining the two using Eqn (6.11). C can range from negative to positive, with negative values indicating a bias toward “yes” responses and positive values a bias toward “no” responses.

6.3.3.3 Criterion $\ln\beta$ for 1AFC

An alternative measure of the criterion is the natural logarithm of the ratio of the heights of the two distributions at C (Macmillan and Creelman, 2005). The heights at C are shown in Figure 6.9 as $\phi[z(pH)]$ and $\phi[z(pF)]$. Thus

$$\ln\beta = \ln \frac{\phi[z(pH)]}{\phi[z(pF)]} \quad (6.12)$$

Now $\phi[z(pH)]$ and $\phi[z(pF)]$ are given by

$$\phi[z(pH)] = \frac{1}{\sqrt{2\pi}} \exp \left[\frac{-\{-z(pH)^2\}}{2} \right] \quad (6.13)$$

and

$$\phi[z(pF)] = \frac{1}{\sqrt{2\pi}} \exp \left[\frac{-\{-z(pF)^2\}}{2} \right] \quad (6.14)$$

Taking natural logarithms of the two equations, i.e., $\ln\{\phi[z(pH)]\}$ and $\ln\{\phi[z(pF)]\}$ and then substituting the results into Eqn (6.12), a little algebra shows that

$$\ln\beta = \frac{[z(pF)^2 - z(pH)^2]}{2} \quad (6.15)$$

$\ln\beta$ behaves in the same way as C . The reason for this is that $\ln\beta = Cd'$. Readers can check the truth of this relationship using the equations above.

6.3.3.4 Criterion C' for 1AFC

A third measure of the criterion, C' , is the value of C expressed as a proportion of d' (Macmillan and Creelman, 2005), i.e.,

$$C' = C/d' = \frac{-[z(pH) + z(pF)]}{2[z(pH) - z(pF)]} \quad (6.16)$$

6.3.3.5 $P_{c_{max}}$ for 1AFC

In Section A we showed graphically that with a 1AFC task the optimum P_c , or $P_{c_{max}}$, is obtained when the observer is unbiased, i.e., when $C = 0$. It follows from Eqn (6.11) that

when $C = 0$, $z(pH) = -z(pF)$. Since $d' = z(pH) - z(pF)$ (Eqn (6.10)), simple algebra reveals that when $C = 0$, $d' = 2z(pH)$. Converting $z(pH)$ to $P_{c_{\max}}$ gives

$$P_{c_{\max}} = \Phi(d'/2) \quad (6.17)$$

The interested reader may wish to prove that P_c reaches a maximum when $C = 0$. One can determine if the observer is operating optimally in a 1AFC task by testing whether $pH = 1 - pF$. When performing optimally, pH is $P_{c_{\max}}$, and d' can be calculated as $2z(pH)$.

6.3.4 Calculation of d' for Unbiased and Biased 2AFC Tasks

In the first section of Section B we derived the formula for calculating d' from P_c for an unbiased standard M -AFC task (Eqn (6.9)). This formula can be used to calculate d' for the standard 2AFC task ($M = 2$), assuming that the observer is unbiased. In the following sections we describe a simpler method for calculating d' for an unbiased 2AFC task and show how both d' and measures of bias can be calculated for observer-biased 2AFC tasks.

6.3.4.1 Alternative Calculation of d' for Unbiased 2AFC

With the standard 2AFC procedure, the N and S stimuli are presented together in a trial as two alternatives. Remember that the decision rule is to choose the alternative in which the internal signal is biggest - the MAX rule. If the observer adopts this rule, trials in which the differences between the S and N samples are positive will result in a correct decision. The distribution of differences between random samples from two equal-variance standard normal distributions, one with mean 0 and the other d' , is itself a normal distribution with a mean of d' but a variance not of unity but 2, or a standard deviation σ of $\sqrt{2}$. This follows from the variance sum law, which states that the variance of the sum, or of the difference, between two uncorrelated random variables is the sum of the variances of the two variables. Thus, if the two distributions each have a σ of 1, the σ of the difference between the two distributions is $\sqrt{(1^2 + 1^2)} = \sqrt{2}$. The $S - N$ difference distribution is illustrated in the lower panel of Figure 6.10. Note that in this graph the abscissa is in z units that have been normalized to the σ s of the N and S distributions, not to the σ of their difference. The proportion correct for 2AFC is thus given by the gray area in the lower panel to the right of 0. This is

$$P_c = \Phi\left[\frac{d'}{\sqrt{2}}\right] \quad (6.18)$$

(Wickens, 2002; Macmillan and Creelman, 2005; McNicol, 2004). Equation (6.18) converts a z -value of $d'/\sqrt{2}$ to a Φ value. However, in most instances we want to obtain d' from P_c , so for this we use the inverse equation:

$$d' = z(P_c)\sqrt{2} \quad (6.19)$$

6.3.4.2 d' for Biased 2AFC

Consider the situation in which the two alternatives are presented sequentially, i.e., 2IFC. Figure 6.11 plots the distribution of differences in sensory magnitude between those in the

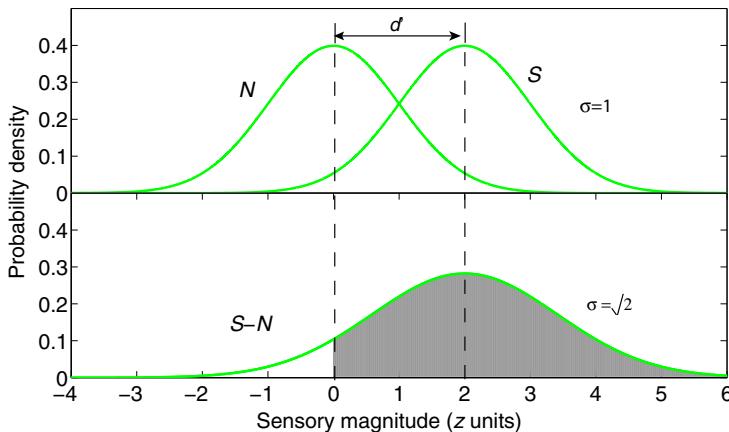


FIGURE 6.10 Graphical illustration of how d' can be calculated for an unbiased 2AFC task. Top: distributions of noise N and signal S separated by d' . Bottom: distribution of the difference between the two distributions: $S - N$. Note the different σ s for the upper and lower distributions. The z values along the abscissa are normalized to the σ of the two distributions in the top but not the bottom panel. See text for further details.

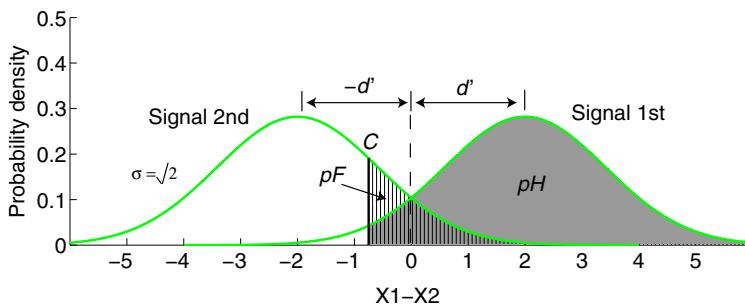


FIGURE 6.11 Relationship between d' , C , pH , and pF in a biased 2AFC task. Each plot gives the distribution of differences between the sensory magnitudes in the first ($X1$) and second ($X2$) alternatives/intervals. If the signal is in the first alternative/interval the distribution is the one shown on the right; if it is in the second interval the distribution is the one shown on the left.

first interval ($X1$) and those in the second interval ($X2$), i.e., the distribution of $X1-X2$. Note that there are now two distributions, not one for S and the other N , but one for signal present in the first interval and one for signal present in the second interval. The two distributions will be separated by $2d'$ and have σ s of $\sqrt{2}$ (see above). If the observer is biased toward responding to one interval more than the other, then the criterion C will be nonzero. The observer's decision rule is "1" (first interval) if $X1-X2 > C$ and "2" (second interval) if $X1-X2 < C$. As explained in [Section A](#), the key to calculating d' for a biased 2AFC task is to classify the responses in terms of hits and false alarms, where a "1" response is scored as a hit when the signal is in the first interval and a false alarm when the signal is in the second interval.

One can see from [Figure 6.11](#) that

$$z(pH) = \frac{(d' - C)}{\sqrt{2}} \quad (6.20)$$

and

$$z(pF) = \frac{(-d' - C)}{\sqrt{2}} \quad (6.21)$$

Combining the two equations and solving for d' gives

$$d' = \frac{[z(pH) - z(pF)]}{\sqrt{2}} \quad (6.22)$$

and solving for C gives

$$C = \frac{[-z(pH) + z(pF)]}{\sqrt{2}} \quad (6.23)$$

The criterion measure $\ln\beta$ defined in [Eqn \(6.12\)](#) uses ϕ s that refer to the heights of $z(pH)$ and $z(pF)$ in the standard normal distribution, i.e., the distributions in the upper panel of [Figure 6.10](#), not to the heights of the difference distributions in [Figure 6.11](#). The calculation of $\ln\beta$ for the 2AFC task is thus identical to that for the 1AFC task (we thank Mark Georgeson for pointing this out) and is hence given by [Eqn \(6.15\)](#). [Box 6.6](#) describes Palamedes routines that embody these equations.

6.3.4.3 $P_{c_{\max}}$ for Biased 2AFC

From [Eqn \(6.23\)](#), if $C = 0$, then $z(pH) = -z(pF)$. Combining this result with [Eqn \(6.22\)](#) reveals that when $C = 0$, $d'/\sqrt{2} = 2z(pH)$. Converting $z(pH)$ to $P_{c_{\max}}$ gives

$$P_{c_{\max}} = \Phi\left(\frac{d'}{\sqrt{2}}\right) \quad (6.24)$$

6.3.5 Calculation of d' for Same-Different Tasks

For the calculation of d' for a same-different task we adopt the convention of referring to the two relevant distributions as S_1 and S_2 (signal 1 and 2), rather than N and S . It would be unusual to employ a same-different task to measure the detectability of a target when the alternative location/interval was a blank. The same-different task is most appropriate to situations in which the observer is required to discriminate two suprathreshold stimuli without necessarily having to know the basis of the discrimination.

6.3.5.1 d' for a 2AFC Same-Different

The computation of d' for the same-different task in which both the same and different pairs are presented together during a trial is described by [Macmillan et al. \(1977\)](#). They use the term 4IAX to characterize the task, since they consider the scenario in which the four stimuli are presented in temporal order, as in an auditory experiment.

Let us begin with the standard assumption that the sensory magnitudes of S_1 and S_2 are normally distributed and separated by d' . According to [Macmillan et al. \(1977\)](#), the most likely strategy employed by observers in this situation is to compare the absolute difference between the two signals in each of the first and second pairs. The observer responds "1" if the difference between the first pair is perceived to be greater than the difference between the

second pair, and "2" otherwise. Suppose that the sensory magnitudes of the four stimuli are represented by the sequence X_1, X_2, X_3 , and X_4 . The decision rule is therefore to respond "1" if $|X_1 - X_2| > |X_3 - X_4|$ and "2" if $|X_1 - X_2| < |X_3 - X_4|$.

Figure 6.12, adapted from Macmillan et al. (1977), illustrates the computation of d' for the task. The abscissa and ordinate in the figure represent, respectively, the decision variables $X_1 - X_2$ and $X_3 - X_4$. The gray areas represent the combinations of decision variables that result in a "1" decision, i.e., areas where $|X_1 - X_2| > |X_3 - X_4|$. The gray areas can be subdivided into four regions: upper left, lower left, upper right, and lower right. On the right side of the figure the gray area defines the space in which $X_1 - X_2$ is more positive than either $X_3 - X_4$ (upper right) or $-(X_3 - X_4)$ (lower right). On the left of the figure the gray area defines the space in which $X_1 - X_2$ is more negative than either $X_3 - X_4$ (lower left) or $-(X_3 - X_4)$ (upper left).

The observer will be correct when making a "1" decision if the samples that fall within the gray regions are from any of the following sequences: $\langle S_1 S_2 S_1 S_1 \rangle$, $\langle S_1 S_2 S_2 S_2 \rangle$, $\langle S_2 S_1 S_1 S_1 \rangle$, or $\langle S_2 S_1 S_2 S_2 \rangle$. On the other hand, the observer will be incorrect when responding "1" if samples from the remaining sequences fall within the gray area, namely $\langle S_1 S_1 S_1 S_2 \rangle$, $\langle S_1 S_1 S_2 S_1 \rangle$, $\langle S_2 S_2 S_1 S_2 \rangle$, or $\langle S_2 S_2 S_2 S_1 \rangle$. P_c is therefore the probability that samples from the first four sequences will fall within either of the two (left or right) gray areas. The four rings in the figure denote volumes of the joint likelihood distributions of the various sequences of S_1 and S_2 . Note that the σ of the distributions is $\sqrt{2}$, because they are distributions of the difference between samples from two standard normal distributions.

Each volume in the left and right gray areas comprises two probabilities, a "small" and a "large." The large probability is the probability that samples from the sequences specified

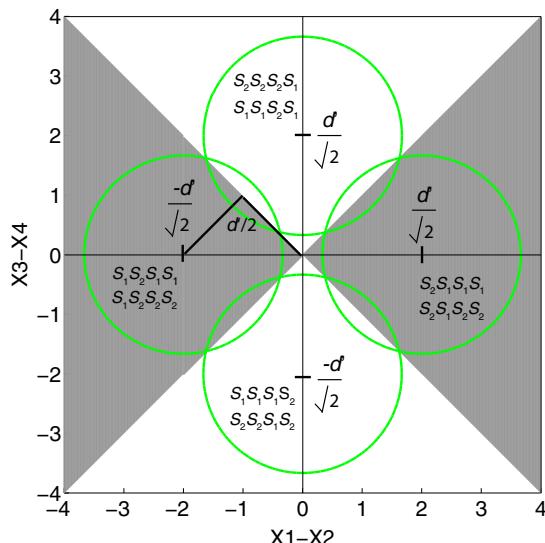


FIGURE 6.12 Graphical representation of the distributions involved in the 2AFC same-different task. $X_1 \dots X_4$ represent the internal sensory magnitudes of the four stimuli. Note that the abscissa plots $X_1 - X_2$ and the ordinate $X_3 - X_4$. The sequences $\langle S_1 S_2 S_1 S_1 \rangle$, etc. denote joint sample distributions of stimulus sequences. Note that the distance to the center of each distribution from the center of the figure is $d'/\sqrt{2}$, but when measured from a point on the diagonal perpendicular to the center of each distribution (shown by the thick black lines in the upper left quadrant) the distance is $d'/2$. The figure is adapted from Figure 6(a) in Macmillan et al. (1977).

within each gray area of the figure will fall within that area. However, there is a small probability that samples from the sequences in the opposite gray area will also fall within the area. For example, although most of the samples that fall within the gray area on the right of the figure will come from sequences $\langle S_2S_1S_1S_1 \rangle$ and $\langle S_2S_1S_2S_2 \rangle$, a few will come from $\langle S_1S_2S_1S_1 \rangle$ and $\langle S_1S_2S_2S_2 \rangle$. This is because even though most of the difference signals $S_1 - S_2$ are “large negative” and hence fall within the gray area on the left, a few will be “large positive” and will fall within the gray area on the right. Remember that it does not matter whether the difference $S_1 - S_2$ is “large negative” or “large positive,” as long as its absolute magnitude is greater than $S_2 - S_1$ or $S_1 - S_2$ (the possible sequences in the other alternative/interval). Either way the response “1” will be correct.

The larger probability within each gray area is given by $[\Phi(d'/2)]^2$ while the smaller probability is given by $[\Phi(-d'/2)]^2$. The denominator of 2 in each expression reflects the fact that the area described by the gray rectangle has sides that, by the Pythagorean Theorem, extend by $d'/2$ to the midpoint of the distribution along the side, as illustrated in the upper left quadrant of the figure. The squaring of each expression reflects the fact that one is dealing with a bivariate, i.e., joint, distribution. To obtain P_c we simply add together the large and small probabilities:

$$P_c = [\Phi(d'/2)]^2 + [\Phi(-d'/2)]^2 \quad (6.25)$$

Following [Macmillan and Creelman \(2005\)](#), the equation can be inverted to obtain d' from P_c using

$$d' = 2z \left[0.5 \left\{ 1 + (2P_c - 1)^2 \right\} \right] \quad (6.26)$$

6.3.5.2 d' for a 1AFC Same-Different: Differencing Model

For the differencing model of the 1AFC same-different task the observer is assumed to encode the perceived difference between the two stimuli in the trial, and if the absolute value of the difference exceeds a criterion the observer responds “different,” if not “same.” Suppose the signal from the first stimulus is X_1 and from the second stimulus is X_2 . The decision rule is therefore “different” if $|X_1 - X_2| > k$, where k = the criterion, and “same” otherwise. As with the 2AFC same-different task discussed in the previous section, it is useful to consider both the positive and negative parts of the difference signal $X_1 - X_2$. The top of [Figure 6.13](#) shows the distributions of sensory magnitudes for the two stimuli S_1 and S_2 , centered on 0. The middle and bottom panels (adapted from Figure 9.5 in [Macmillan and Creelman, 2005](#)) show the relative likelihoods of the various stimulus pairs as a function of the decision variable $X_1 - X_2$. The middle panel shows the Same distributions $\langle S_1S_1 \rangle$ and $\langle S_2S_2 \rangle$, and the bottom panel shows the Different distributions $\langle S_1S_2 \rangle$ and $\langle S_2S_1 \rangle$.

All the Different distributions have a σ of $\sqrt{2}$, in accordance with the variance sum law. To understand how pH and pF are calculated the criterion has been placed to one side of the midpoint. Given the value of d' and k in the figure, most of the $\langle S_2, S_1 \rangle$ signals fall above the criterion k and constitute a “large” probability. Although most of the $\langle S_1, S_2 \rangle$ signals fall to the left of k , a few will be “large positive” and fall to its right. As with the 2AFC same-different task we have to include the small probability in the calculation,

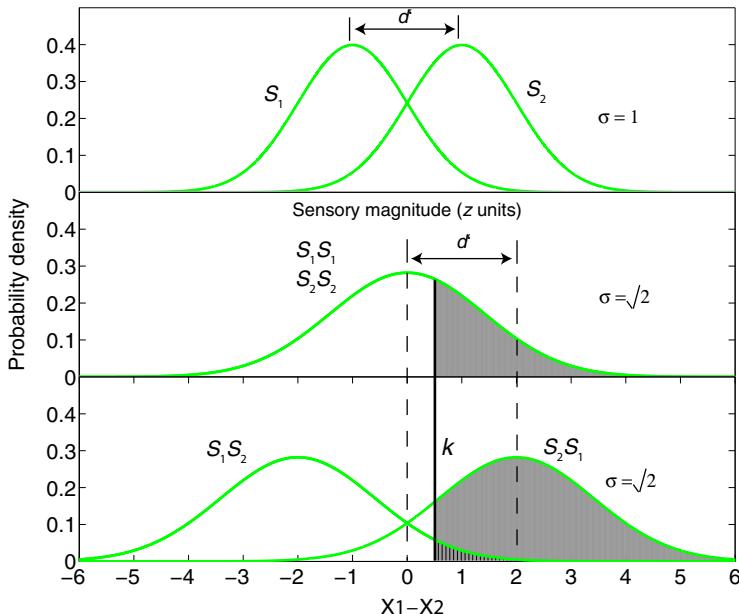


FIGURE 6.13 Method for calculating d' for a 1AFC same-different task assuming a differencing model. See text for details.

because it accords with the adopted decision rule. From Figure 6.13 the proportion of hits, pH , is given by

$$pH = \Phi\left[\frac{(d' - k)}{\sqrt{2}}\right] + \Phi\left[\frac{(-d' - k)}{\sqrt{2}}\right] \quad (6.27)$$

where the larger of the two terms is given by the gray area to the right of k and the smaller of the two terms by the hatched area to the right of k . The proportion of false alarms, pF , is given by the area to the right of the criterion line in the middle panel, multiplied by 2 since there are two distributions, i.e.,

$$pF = 2\Phi\left(-\frac{k}{\sqrt{2}}\right) \quad (6.28)$$

To calculate d' and k from pH and pF , we exploit the fact that k can be obtained directly from pF , as from Eqn (6.28): $k = -z(pF/2)\sqrt{2}$. The value of k is then substituted into Eqn (6.27) and an iterative search performed to find the value of d' that results in the input value of pH . Further details of the 1AFC same-different differencing model can be found in Macmillan and Creelman (2005).

6.3.5.3 d' for a 1AFC Same-Different: Independent Observation Model

According to Macmillan and Creelman (2005), the observer's optimum strategy for the 1AFC same-different task is to respond "different" when the signals from S_1 and S_2 fall on opposite sides of a criterion centered midway between the two distributions and if not

respond “same”. They term this model the independent observation model and suggest d' should be calculated as follows. First, calculate the P_c that an observer would obtain for this task were they to operate optimally—this is $P_{c_{\max}}$ —using pH and pF . Second, compute d' based on $P_{c_{\max}}$. Third, use the values of pH and pF to compute the criterion C in the same way as for the standard 1AFC task.

As elsewhere, it is best to begin with the method for calculating $P_{c_{\max}}$ from d' , rather than the reverse. [Macmillan and Creelman \(2005\)](#) provide a three-dimensional representation of the decision space for this model, as the calculations involve joint likelihood distributions. The two-dimensional representation provided in [Figure 6.14](#) should, however, be sufficient to understand the principle behind the calculation.

In [Figure 6.14](#), the probability that signals from both S_1 and S_2 will fall on opposite sides of the criterion positioned at 0 is the probability that S_1 falls to the left of the criterion multiplied by the probability that S_2 falls to its right (since we are dealing here with the joint probability of two events). In the figure, given the value of $d' = 1$, most of the S_2 signals fall to the right of the criterion and most of the S_1 signals fall to the left of the criterion, so the product of the two signals will be a “large” probability given by $[\Phi(d'/2)]^2$. However, there is a small probability that both a high value of S_1 and a low value of S_2 will fall on opposite sides of the criterion. These probabilities are the smaller hatched areas in the figure. The observer will also be correct in these instances, since the decision rule is always to respond “different” when the signals from the two stimuli fall on opposite sides of the criterion. The joint probability in this case is given by the product of the hatched areas, which is $[\Phi(-d'/2)]^2$. Thus, to obtain $P_{c_{\max}}$ we add up the two joint probabilities:

$$P_{c_{\max}} = \left[\Phi\left(\frac{d'}{2}\right) \right]^2 + \left[\Phi\left(-\frac{d'}{2}\right) \right]^2 \quad (6.29)$$

and from this equation, d' is given by:

$$d' = 2z\left\{ 0.5\left[1 + \sqrt{2P_{c_{\max}} - 1} \right] \right\} \quad (6.30)$$

[\(Macmillan and Creelman, 2005\)](#). To calculate d' from pH and pF , $P_{c_{\max}}$ is first estimated using

$$P_{c_{\max}} = \Phi\left\{ \frac{[z(pH) - z(pF)]}{2} \right\} \quad (6.31)$$

and the result substituted into [Eqn \(6.30\)](#). The observer’s criterion can be calculated using $C = -0.5[z(pH) + z(pF)]$.

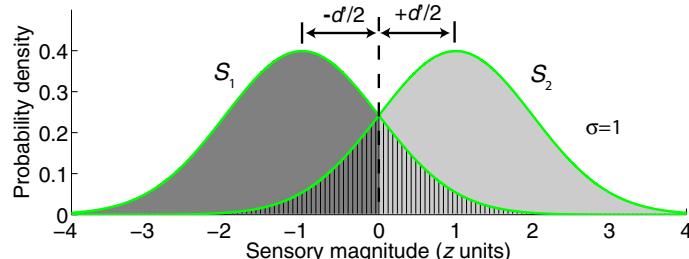


FIGURE 6.14 Principle behind the computation of d' for the independent observation model for the 1AFC same-different task. See text for details.

6.3.6 Calculation of d' for Match-to-Sample Tasks

6.3.6.1 Independent Observation Model

The computation of d' for the 2AFC match-to-sample task under the independent observation model parallels that of the 1AFC same-different task. According to [Macmillan and Creelman \(2005\)](#), who refer to the task as ABX, P_c for an unbiased observer is given by

$$P_c = \Phi\left(\frac{d'}{\sqrt{2}}\right)\Phi\left(\frac{d'}{2}\right) + \Phi\left(-\frac{d'}{\sqrt{2}}\right)\Phi\left(-\frac{d'}{2}\right) \quad (6.32)$$

We refer readers to [Macmillan and Creelman \(2005\)](#) for the derivation of this equation.

6.3.6.2 Differencing Model

For the 2AFC match-to-sample differencing model, the observer is assumed to encode the difference in sensory magnitude between the sample and each of the match stimuli, and choose the match with the smallest absolute sample-minus-match difference. According to [Macmillan and Creelman \(2005\)](#) the differencing strategy, as with the same-different task, is the more likely to be adopted in a roving experiment where many different stimuli are presented during a session. [Macmillan and Creelman \(2005\)](#) have derived the following equation for the unbiased observer:

$$P_c = \Phi\left(\frac{d'}{\sqrt{2}}\right)\Phi\left(\frac{d'}{\sqrt{6}}\right) + \Phi\left(-\frac{d'}{\sqrt{2}}\right)\Phi\left(-\frac{d'}{\sqrt{6}}\right) \quad (6.33)$$

6.3.7 Calculation of d' for M-AFC Oddity Tasks

As a rule with oddity tasks there are two stimuli, call them S_1 and S_2 . Oddity tasks are different from, say, the standard M -AFC task in that on any trial the oddity is randomly selected to be either one or the other of the two stimuli. Once again there are two possible strategies: differencing and independent observation.

6.3.7.1 Differencing Model

With the differencing strategy, the decision rule is to select from the M alternatives the one most different from the average of all the alternatives. To calculate d' for this strategy we perform a Monte Carlo simulation, as suggested by [Craven \(1992\)](#). On each simulation trial a random sample is drawn from one S_1 and $M - 1$ S_2 s. The next step is to calculate the absolute (or squared) difference between each sample and the average of all M samples. If the sample producing the biggest difference is S_1 then the simulation trial is scored "correct," else "incorrect." The process is repeated over a large number of trials and the proportion correct calculated across trials. Note that had we performed the simulation using one S_2 and $M - 1$ S_1 s the result would be the same since the absolute (or squared) difference is the computed measure. As with other computations using Monte Carlo simulation, the computed P_c will not be identical each time the simulation is run, and its variability will depend on the number of trials per simulation. However, a million trials should achieve a P_c with an accuracy of around two decimal places.

6.3.7.2 Independent Observation Model

When the independent observation strategy is employed with the oddity task, the decision rule is to select the alternative that is most likely to be S_1 and the remaining alternatives S_2 , or S_2 and the remaining alternatives S_1 . For the simplest case in which $M = 3$, d' has been calculated by Versfeld et al. (1996) as:

$$P_c = \Phi^3\left(\frac{d'}{2}\right) + \int_{-\infty}^{-d'/2} \phi(t)\Phi^2(t+d')dt \\ \dots + \left[1 - \Phi\left(\frac{d'}{2}\right)\right]^3 + \int_{-d'/2}^{\infty} \phi(t)[1 - \Phi(t+d')]^2 dt \quad (6.34)$$

To calculate d' for the M -AFC oddity task under the independent observation model, Monte Carlo simulation must be employed. The procedure is illustrated in Figure 6.15. On the first simulation trial a sample from S_1 is taken and placed into the first stimulus position, with samples from S_2 placed in the remaining $M-1$ positions. Call these samples t_1, t_2, \dots, t_M . We then calculate for each sample t_1, t_2, \dots, t_M , the likelihood that it came from S_1 and the likelihood that it came from S_2 . These likelihoods are given by the heights under the normal distribution, e.g., $\phi(t_i)$ for S_1 and $\phi(t_i - d')$ for S_2 where $i = 1 \dots M$ —see Figure 6.15. Next we

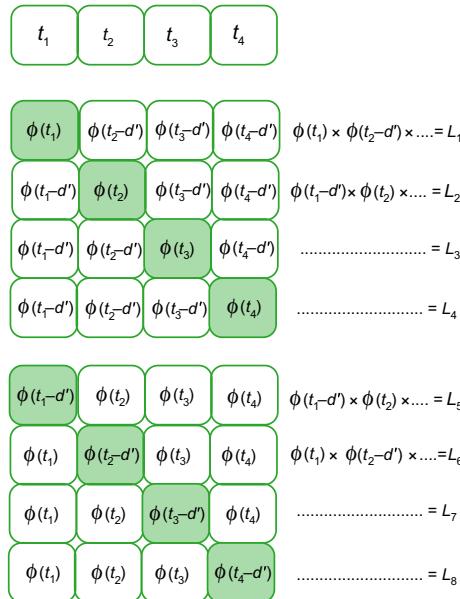


FIGURE 6.15 Principle behind the computation of d' for the M -AFC independent observation oddity task with $M = 4$ as example. The figure schematizes the computations performed on just one trial of the Monte Carlo simulation. t_1 is a random sample from the S_1 distribution, while t_2-t_4 are random samples from the S_2 distribution. $\phi(t_i)$ and $\phi(t_i - d')$ are the heights of the S_1 and S_2 distributions at t . L_1-L_8 are joint likelihood distribution. See text for further details.

consider the *joint* likelihood that S_1 is in the first position and S_2 in the remaining $M-1$ positions—denote this joint likelihood L_1 —by multiplying the likelihoods together, i.e.,

$$L_1 = \phi(t_1 - d') \times \phi(t_2) \times \phi(t_3) \times \dots \times \phi(t_M)$$

This is the computation schematized in the top row of [Figure 6.15](#). Moving to the next row, we calculate the joint likelihood L_2 that S_1 is in the second position (with S_2 in all other positions) then in row three the joint likelihood L_3 that S_1 is in the third position, and so on up to M positions. This completes the computations labelled L_1-L_4 in the figure. Next we must also consider the opposite situation, namely that S_2 occurs in each position, with S_1 filling the remaining positions. The resulting likelihoods are schematized in the rows L_5-L_8 . This gives us a set of $2M$ joint likelihoods, i.e., L_1, L_2, \dots, L_{2M} . We then determine the maximum of those joint likelihoods. If the maximum likelihood is either L_1 or L_5 we score “correct” and if not “incorrect.” We then repeat the entire procedure over many trials and calculate the average P_c .

FURTHER READING

The best starting points for SDT are [McNicol \(2004\)](#), Chapters 5–8 of [Gescheider \(1997\)](#), and [Macmillan and Creelman \(2005\)](#). The most comprehensive treatment of SDT that is accessible to the nonexpert is [Macmillan and Creelman \(2005\)](#). More mathematical treatments can be found in [Wickens \(2002\)](#) and [Green and Swets \(1974\)](#). Further details of the computation of d' for the same-different tasks can be found in [Macmillan et al. \(1977\)](#).

EXERCISES

1. Consider the $M = 2$ versions of the standard forced-choice, oddity, and match-to-sample tasks. The Palamedes routines for the M -AFC versions of these tasks assume that there are just two stimuli, S_1 and S_2 , and that the observer is unbiased and employs the following decision rules: for the standard forced-choice task select the alternative with the largest stimulus magnitude; for the oddity task select the alternative most different from the mean of all the alternatives; and for the match-to-sample task select the match most similar to the sample. For a given d' , which task would you expect to produce the biggest and which the smallest P_c ? Write a script using the Palamedes routines to plot P_c against M for a given d' for each task to test your predictions.
2. [Table 6.5](#) presents the results of an experiment aimed at measuring a psychometric function of proportion correct against stimulus magnitude using a standard 2AFC task. The experimenter is interested in the effects of bias on the estimates of the threshold and slope of the psychometric function, so the results are presented in terms of proportion hits pH and proportion false alarms pF , as calculated according to the convention in [Section 6.2.5](#). Use Palamedes to calculate d' , criterion C , and proportion correct P_c for each pair of pH and pF . Then calculate the values of $P_{c_{\max}}$ that would be expected if the observer was unbiased (see [Section 6.2.9](#)). Plot Weibull psychometric functions of both P_c and $P_{c_{\max}}$ against stimulus magnitude (Chapter 4) and obtain estimates of the

TABLE 6.5 Results of a hypothetical experiment aimed at deriving a psychometric function using a standard 2AFC task

Stimulus magnitude	pH	pF
1	0.61	0.53
2	0.69	0.42
3	0.79	0.33
4	0.88	0.18
5	0.97	0.06
6	0.99	0.03

thresholds α and slopes β for each function. Are the thresholds and slopes significantly different for the two functions (see Chapter 9)?

3. Using the stimulus magnitudes and $P_{c\max}$ values from Exercise 2, use Palamedes to fit the appropriate SDT psychometric function to the data and estimate the value of the exponent on the transducer function τ . What is the relationship between Weibull β and SDT τ ?

References

- Craven, B.J., 1992. A table of d' for M-alternative odd-man-out forced-choice procedures. *Percept. Psychophys.* 51, 379–385.
- Elliot, P.B., 1964. Tables of d' once again. In: Swets, J.A. (Ed.), *Signal Detection and Recognition by Human Observers*. Wiley, New York.
- Gescheider, G.A., 1997. *Psychophysics: The Fundamentals*. Lawrence Erlbaum Associates, Mahwah, NJ.
- Green, D.A., Swets, J.A., 1974. *Signal Detection Theory and Psychophysics*. Krieger, Huntington, New York.
- Heeger, D.J., 1991. Nonlinear model of neural responses in cat visual cortex. In: Landy, M., Movshon, J.A. (Eds.), *Computational Models of Visual Processing*. MIT Press, Cambridge, MA, pp. 119–133.
- Kingdom, F., Moulden, B., Hall, R., 1987. Model for the detection of line signals in visual noise. *J. Opt. Soc. Am.* 4, 2342–2354.
- Legge, G.E., Foley, J.M., 1980. Contrast masking in human vision. *J. Opt. Soc. Am. A* 70, 1458–1471.
- Macmillan, N.A., Creelman, C.D., 2005. *Detection Theory: A User's Guide*. Lawrence Erlbaum Associates, Mahwah, NJ.
- Macmillan, N.A., Kaplan, H.L., Creelman, C.D., 1977. The psychophysics of categorical perception. *Psychol. Rev.* 84, 452–471.
- May, K.A., Solomon, J.A., 2013. Four theorems on the psychometric function. *PLoS ONE* 8 (10), e74815.
- McNicol, D., 2004. *A Primer of Signal Detection Theory*. Lawrence Erlbaum Associates, Mahwah, NJ.
- Meese, T.S., Summers, R.J., 2009. Neuronal convergence in early contrast vision: binocular summation is followed by response nonlinearity and linear area summation. *J. Vis.* 9 (4), 7.1–7.16.
- Meese, T.S., Summers, R.J., 2012. Theory and data for area summation of contrast with and without uncertainty: evidence for a noisy energy model. *J. Vis.* 12 (11), 9.1–9.28.
- Strasburger, H., 2001. Converting between measures of slope of the psychometric function. *Percept. Psychophys.* 63 (8), 1348–1355.
- Tanner, W.P., Swets, J.A., 1954. A decision-making theory of visual detection. *Psychol. Rev.* 61 (6), 401–409.
- Versfeld, N.J., Dai, H., Green, D.M., 1996. The optimum decision rules for the oddity task. *Percept. Psychophys.* 58, 10–21.
- Wickens, T.D., 2002. *Elementary Signal Detection Theory*. Oxford University Press, Oxford, NY.
- Yeshurun, Y., Carrasco, M., Maloney, L.T., 2008. Bias and sensitivity in two-interval forced procedures. *Vision Res.* 48, 1837–1851.

Summation Measures*

Frederick A.A. Kingdom¹, Nicolaas Prins²

¹McGill University, Montreal, Quebec, Canada; ²University of Mississippi, Oxford, MS, USA

OUTLINE

7.1 Introduction	189	7.3 Part B: Summation Modeled under High-Threshold Theory (HTT)	218
7.1.1 Summation Types, Scenarios, and Frameworks	190	7.3.1 Probability Summation under HTT	218
7.2 Part A: Summation Modeled under Signal Detection Theory (SDT)	194	7.3.2 Additive Summation under HTT	222
7.2.1 Preliminaries	194	Further Reading	223
7.2.2 Additive Summation under SDT	195	References	223
7.2.3 Probability Summation under SDT	203		
7.2.4 Using the SDT Summation Formulae	209		

7.1 INTRODUCTION

An enduring question in psychophysical research concerns how multiple stimuli contribute to a visual detection task—the “summation question.” Introducing more stimuli into the target alternative/interval of a forced-choice task invariably improves detection, but it can also affect measures other than detectability, for example the slope of the psychometric function. An understanding of the behavioral consequences of introducing multiple stimuli into a detection task is important to our understanding of sensory mechanisms. For example, by comparing detection for two or more stimuli with detection for one stimulus,

*This chapter was written primarily by Frederick Kingdom.

one can go some way toward determining whether the stimuli are detected by the same or by different mechanisms. Specifically, if they are detected by separate mechanisms the stimuli will contribute independently to the detection process via a process termed “probability summation,” or PS, whereas if detected by the same mechanism, they will contribute nonindependently via a process termed here “additive summation,” or AS.

This chapter provides a detailed exposition of how one might go about deciding whether detection data conforms to one or another of PS and AS. The two types of summation are considered within two broad theoretical frameworks: Signal Detection Theory (SDT) and High-Threshold Theory (HTT). Moreover, PS and AS are considered within two different summation scenarios, termed here the Fixed Attention Window and Matched Attention Window scenarios. This means $2 \times 2 \times 2 = 8$ combinations of summation theory (SDT versus HTT), summation type (PS versus AS), and summation scenario (Fixed versus Matched Attention Window). This may seem daunting, but our step-by-step tour through the mathematical basis of each combination will hopefully make them accessible, even to the mathematical nonexpert.

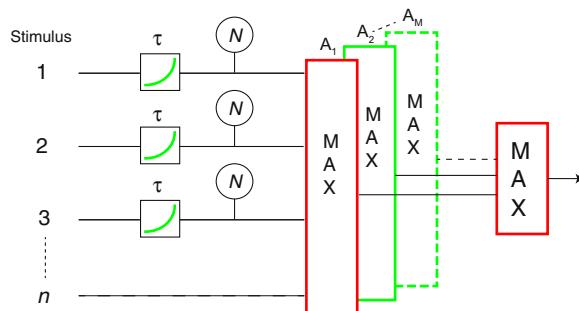
Before proceeding there are some important caveats. The models and procedures described below deal with just two forms of summation: PS and AS. However, these are not the only types of summation. There is, for example, multiplication, or one or another of its mathematical equivalents (for example the addition of logarithmically-transformed signals or their combination through AND-gating, in which the combined signal is only detectable when all the individual signals have reached threshold, e.g., see Simmons and Kingdom, 1994). Furthermore, all analyses in this chapter assume that the internal noise that limits detection is, firstly, additive (does not change in variance with signal intensity), and, secondly, independent (stochastically unrelated across signals). If the internal noise from different stimuli is multiplicative, i.e., grows with stimulus intensity, or correlated, i.e., non-independent across stimuli, then the summation predictions are different from those described here (e.g., see [Tyler and Chen, 2000](#)).

7.1.1 Summation Types, Scenarios, and Frameworks

[Figure 7.1](#) is a schematic of both PS and AS considered within our favored framework: SDT. Each stimulus intensity is first subject to a nonlinear transformation, expressed by the exponent τ . With threshold behavior, τ is typically > 1 , embodying an accelerating nonlinearity. Next, internal noise N is added. In the case of PS, the signals from the stimuli remain separate up to detection, whereas with AS they are summed together within each alternative/interval of the forced-choice task. The figure shows that the decision variable under both PS and AS summation scenarios is the MAX rule, according to which the observer chooses as the target the alternative with the biggest signal. Under PS, the MAX rule implies that the observer selects the alternative in which the signal from any of the monitored mechanisms, or “channels,” is the biggest. With AS, on the other hand, the observer first sums the signals within each alternative by a single mechanism, then chooses as the target the alternative with the biggest, summed signal. The special cases of AS are when $\tau = 1$, termed “linear” summation, and $\tau = 2$, termed “square-law” or “energy” summation.

Within both PS and AS there are additional scenarios, as schematized in [Figure 7.2](#). The superordinate classification is between the Matched Attention Window (a term coined by

Probability summation



Additive summation

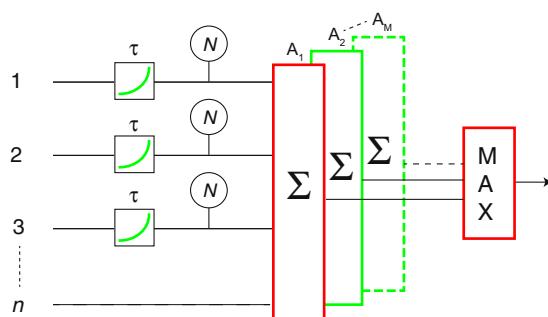


FIGURE 7.1 Two types of summation considered within an SDT framework. τ = exponent of transducer function, N = internal noise, n = number of stimuli. A_1 is the target alternative and A_2 – A_M the nontarget alternatives in the forced-choice task. MAX = MAX decision rule. See text for further details. Taken from Figure 1 in [Kingdom et al. \(2015\)](#).

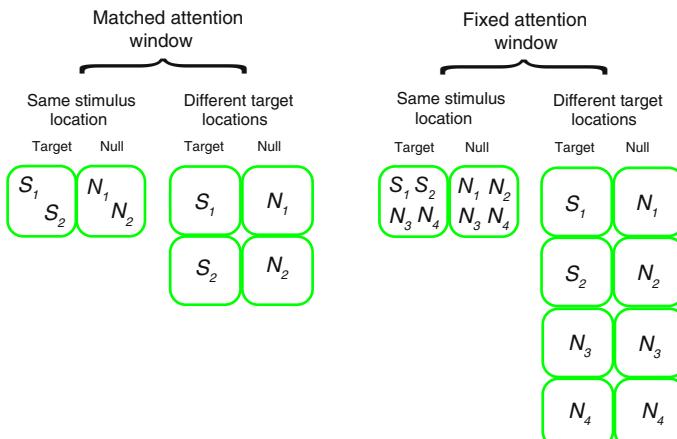


FIGURE 7.2 Summation scenarios for an $M = 2$ AFC task with $n = 2$ stimuli. Boxes along the vertical represent different potential stimulus locations. Target and Null intervals are shown next to one another. For both Matched and Fixed Attention Window scenarios there are $n = 2$ stimuli in the target interval. The number of monitored channels, Q , however, is 2 for the Matched and 4 for the Fixed Attention Window scenario. S = stimulus/signal; N = noise. Based on Figure 2 in [Kingdom et al. \(2015\)](#).

Kingdom et al., 2015), and the Fixed Attention Window (a term coined by Tyler and Chen, 2000) scenarios. In the Matched Attention Window scenario the observer is aware in advance of which stimuli are being presented on every trial, so it is safe to assume that the observer only monitors those channels that are sensitive to those stimuli. An example of this situation is when the observer knows on each block of trials whether it is stimulus A, stimulus B, or stimulus A + B that is being presented. On the other hand, with the Fixed Attention Window scenario the observer is unaware of which stimuli are being presented on a trial, whether A, B, or A + B, because the trials are randomly interleaved. In this scenario it is safe to assume that the observer monitors both A and B channels on every trial.

Figure 7.2 schematizes another important distinction: that between a compound stimulus in which the individual stimuli are located together and a compound stimulus in which the individual stimuli are located separately. The first type of compound stimulus comprises different stimuli, for example grating patches with different spatial frequencies, orientations, depths, or directions of motion. With the second type of compound stimulus, the same stimulus is presented in different locations. For example, a number of studies have measured summation in radial-frequency (RF) patterns. RF patterns are circles perturbed by sinusoidal-shaped modulations, and typically the observer is required to choose between a circle and the stimulus with the perturbation. The summation question in this case concerns how steeply performance rises with the number of cycles of the perturbation around the circle, with the view of deciding whether or not the perturbations are detected by a global mechanism that linearly integrates the information from different modulation cycles (Loffler et al., 2003; Bell and Badcock, 2008; Dickinson et al., 2010; Schmidtmann et al., 2012; Tan et al., 2013). Testing for linear global summation therefore involves measuring detection thresholds as a function of the number of RF cycles, e.g., for 1, 2, 3, 4, etc. cycles. Other examples of multiple location compound stimuli are luminance gratings with different numbers of cycles (Robson and Graham, 1981; Meese and Summers, 2007, 2012) or compound textures whose elements are positioned at various points on a grid (Graham and Sutter, 1998; Meese and Williams, 2000; Meese, 2010; Meese and Baker, 2011).

As one would expect, in all four of the scenarios in Figure 7.2, adding more stimuli invariably reduces thresholds. Moreover, certain combinations of summation type and scenario may predict changes to the slopes of the psychometric functions as more stimuli are introduced, as we shall see later. The question, however, remains: are the measured changes consistent with PS or AS? How, then, do we use threshold and slope measurements to choose between PS and AS? Two distinct frameworks have been advanced to answer this question, and they are illustrated in Figure 7.3. The earlier and simpler framework is HTT, which was first discussed in Chapter 4. Under HTT, the mechanisms that monitor each part of the stimulus (e.g., the individual cycles of a grating or radial-frequency pattern) will be activated only if their input exceeds some fixed threshold value. This threshold is assumed to be sufficiently high such that it is only very rarely surpassed by the system's internal noise on its own. This means that there is almost no penalty under HTT for monitoring additional nonstimulus or nontarget channels, as the internal noise in those channels will have a vanishingly small effect on performance. This property of the HTT framework has a significant effect on how people design their experiments. For example, there is no practical difference in the HTT PS predictions between experiments that interleave their different summation conditions and those that block those conditions—two scenarios that we have identified with the Fixed and

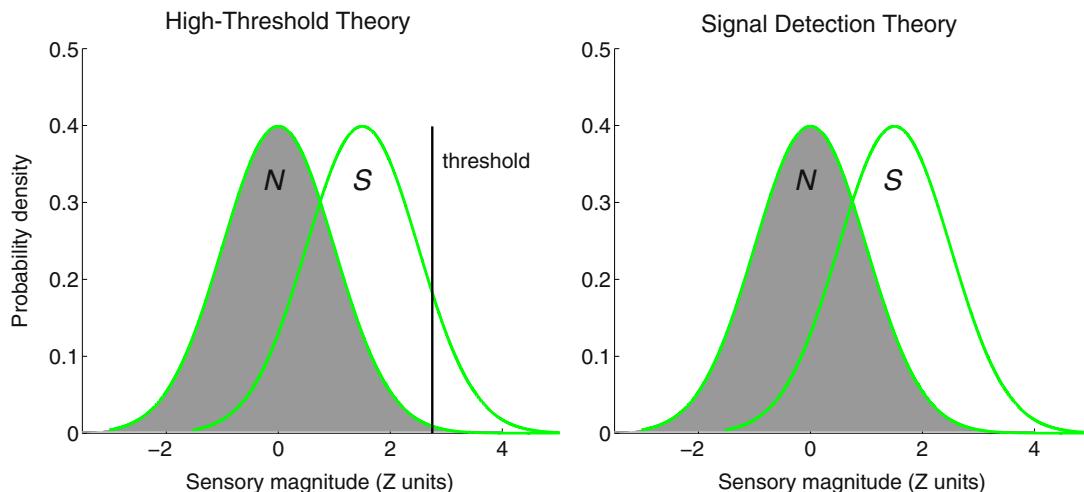


FIGURE 7.3 Two frameworks for detection. Left: HTT; right: SDT. N = noise, S = signal-plus-noise.

Matched Attention Window (Figure 7.2). Thus it makes no difference from the vantage point of HTT whether the observer does or does not know on each trial whether it is stimulus A, stimulus B, or stimulus A + B that is presented. The critical parameter for predicting PS from HTT is the slope β of the psychometric function that relates proportion correct to stimulus intensity. If we designate T as the threshold for detection and n as the number of stimuli, then PS under HTT predicts that the slope of the function relating \log threshold T to $\log n$ is $-1/\beta$. Full details of the derivation of this prediction is given in Part B of this chapter.

The other framework for considering summation is that formulated under SDT, which was the subject of the previous chapter. To recap, there is no threshold imposed upon the detection process in SDT, and so detectability is influenced by the mechanism's internal noise in both target and nontarget intervals. The SDT model takes the response from the most activated mechanism and uses that to make a decision about the stimulus, even if the most activated mechanism happens to come from the nontarget interval or in the case of a summation experiment from a channel that contains only noise. Because of this, whether the experimenter blocks or interleaves the trials for the different conditions in a summation experiment can have a profound influence on the SDT model predictions. When blocked, it is generally safe to assume that the observer monitors only those channels that contain a stimulus, ignoring those that do not. When interleaved, one instead assumes that the observer monitors all channels that potentially contain a stimulus, even when they only contain noise.

It is now widely accepted that SDT is a better framework for detection than HTT (Nachmias, 1981; Green and Swets, 1988; Tyler and Chen, 2000; Shimozaki et al., 2003; Meese and Summers, 2012; Laming, 2013; Kingdom et al., 2015). Perhaps surprisingly then, HTT is still widely employed as the framework for PS. There are arguably a number of reasons for this. First, the mathematical basis for calculating PS under the assumptions of SDT is more complex than with HTT and only recently has been fully articulated (Kingdom et al., 2015). In addition, existing formulations do not always incorporate the essential term for

the nonlinearity of the transducer function that relates internal response to stimulus intensity. The “default” assumption with SDT is that the transducer function is linear, yet an abundance of evidence points to an accelerating transducer function near threshold (Tanner and Swets, 1954; Legge and Foley, 1980; Wilson, 1980; Heeger, 1991; Meese and Summers, 2009; Meese and Summers, 2012). For these reasons, many researchers who have opted to model their data using the SDT model of PS have employed not formulae but the method of Monte Carlo simulation (e.g., Meese and Summers, 2012). Although the Monte Carlo method is simple to implement and given a sufficient number of samples accurate for most purposes, it is expensive in terms of computer processing time, prohibitively so when thousands of calculations are required, as in the psychometric function fitting procedures discussed later.

In Part A we derive from first principles formulae for calculating AS and PS under the assumptions of SDT for both Matched and Fixed Attention Window scenarios and for both equal and unequal stimulus intensities. We show how the formulae can be used as models of psychometric functions that can be used to predict how thresholds and slopes vary with different summation scenarios. Finally we show how psychometric functions can be fitted to actual data in order to determine whether AS or PS is the better model.

7.2 PART A: SUMMATION MODELED UNDER SIGNAL DETECTION THEORY (SDT)

7.2.1 Preliminaries

In Section 6.3.2 of Chapter 6 we took the reader through the logic of how, for a single stimulus, the proportion correct P_c could be calculated from the SDT measure d' , given the number of alternatives in the forced-choice task M . The basis of the calculation was the MAX decision rule, which states that on each trial the observer chooses as the target the alternative/interval with the biggest signal. The MAX decision rule also underpins all the AS and PS calculations detailed below. Before proceeding, therefore, the reader is encouraged to familiarize himself/herself with the basic theory of SDT provided in Chapter 6 and in particular Sections 6.2.11 and 6.3.2.

We begin by defining some new parameters and these are illustrated in Figure 7.4. The figure shows two Gaussian distributions that represent internal distributions of response activity, one for noise N alone, the other for the stimulus (or signal) plus noise, denoted by S . The two distributions are separated by d' . The parameter t in the figure represents a sample signal. The likelihood of obtaining t on any trial is given by the height of the distribution at t , which is $\phi(t)$ in the noise and $\phi(t - d')$ in the stimulus distribution, since we have (arbitrarily) placed 0 at the center of the noise alone distribution. The area to the left of t is $\Phi(t)$ in the noise distribution and $\Phi(t - d')$ in the stimulus distribution: these values are the probabilities that a random sample from those distributions will be less than t . The formulae for calculating ϕ and Φ are given in the previous chapter.

Since we will be dealing with psychometric functions of proportion correct as a function of stimulus intensity, we will need to convert the intensities (or amplitudes) of our stimuli into d' s. For this we need three parameters: x , g , and τ . Following Section 6.2.11, we use x to

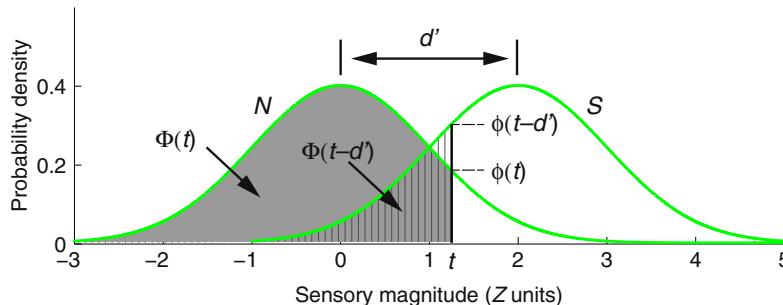


FIGURE 7.4 Parameters for calculating summation under the assumptions of SDT. N = noise distribution, S = stimulus (or signal) distribution, t = sample sensory magnitude, d' = separation between noise and signal distributions, $\Phi(t)$ and $\Phi(t - d')$ = areas under noise and signal distribution to the left of t ; $\phi(t)$ and $\phi(t - d')$ = heights of noise and signal distributions at t . Taken from Figure 3 in Kingdom et al. (2015).

denote the intensity of the stimulus and g as the scaling factor that translates stimulus intensity into d' . However, since d' is a measure of the internal strength of the stimulus it must also embody any nonlinearity in stimulus transduction. For this we raise stimulus intensity to a power, τ (see Figure 7.1). For example, a τ of 1 defines a linear transducer, a τ of 2 a square-law transducer, and so on. One might suppose that simply raising d' to the power of τ would be a straightforward way to model the transducer, but this would compromise the fact that d' is itself agnostic to any transducer nonlinearity. Moreover, because d' is an internal measure given in z , i.e., standard deviation units of the noise distribution, it is in any case a scaled version of the intensity or amplitude of the stimulus. In Section 6.2.11, we defined the relationship between d' , x , g , and τ for a single stimulus as

$$d' = (gx)^\tau \quad (7.1)$$

Two more parameters are needed to complete the picture: n and Q . n is the number of stimuli present in the target alternative/interval. Elsewhere in this book we have used N to denote the number of stimuli per trial to help clarify the differences between various types of task, for example $N = 2$ for a standard 2AFC task, $N = 3$ for a 3AFC oddity task, etc. n is different: it is the number of stimuli within the target alternative. Finally, there is Q , the number of channels or stimulus locations that are being monitored by the observer during a trial. The distinction between Q and n underpins the distinction between the Fixed and Matched Attention Window scenarios described above: for the former n can be less than Q while for the latter n is equal to Q .

7.2.2 Additive Summation under SDT

Having defined the necessary parameters we begin with AS, as this is simpler to formulate than PS. To recap, with AS the signals from the various stimuli are summed within a single mechanism. Our approach to modeling AS under SDT is to first calculate the “effective” d' of the combined stimulus. For some purposes we will need to convert the resulting d' to P_c , or vice versa, but for others this will not be necessary and we can work just with d' .

7.2.2.1 Equations for Additive Summation

Starting with the Fixed Attention Window scenario, remember that with this scenario the observer is assumed to monitor all Q channels/locations on each trial, irrespective of how many stimuli are present. This means that Q determines the total amount of internal noise that is involved. The amount of internal noise will therefore increase with Q , but when adding noise we must add their variances, not standard deviations. If σ is the standard deviation of each monitored channel's noise distribution, the resulting σ of Q noise distributions is $\sqrt{Q}\sigma^2$ or $\sigma\sqrt{Q}$, meaning that σ increases with Q by a factor of \sqrt{Q} . Thus if there are n stimuli with intensities x_1, x_2, \dots, x_n , subject to gains g_1, g_2, \dots, g_n and transducer exponents $\tau_1, \tau_2, \dots, \tau_n$, the resulting value of d' is

$$d' = \frac{1}{\sqrt{Q}} \sum_{i=1}^n (g_i x_i)^{\tau_i} \quad (7.2)$$

Note that because d' is in units of σ we do not need to specify σ as a separate parameter. If the x , g , and τ values are the same across stimuli, this simplifies to

$$d' = \frac{n(gx)^\tau}{\sqrt{Q}} \quad (7.3)$$

For the Matched Attention Window scenario, i.e., when $Q = n$, the corresponding expressions for unequal and equal x , g , and τ are

$$d' = \frac{1}{\sqrt{n}} \sum_{i=1}^n (g_i x_i)^{\tau_i} \quad (7.4)$$

and

$$d' = (gx)^\tau \sqrt{n} \quad (7.5)$$

If we wish to calculate proportion correct P_c , we substitute the value of d' into the standard formula for an M -AFC task, which from Section 6.3.2 is

$$P_c = \int_{-\infty}^{\infty} \phi(t - d') \Phi(t)^{M-1} dt \quad (7.6)$$

For the special case of $M = 2$, one can instead use the simpler formula:

$$P_c = \Phi\left(\frac{d'}{\sqrt{2}}\right) \quad (7.7)$$

With d' incorporated into one or other of the above two equations, we can denote three functions for AS under SDT:

$$P_c = AS_{SDT}(x, g, \tau, M, Q, n) \quad (7.8)$$

$$x = AS_{SDT}INV(P_c, g, \tau, M, Q, n) \quad (7.9)$$

$$P_c = AS_{SDT}uneq([x_1 x_2 \dots x_n], [g_1 g_2 \dots g_n], [\tau_1 \tau_2 \dots \tau_n], M, Q) \quad (7.10)$$

Function 7.8, abbreviated to AS_{SDT} , can be used to model AS when all x s are of equal magnitude. Function 7.9 deals with the inverse situation and is hence abbreviated to AS_{SDTINV} . For the unequal x situation we have $AS_{SDTuneq}$. Box 7.1 describes the Palamedes routines that implement the above functions.

BOX 7.1

BASIC SUMMATION COMPUTATIONS USING PALAMEDES

Summation calculations with Palamedes (Prins and Kingdom, 2009) are relatively straightforward. They are currently based on the following 12 routines, two of which, the Monte Carlo simulation routines, serve only to verify the other routines.

	Additive summation “AS”	Probability summation “PS”
Core routines	PAL_SDT_AS_SLtoPC	PAL_SDT_PS_SLtoPC
	PAL_SDT_AS_uneqSLtoPC	PAL_SDT_PS_uneqSLtoPC
	PAL_SDT_AS_PctoSL	PAL_SDT_PS_PctoSL
	PAL_SDT_AS_2uneqSLtoPC	PAL_SDT_PS_2uneqSLtoPC
	PAL_SDT_AS_Pcto2uneqSL	PAL_SDT_PS_Pcto2uneqSL
Monte Carlo simulations		PAL_SDT_PS_MonteCarlo_SLtoPC
		PAL_SDT_PS_MonteCarlo_uneqSLtoPC

The six core routines convert stimulus level x (“SL”) to proportion correct P_c (“PC”) and vice versa; the ordering of the last term in the function name defines the direction of the conversion. The inputs to the routines are the parameters x [or P_c], g , τ , M , Q , and n .

Suppose we want to calculate P_c for the additive summation of two stimuli, each with the same intensity x of 0.5, same stimulus gain g of 2.5, and same transducer exponent τ of 1.75. Say we want to do this for an $M = 2$ AFC task under the Fixed Attention Window scenario, with $Q = 4$ and $n = 2$. For this we execute

```
>>Pc = PAL_SDT_AS_SLtoPC(0.5,2.5,1.75,2,4,2)
Pc =
0.8520
```

With unequal stimulus intensities we use the routines with “uneq” in the name and input vectors, not scalars of x , g , and τ . The following example calculates P_c for the probability summation of three stimuli with unequal stimulus intensities ($x = 0.6$, 1.25, and 0.25), each with $g = 1$ and $\tau = 2$ (square-law transducer). This time set $M = 2$ and $Q = 2$ and execute

```
>> Pc = PAL_SDT_PS_uneqSLtoPC([0.6 1.25 0.25],[1 1 1],[2 2
2],2,2)
Pc =
0.8321
```

Continued

BOX 7.1 (*cont'd*)

Note that for the unequal stimulus intensity routines it is not necessary to input a value for n as this is implicit. Try other combinations of input parameters for all six of the core routines.

With unequal stimulus intensities the core routines that convert x to P_c are not invertible. However, the conversion of x to P_c can be inverted for the special case of two stimuli provided the ratio of their intensities can be specified. The four additional routines are for this purpose. Suppose the two stimuli are A and B and their corresponding intensities x_A and x_B . The ratio of their intensities r is therefore x_B/x_A . The parameter inputs to these routines are now x_A (or P_c), r , g_A , g_B , τ_A , τ_B , M , and Q . For example, let $x_A = 0.25$ and $x_B = 0.375$. This gives an r of 1.5. Set $g_A = g_B = 3$, $\tau_A = \tau_B = 1.6$, $M = 2$, and $Q = 4$. To model PS we execute

```
>>Pc = PAL_SDT_PS_2uneqSLtoPC(0.25,1.5,[3 3],[1.6 1.6],2,4)
Pc =
```

0.7320

With r specified we can perform the inverse of the routine, i.e., convert P_c to x . Let the input P_c be the same value as the output P_c from the above routine and execute

```
>>x = PAL_SDT_PS_PCTo2uneqSL(0.7320,1.5,[3 3],[1.6 1.6],2,4)
```

x =

0.2500 0.3750

The outputs are the original values of x_A and x_B .

The two Monte Carlo simulation routines enable users to confirm the operation of the corresponding PS routines.

7.2.2.2 Many versus One with Additive Summation

Often one wishes to know how much better performance is with multiple stimuli compared to one. For example, take binocular summation. In a binocular summation experiment, detection thresholds are measured for stimuli presented to the left, to the right, and to both eyes. As one would expect, thresholds tend to be lower (or sensitivity higher) with stimuli presented to both eyes, that is binocularly, compared to just one eye, that is monocularly. A thorough understanding of binocular vision, however, requires that we determine how much better performance is with two eyes compared to one. In general, the results of binocular summation experiments favor AS when the stimuli are the same in both eyes and PS when they are different (e.g., [Blake et al., 1981](#); [Meese et al., 2006](#); [Kingdom et al., 2015](#)). In a popular form of the experiment, the observer is unaware on each trial of the origin of each stimulus,

whether left eye alone, right eye alone, or in both eyes, as the stimuli are not blocked and are presented in random order. This is an example of the Fixed Attention Window scenario.

In this section we show how to calculate the relative improvements in performance with multiple stimuli that would be expected from AS. Taking first the Fixed Attention Window scenario, if we define the value of d' that produces a threshold level of P_c as d'_T ($T = \text{threshold}$), then for the Fixed Attention Window scenario ($n < Q$) we have

$$d'_T = \frac{n(gx_T)^\tau}{\sqrt{Q}} \quad (7.11)$$

where x_T is the threshold of each stimulus when presented alone. We can rearrange Eqn (7.11) to obtain x_T , thus

$$x_T = \frac{\left(\frac{d'_T \sqrt{Q}}{n}\right)^{\frac{1}{\tau}}}{g} \quad (7.12)$$

The green lines in the left-hand plot in Figure 7.5 show x_T as a function of n for various Q calculated from this equation, assuming a linear transducer of $\tau = 1$ and unit values of d'_T and g . The red line shows the prediction for the Matched Attention Window scenario, i.e., for the special case where $Q = n$. With the axes logarithmically spaced, as in the figure, the functions relating x_T to n are all straight lines. Put another way, and this is evident from inspection of Eqn (7.12), x_T is a power function of n . For the Fixed Attention Window scenario, the power function has an exponent of $-1/\tau$, whereas for the Matched Attention Window scenario the exponent is $-1/2\tau$. To see why this is, we take logarithms of both sides of the Eqn (7.12):

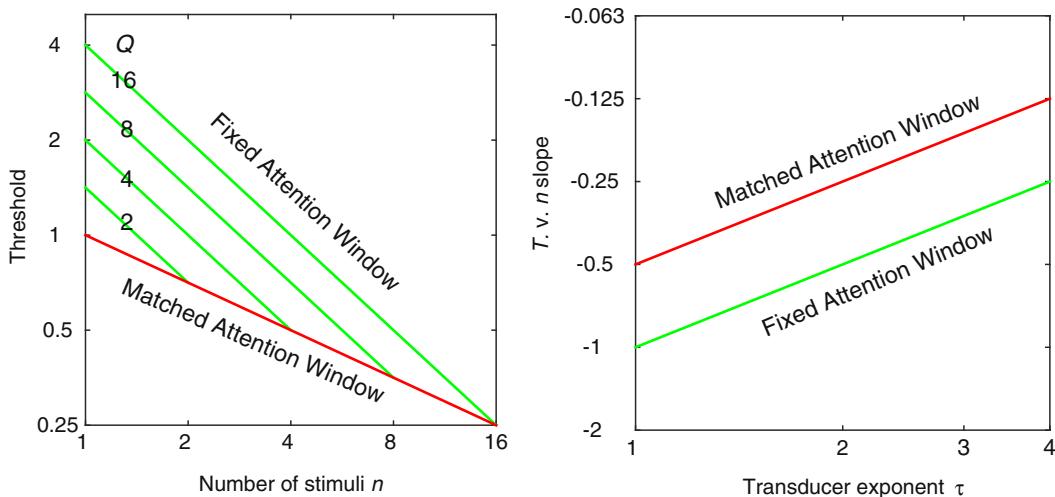


FIGURE 7.5 Predictions for AS. Left: thresholds x_T are plotted as a function of the number of stimuli n for various numbers of monitored channels Q for both Fixed (green lines) and Matched (red line) Attention Window scenarios, for a linear transducer, i.e., with $\tau = 1$ and unit d' and g . The slopes of the Fixed Attention Window plots in log–log units are all -1 and the slope for the Matched Attention Window plot is -0.5 . Right: effect of transducer exponent τ on the threshold–v. n ($T.v.n$) slopes.

$$\log x_T = \frac{1}{\tau} \log d' + \frac{1}{\tau} \log(\sqrt{Q}) - \frac{1}{\tau} \log n - \log g$$

However, because d' , g , and Q are constants we can combine the terms with these parameters into a single constant K , hence

$$\log x_T = -\frac{1}{\tau} \log n + K \quad (7.13)$$

revealing that $\log x_T$ is a linear function of $\log n$ with a slope of $-1/\tau$. If we set Q equal to n to model the Matched Attention Window scenario, we obtain instead

$$\log x_T = -\frac{1}{2\tau} \log n + K \quad (7.14)$$

revealing the $-1/2\tau$ slope. The right plot in [Figure 7.5](#) shows how these slopes vary with τ .

To summarize, for AS under SDT, thresholds are a power function of n with exponents $-1/\tau$ for the Fixed and $-1/2\tau$ for the Matched Attention Window scenarios, respectively.

The above analysis reveals that a measure of how much better we are at detecting multiple stimuli compared to one is the slope of the function relating $\log x_T$ to $\log n$, termed here the “*T.v.n*” function. Thus the greater the absolute value of the negative slope of the *T.v.n* function, the stronger the summation. *T.v.n* slopes are generic in the sense that they measure improvements from summation across n rather than for specific values of n . In the following section we will describe another generic metric of summation, the Minkowski metric, which is closely related to the *T.v.n* slope.

A metric for expressing how thresholds improve for a *specific number* of stimuli n is the summation ratio, or *SR*, defined as the ratio of single-stimulus to multiple-stimuli thresholds. Because the numerator in *SR* is the single-stimulus threshold (i.e., the higher value) *SRs* measure how sensitivity improves with the n rather than how thresholds decline with n . In some studies, for example those involving summation of contrasts at threshold, stimulus strength is sometimes measured in decibels or dBs, defined as $20 \times \log_{10}(C)$, where C is contrast. Expressed in decibels, the *SR* is given by $20 \times \log_{10}(SR)$.

To calculate *SR* we divide [Eqn \(7.12\)](#) for a single stimulus (n set to 1) by the same equation for n stimuli. A little algebra reveals that for the Fixed Attention Window scenario

$$SR = \sqrt[n]{n} \quad (7.15)$$

and for the Matched Attention Window scenario

$$SR = \sqrt[2n]{n} \quad (7.16)$$

Let us try some examples. For the Fixed Attention Window scenario, the case of $n = 2$ and $\tau = 1$ leads to an *SR* of 2 or 6 dBs. If $n = 2$ and $\tau = 2$ the *SR* is $\sqrt{2}$ or 3 dBs. For the Matched Attention Window scenario, $n = 2$ and $\tau = 1$ gives an *SR* of $\sqrt{2}$ or 1.414, or 3 dBs, while $n = 2$ and $\tau = 2$ gives an *SR* equal to the fourth root of 2, which is 1.189 or 1.505 dBs. These values are summarized in [Table 7.1](#), along with those calculated for PS under the same conditions. Note that *SR* is greater for the Fixed compared to the Matched Attention Window scenarios (for $\tau = 1$, values of 2 versus 1.414). This is because in the Fixed Attention Window scenario the same amount of noise is involved irrespective of whether one or multiple stimuli are

TABLE 7.1 Two stimuli versus one: predicted improvements in sensitivity expressed as a SR, in decibels dBs, the log–log $T.v.n$ slope, and Minkowski m , for two values of transducer exponent τ

		$\tau = 1$				$\tau = 2$			
		SR	dBs	$T.v.n$ slope	m	SR	dBs	$T.v.n$ slope	m
PS	MAW	1.216	1.7	-0.28	3.54	1.103	0.852	-0.14	7.07
	FAW	1.574	3.94	-0.65	1.53	1.255	1.973	-0.33	3.05
AS	MAW	$\sqrt{2} = 1.414$	3.01	-0.5	2	$\sqrt[4]{2} = 1.189$	1.505	-0.25	4
	FAW	2	6.02	-1	1	$\sqrt{2} = 1.414$	3.01	-0.5	2

PS = probability summation; AS = additive summation; MAW = Matched Attention Window; FAW = Fixed Attention Window

present, having the effect of disproportionately benefitting performance as more stimuli are added. In the Matched Attention Window scenario, on the other hand, the amount of internal noise grows with n , so there is no such benefit.

SRs from AS can also be calculated using Function 7.9, which converts P_c to x , given g , τ , M , Q , and n . For example, with $M = 2$, $Q = 2$, $n = 2$, and $\tau = 1$, and a threshold P_c of 0.75, we can calculate the SR thus

$$SR = \frac{AS_{SDT}INV(0.75, 1, 1, 2, 2, 1)}{AS_{SDT}INV(0.75, 1, 1, 2, 2, 2)}$$

which gives a value of 2. This method is easily implemented using Palamedes, as shown in Box 7.2.

7.2.2.3 Expressing Summation Using the Minkowski Formula

In articles about summation one often reads about “Minkowski summation” as if it were a theory of summation. However, Minkowski summation should not as a rule be thought of as a theory of summation but rather as a measure of it, an alternative to the two other summation measures described above, namely the $T.v.n$ slope and the SR. The one exception to this rule is Quick’s (1974) use of the Minkowski formulation to model PS under the assumptions of HTT. Quick’s model is described in Section 7.3.1.5 in Part B.

If S and S_{cmpd} (“cmpd” = compound) are the sensitivities to single and compound stimuli (equal to the reciprocal of the threshold measures x_T and $x_{T,cmpd}$) the expression for Minkowski summation is

$$S_{cmpd} = \left[\sum_i^n S_i^m \right]^{1/m} \quad (7.17)$$

where n is the number of stimuli and m is the “Minkowski exponent” that defines the inverse of the degree of summation (note that m is not the same as the number of alternatives/intervals in the forced-choice task, M). When the sensitivities of the stimuli are unequal m must be obtained by iterative search but for stimuli with equal sensitivities, Eqn (7.17) simplifies to

$$S_{cmpd} = [nS^m]^{1/m} \quad (7.18)$$

BOX 7.2**CALCULATING SUMMATION RATIOS**

Suppose we want to compute the *SR* for the additive summation of two stimuli under the Matched Attention Window scenario. For example, let us our criterion threshold P_c to 0.75, stimulus gain g to 3, and transducer exponent τ to 1.25. Since we are computing the ratio of thresholds for one compared to two stimuli, we set $n = 1$ for the numerator routine, $n = 2$ for the denominator routine, and Q equal to n in both routines. Thus we execute

```
>>SR = PAL_SDT_AS_PCToSL(0.75,3,1.25,2,1,1)/PAL_SDT_AS_PCToSL(0
.75,3,1.25,2,2,2)
```

SR =

1.3195

To calculate the result for *PS* for the same set of parameters we simply replace the “A” in *AS* with “P”, then execute

```
SR=PAL_SDT_PS_PCToSL(0.75,3,1.25,2,1,1)/PAL_SDT_PS_PCToSL(0.7
5,3,1.25,2,2,2)
```

SR =

1.1690

Try other *SR* scenarios using the core summation routines.

In this case m is easily solved given S_{cmpd} , S , and n . If we replace sensitivity in Eqn (7.18) with the reciprocal of threshold, we have the alternative formulation

$$\frac{1}{x_{T,cmpd}} = \left[n \left(\frac{1}{x_T} \right)^m \right]^{1/m} \quad (7.19)$$

Applying a little algebra leads to

$$\log x_{T,cmpd} = -\frac{1}{m} \log n + \log x_T \quad (7.20)$$

revealing that the slope of the function relating $\log x_{T,cmpd}$ to $\log n$ is $-1/m$. From the previous section readers should see that this is the *T.v.n* slope. Therefore, the two measures of summation, Minkowski m and the *T.v.n* slope, are simply related: the former is the negative reciprocal of the latter. For example, with linear summation, m and the *T.v.n* slope are 1 and -1 , respectively; with no summation m is infinity and the *T.v.n* slope is 0; and with m equal to 2

the $T.v.n$ slope is -0.5 . Always remember that with Minkowski m the greater the summation the smaller the value of m .

What is the relationship between m and the SR , defined as the ratio of the single to compound stimulus thresholds, i.e., $x_T/x_{T.cmpd}$? Rearranging Eqn (7.19) and putting it in terms of SR gives

$$SR = \sqrt[n]{n} \quad (7.21)$$

Note that this expression for SR has the same form as that derived earlier for AS under the Fixed Attention Window scenario but with m rather than τ . This, however, is a special case. Solving Eqn (7.21) for m gives

$$m = \frac{\log n}{\log SR} \quad (7.22)$$

Why use Minkowski m , or for that matter the $T.v.n$ slope, rather than SR ? As we mentioned earlier, SR is specific to n , whereas Minkowski m and the $T.v.n$ slope are generic in that they apply across n .

Table 7.1 calculates our various measures of summation for the two-versus-one stimulus case, assuming equal sensitivity to both stimuli. The table provides measures for both AS and PS, for both Matched and Fixed Attention scenarios, and for both $\tau = 1$ and $\tau = 2$. Summation is expressed in four ways: SR , dBs, $T.v.n$ slope, and Minkowski m .

One can glean from **Table 7.1** that m is proportional to τ for a given condition. Given that τ is itself proportional to the slope of the psychometric function for a given condition, it follows that m is also proportional to the slope of the psychometric function for a given condition.

7.2.3 Probability Summation under SDT

To recap, with PS the signals from compound stimuli are processed separately, and the relative ease with which a compound compared to a single stimulus is detected is due to the increased chance that one of the signals in the compound will be the biggest.

7.2.3.1 Equations for Probability Summation

The calculation of PS under SDT is more complex than with AS but is nevertheless tractable. **Box 7.3** provides an exposition that closely follows the one in [Kingdom et al. \(2015\)](#). The formulae here may be considered as extensions of Eqn B10 in [Shimozaki et al. \(2003\)](#), which calculated PS for the specific situation of two different stimuli in an M -AFC task under the Matched Attention Window scenario. The more general formula derived in **Box 7.3** for the equal intensity stimulus case is

$$\begin{aligned} P_c = n & \int_{-\infty}^{\infty} \phi(t - d') \Phi(t)^{QM-n} \Phi(t - d')^{n-1} dt \\ & + (Q - n) \int_{-\infty}^{\infty} \phi(t) \Phi(t)^{QM-n-1} \Phi(t - d')^n dt \end{aligned} \quad (7.23)$$

and for the unequal stimulus intensity case

BOX 7.3

DERIVATION OF EQUATIONS FOR PROBABILITY SUMMATION UNDER SDT

B.7.3.1 Equal Stimulus Intensities

Consider the situation shown in [Figure 7.2](#) for the Matched Attention Window scenario in which there are two stimuli with equal intensities, i.e., $n = Q = 2$. Since we are dealing with PS we assume that the two stimuli are detected independently or by different “channels.” The decision rule here is the same as in an AS task: the MAX rule. However, because the observer is monitoring two channels, he/she will make a correct decision if either S_1 or S_2 produces the biggest signal. In order to calculate the expected proportion correct P_c for this situation we must first calculate the probability that S_1 will produce the biggest signal, second that S_2 will produce the biggest signal, and then add the two probabilities together. Note that for either one of the two stimuli to produce the biggest signal, the samples from *both* noise locations in the null interval *and* the sample from the other stimulus in the target interval must be less than the sample t from S_1 . Taking first the two noise samples N_1 and N_2 , the probability that both will be less than the sample t from S_1 is $\Phi(t) \times \Phi(t) = \Phi(t)^2$. For the other stimulus sample S_2 , the probability that it will be less than t from S_1 is $\Phi(t - d')$. To obtain the probability that both noise samples and the other stimulus sample will be smaller than t from S_1 we multiply these two probabilities together, i.e., $\Phi(t)^2\Phi(t - d')$. Finally, to obtain the probability p that a random sample t from S_1 will produce the biggest signal, we integrate this product across all values of t , taking into account the relative probability of obtaining t , which is its height in the stimulus distribution $\phi(t - d')$. The result is

$$p = \int_{-\infty}^{\infty} \phi(t - d')\Phi(t)^2\Phi(t - d')dt$$

However, this equation only deals with the trials in which S_1 produces the biggest signal; to calculate the total proportion correct we must also include those trials in which S_2 gives the biggest signal. Since S_1 and S_2 are identical, we only need to multiple the above equation by 2 to obtain the total proportion correct i.e.,

$$P_c = 2 \int_{-\infty}^{\infty} \phi(t - d')\Phi(t)^2\Phi(t - d')dt \quad (\text{B7.3.1})$$

[Equation \(B7.3.1\)](#) thus calculates the expected proportion correct for a 2AFC task with two identical stimuli under the Matched Attention Window scenario, assuming PS under SDT.

The next step is to generalize this equation to any M and n . The number of noise signals in the null intervals in the Matched Attention Window scenario is $n(M - 1)$. The number of stimuli is n , and the number of other stimuli with which each stimulus signal has to be compared is $n - 1$. Incorporating these terms into [Eqn \(B7.3.1\)](#) we obtain

$$P_c = n \int_{-\infty}^{\infty} \phi(t - d')\Phi(t)^{n(M-1)}\Phi(t - d')^{n-1}dt \quad (\text{B7.3.2})$$

BOX 7.3 (*cont'd*)

For the Fixed Attention Window scenario in [Figure 7.2](#), in which some of the channels in the target interval contain noise rather than stimulus, we must also consider the possibility that these target-interval noise samples, of which there are $Q - n$, might produce the biggest signal, for if any of them does it will also result in a correct decision under the MAX rule. To incorporate these target-interval noise signals we follow the same logic as above and introduce a second line to the equation. We must also replace the exponent $n(M - 1)$ in the first line with $QM - n$, the total number of noise samples, and in the second line use the exponent $QM - n - 1$, the total number of noise samples with which each noise sample in the target interval must be compared. The result is

$$\begin{aligned} P_c = & n \int_{-\infty}^{\infty} \phi(t - d') \Phi(t)^{QM-n} \Phi(t - d')^{n-1} dt \\ & + (Q - n) \int_{-\infty}^{\infty} \phi(t) \Phi(t)^{QM-n-1} \Phi(t - d')^n dt \end{aligned} \quad (\text{B7.3.3})$$

Note that [Eqn \(B7.3.3\)](#) reduces to [Eqn \(B7.3.2\)](#) when $n = Q$. For this reason we designate [Eqn \(B7.3.3\)](#) as the general equation for computing PS under SDT for both Matched and Fixed Attention Window scenarios, when all n stimuli produce the same d' .

[Equation \(B7.3.3\)](#) is invertible because only a single d' value is involved. As with the general equation for computing P_c from d' for an M -AFC task that we derived in the previous chapter, the inverse of [Eqn \(B7.3.3\)](#) can only be implemented by iterative search.

B7.3.2 Unequal Stimulus Intensities

The above analysis deals with the situation in which all stimuli are of equal intensity (or rather equal d'). What if they are unequal? In the unequal stimulus intensity scenario, our two stimuli, S_1 and S_2 , have different d' 's: call these d'_1 and d'_2 . Following the same argument as above, we begin with the probability that a sample t from S_1 will be bigger than the two noise samples from null interval. This is $\Phi(t)^2$. The probability that sample t will be bigger than the one other stimulus sample is $\Phi(t - d'_2)$. Integrating across all t samples of S_1 and then adding in the corresponding integral for the probability that a sample from S_2 will be the biggest signal, we obtain

$$\begin{aligned} P_c = & \int_{-\infty}^{\infty} \phi(t - d'_1) \Phi(t)^2 \Phi(t - d'_2) dt \\ & + \int_{-\infty}^{\infty} \phi(t - d'_2) \Phi(t)^2 \Phi(t - d'_1) dt \end{aligned} \quad (\text{B7.3.4})$$

Continued

BOX 7.3 (*cont'd*)

If we generalize Eqn (B7.3.4) to an M -AFC task we arrive at Eqn (B10) in Shimozaki et al. (2003).

$$\begin{aligned} P_C = & \int_{-\infty}^{\infty} \phi(t - d'_1) \Phi(t)^{2(M-1)} \Phi(t - d'_2) dt \\ & + \int_{-\infty}^{\infty} \phi(t - d'_2) \Phi(t)^{2(M-1)} \Phi(t - d'_1) dt \end{aligned} \quad (\text{B7.3.5})$$

Extending Shimozaki et al.'s equation to the three stimulus case with d' 's d'_1 , d'_2 , and d'_3 we obtain

$$\begin{aligned} P_C = & \int_{-\infty}^{\infty} \phi(t - d'_1) \Phi(t)^{3(M-1)} \Phi(t - d'_2) \Phi(t - d'_3) dt \\ & + \int_{-\infty}^{\infty} \phi(t - d'_2) \Phi(t)^{3(M-1)} \Phi(t - d'_1) \Phi(t - d'_3) dt \\ & + \int_{-\infty}^{\infty} \phi(t - d'_3) \Phi(t)^{3(M-1)} \Phi(t - d'_1) \Phi(t - d'_2) dt \end{aligned} \quad (\text{B7.3.6})$$

Note that as we introduce more stimuli we increase the number of terms in each integral because there are now more stimuli with individual d' values that must be compared to the one under consideration. However, if we replace the right-hand part of each integral with the terms containing different d' values by the product notation, and then use the sum notation to add together the different integrals, we can generalize Eqn (B7.3.6) to n signals to obtain

$$P_C = \sum_{i=1}^n \int_{-\infty}^{\infty} \phi(t - d'_i) \Phi(t)^{n(M-1)} \prod_{j=1, j \neq i}^n \Phi(t - d'_j) dt \quad (\text{B7.3.7})$$

For the Fixed Attention Window scenario with unequal stimuli and Q monitored channels we apply the same logic as for the equal stimulus case. The result is

$$\begin{aligned} P_C = & \sum_{i=1}^n \left[\int_{-\infty}^{\infty} \phi(t - d'_i) \Phi(t)^{QM-n} \prod_{j=1, j \neq i}^n \Phi(t - d'_j) dt \right] \\ & + (Q-n) \int_{-\infty}^{\infty} \phi(t) \Phi(t)^{QM-n-1} \prod_{j=1}^n \Phi(t - d'_j) dt \end{aligned} \quad (\text{B7.3.8})$$

Equation (B7.3.8) thus calculates P_C for n independently detected stimuli with internal stimulus strengths $d'_1, d'_2, d'_3, \dots, d'_K$, for an M -AFC task with Q monitored channels, according to the MAX decision rule under the assumptions of SDT.

$$\begin{aligned}
 P_c = & \sum_{i=1}^n \left[\int_{-\infty}^{\infty} \phi(t - d'_i) \Phi(t)^{QM-n} \prod_{j=1, j \neq i}^n \Phi(t - d'_j) dt \right] \\
 & + (Q - n) \int_{-\infty}^{\infty} \phi(t) \Phi(t)^{QM-n-1} \prod_{j=1}^n \Phi(t - d'_j) dt
 \end{aligned} \tag{7.24}$$

As with the equations for AS described earlier, we can substitute d' with $(gx)^\tau$ in order to incorporate the stimulus scaling factor g and transducer exponent τ . We therefore have three functions for calculating PS under the assumptions of SDT that are analogs of the three AS functions given earlier:

$$P_c = PS_{SDT}(x, g, \tau, M, Q, n) \tag{7.25}$$

$$x = PS_{SDT}INV(P_c, g, \tau, M, Q, n) \tag{7.26}$$

$$P_c = PS_{SDT}tuneq([x_1 \ x_2 \dots x_n], [g_1 \ g_2 \dots g_n], [\tau_1 \ \tau_2 \dots \tau_n], M, Q) \tag{7.27}$$

7.2.3.2 Applying the PS_{SDT} Functions

Figure 7.6 shows example outputs of the function $PS_{SDT}INV$. The plot on the left shows compound stimulus thresholds at a criterion P_c of 0.75 as a function of the number of stimuli n for a 2AFC task assuming a linear transducer, i.e., $\tau = 1$. The continuous green curves are

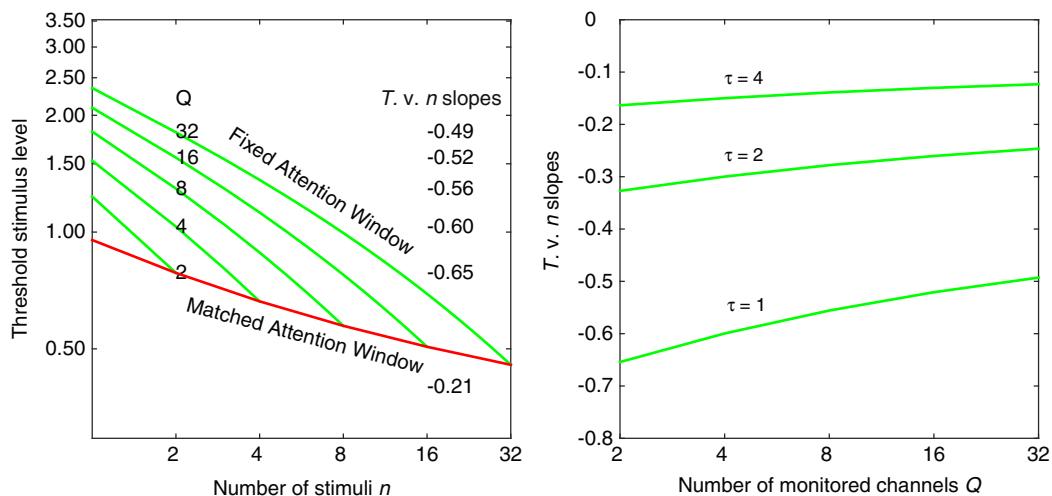


FIGURE 7.6 Left: results of applying the function $PS_{SDT}INV$ for an $M = 2$ AFC task and a linear transducer, i.e., $\tau = 1$. The graph plots thresholds at 0.75 proportion correct as a function of the number of stimuli n for various numbers of monitored channels Q . $T.v.n$ slopes are best fitting straight lines to the log-log data. Green lines are for the Fixed Attention Window scenario, red line for the Matched Attention Window scenario. Right: $T.v.n$ slopes as a function of Q for various τ for the Fixed Attention Window scenario.

for various values of Q under the Fixed Attention Window scenario ($n < Q$), while the red curve is for the Matched Attention Window scenario ($Q = n$).

The $T.v.n$ slope values in the left graph are obtained by fitting straight lines to the log–log data. Note, however, that the curves are not perfectly straight on the log–log spaced plots: they become slightly steeper as n increases. Note also that the $T.v.n$ slopes decrease in absolute magnitude as Q increases. The right-hand graph in Figure 7.5 shows how the $T.v.n$ slopes vary as a function of Q for various values of the transducer exponent τ . Note how the absolute magnitudes of the $T.v.n$ slopes decrease proportionately with τ .

For the Matched Attention Window scenario, shown by the red line in the left figure, the $T.v.n$ slope for $\tau = 1$ is around -0.25 for the lower range of n , declining in absolute magnitude as the range of n increases to a value of around -0.21 . Summation slopes of around -0.25 are often termed “fourth root summation” (Quick, 1974; Graham et al., 1978; Pelli, 1985; Chen and Tyler, 1999). Thus fourth-root summation describes the improvement in thresholds by PS for the Matched Attention Window scenario for lower ranges of n , assuming a linear transducer. As with the Fixed Attention Window scenario, $T.v.n$ slopes for the Matched Attention Window scenario decrease in absolute magnitude proportionally with τ ; for example, for $\tau = 2$ the $T.v.n$ slope is -0.11 across the full range of n in the graph.

7.2.3.3 Many versus One with Probability Summation

To calculate the SR for both Matched and Fixed Attention Window scenarios under PS we use the function $PS_{SDT}INV$, which was given earlier as

$$x = PS_{SDT}INV(Pc, g, \tau, M, Q, n) \quad (7.28)$$

Take the simplest case of a 2AFC task, with two stimuli compared to one, a linear transducer ($\tau = 1$), and a Matched Attention Window ($Q = n$). Since we are taking a ratio, the value of g is arbitrary, so it can be set to unity. If we adopt the criterion Pc of 0.75, the SR can be calculated thus

$$SR = \frac{PS_{SDT}INV(0.75, 1, 1, 2, 1, 1)}{PS_{SDT}INV(0.75, 1, 1, 2, 2, 2)}$$

which gives a value of 1.216. Expressed in dBs this is 1.7. Note that the only differences between the numerator and denominator in the above expression are the values of Q and n . For the same example with a τ of 2, i.e., a square-law transducer, the result is 1.103 or 0.85 dBs.

For the Fixed Attention Window scenario, we set Q to 2 in both the numerator and denominator, such that now the only difference is the value of n :

$$SR = \frac{PS_{SDT}INV(0.75, 1, 1, 2, 2, 1)}{PS_{SDT}INV(0.75, 1, 1, 2, 2, 2)}$$

which gives 1.574 or 3.9 dBs. For $\tau = 2$ the result is 1.255 or 1.97 dBs. Note that as with AS, the SR under PS is also greater for Fixed compared to Matched Attention Window scenarios (for $\tau = 1$ 1.574 versus 1.216). Table 7.1 gives the various measures of summation for the two-versus-one case for PS, and Box 7.2 shows how to do the calculations using Palamedes.

7.2.4 Using the SDT Summation Formulae

7.2.4.1 Modeling Summation with Simulated Psychometric Functions

Typically in summation experiments, we measure psychometric functions of proportion correct versus stimulus intensity. For example, one might measure three psychometric functions, one for stimulus A presented on its own, one for stimulus B presented on its own, and one for stimulus A + B in combination. The question for us is whether the psychometric function for A + B is best explained by AS or PS.

There are different ways one can approach the modeling of psychometric functions for the summation of compound stimuli. One approach is to simulate psychometric functions based on AS and PS and see how they differ. Let us begin with a relatively simple example.

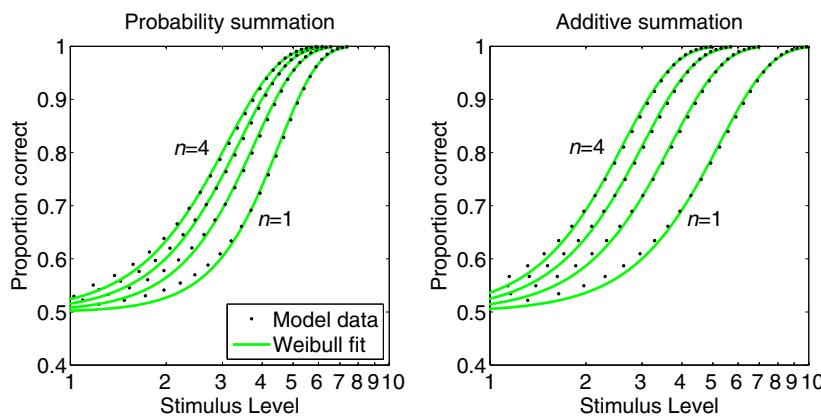


FIGURE 7.7 Example psychometric functions generated using the PS_{SDT} function for PS (left) and the AS_{SDT} function for AS (right). For both graphs the fixed parameters are $M = 2$, $\tau = 2$, and $Q = 4$, while n varies from 1 to 4. Each plot has been fitted with a Weibull function shown as the continuous green lines. *Taken from Figure 4 in Kingdom et al. (2015).*

TABLE 7.2 Weibull threshold (α) and slope (β) values for the psychometric functions in Figure 7.7

n	PS		AS	
	α	β	α	β
1	1.36	3.53	1.57	2.68
2	1.13	3.10	1.11	2.68
3	1.01	2.84	0.91	2.68
4	0.93	2.66	0.79	2.68

PS = Probability summation; AS = additive summation.

Figure 7.7 shows psychometric functions generated by the PS_{SDT} and AS_{SDT} functions for a Fixed Attention Window scenario, with parameters $M = 2$, $\tau = 2$, and $Q = 4$. The four curves in each graph are for four values of n : 1, 2, 3, and 4. The dots are proportions correct for 30 linearly spaced x values derived from the summation model. The continuous green lines are Weibull psychometric functions fitted to the dots (see Chapter 4). We use the Weibull as it is widely employed to fit data from summation experiments. The abscissa is logarithmically spaced to show more clearly how the slopes of the psychometric functions change with n . **Table 7.2** gives the Weibull threshold α and slope β values for the psychometric functions in the figure.

Figure 7.7 and **Table 7.2** reveal important properties of PS and AS when a Weibull is used to fit the psychometric functions. Note that the Weibull fit is not perfect—a Cumulative Normal or Logistic would do better—but it is sufficient for providing estimates of the thresholds and slopes. As one would expect, thresholds fall with n for both types of summation but more steeply for AS than PS. Do not forget, however, that the steepness of the decline in thresholds for both types of summation depends on τ , the exponent on the transducer function, with the decline becoming less as τ increases. In the case of AS, **Table 7.2** shows that for a square-law transducer, Weibull thresholds fall as the square root of n ; this can also be gleaned from [Eqn \(7.12\)](#). The most interesting feature of [Fig. 7.7](#), however, is the way that the slopes β of the psychometric functions vary with n . With AS the slopes remain constant, but with PS they decline. This is a property of β that occurs irrespective of τ and constitutes a critical behavioral signature that allows one to distinguish between PS and AS for the Fixed Attention Window scenario modeled under SDT.

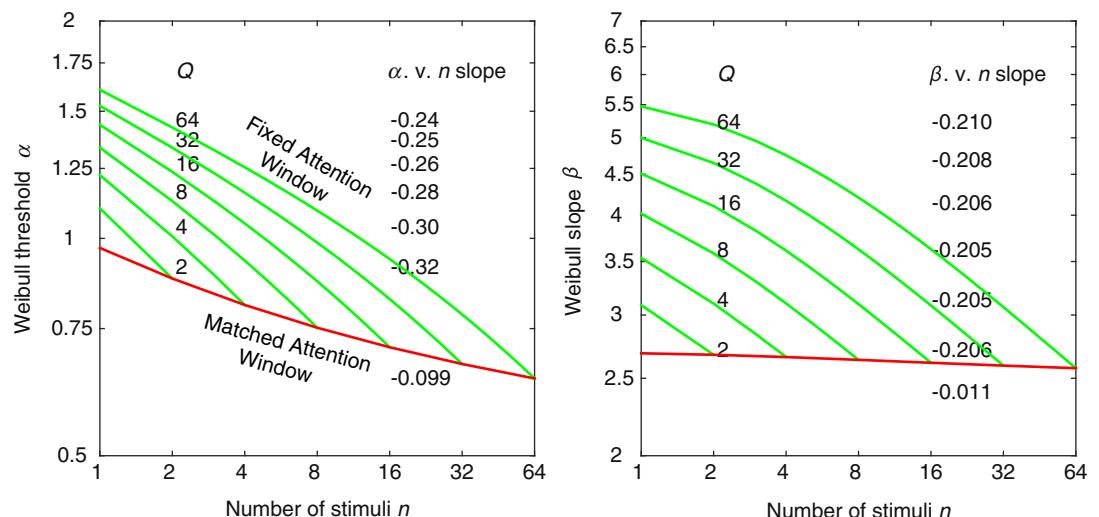


FIGURE 7.8 Weibull thresholds α (left) and slopes β (right) fitted to psychometric functions generated by the PS_{SDT} function, as a function of the number of stimuli n for a transducer exponent τ of 2. Green lines = Fixed Attention Window; red lines = Matched Attention Window. Q = number of monitored channels. $\alpha.v.n$ and $\beta.v.n$ slope values are calculated from straight line fits to the log–log data. Taken from Figure 5 in [Kingdom et al. \(2015\)](#).

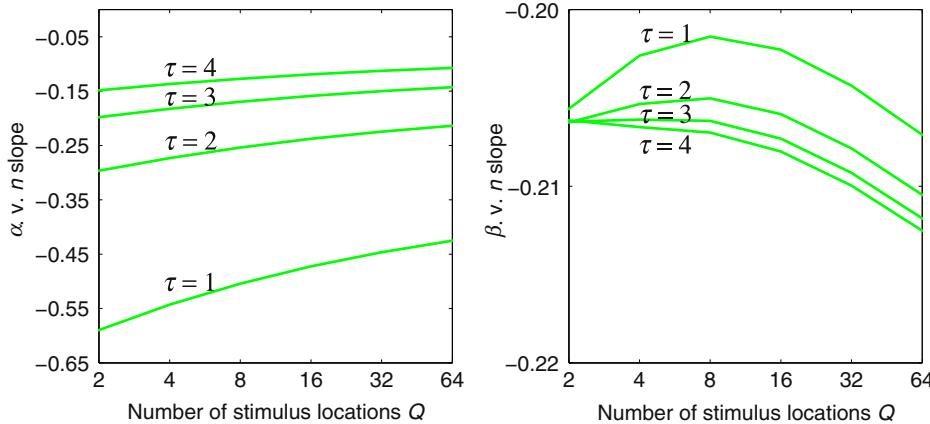


FIGURE 7.9 $\alpha.v.n$ (left) and $\beta.v.n$ (right) slopes as a function of Q for various values of the transducer exponent τ . Note that the $\alpha.v.n$ slopes vary moderately with Q but significantly with τ whereas the $\beta.v.n$ slopes vary minutely with both Q and τ . Taken from Figure 6 in [Kingdom et al. \(2015\)](#).

Figure 7.8 shows how with PS the Weibull a and β estimates vary with n and Q for both Fixed and Matched Attention Window scenarios assuming a square-law transducer, i.e., $\tau = 2$. Note that the reason why the $\alpha.v.n$ slopes in the figure are less in absolute magnitude than the $T.v.n$ slopes shown earlier in Figure 7.6 is not because here thresholds are given as Weibull α but as T in Figure 7.6 but because τ is 2 here but 1 in Figure 7.6. Figure 7.9 shows how the $\alpha.v.n$ and $\beta.v.n$ slopes vary as a function of τ and Q .

7.2.4.2 Simulating Summation Squares

A popular method for measuring summation is the summation-square experiment. In this type of experiment, thresholds are obtained for various ratios of intensities of two stimuli, A and B. The thresholds are then plotted on a graph whose axes are the intensities of A and B.

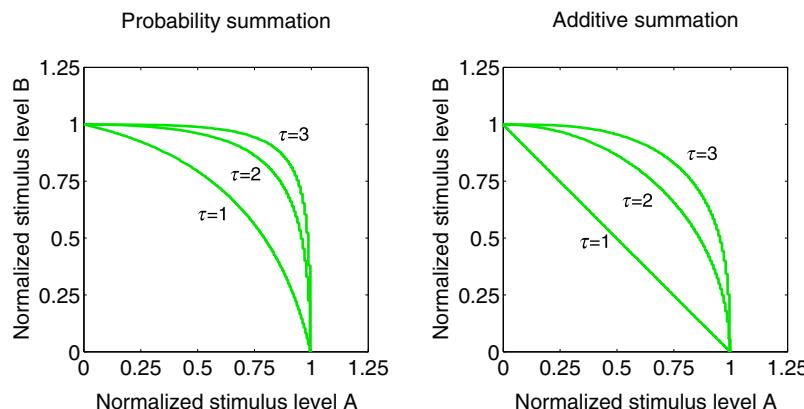


FIGURE 7.10 Predicted summation-square thresholds for combinations of two stimuli as a function of the transducer exponent τ , normalized to their individual detection thresholds. Left: PS; right: AS.

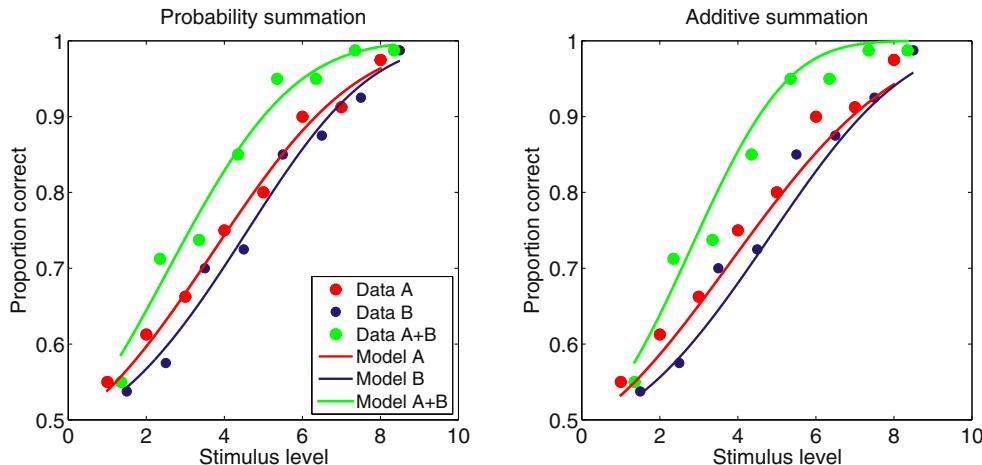


FIGURE 7.11 Hypothetical psychometric functions for a 2AFC summation experiment that measures the detection of stimulus A, stimulus B, and stimulus A + B under the Fixed Attention Window scenario. Proportion correct is plotted against stimulus level for each psychometric function. Continuous lines are the PS (left) and AS (right) fits obtained by fitting all three psychometric functions simultaneously according to each model.

The points that lie on the axes are the “A-alone” and “B-alone” thresholds, while the points that lie in between are the thresholds for the various ratios of A to B. The threshold values that are plotted are usually estimated from psychometric functions of P_c across stimulus intensity for each ratio of A to B.

Figure 7.10 shows the predictions for Weibull thresholds for both PS and AS as a function of τ . The intensities of A and B have been normalized to their individual detection thresholds to bring out the effect of τ more clearly. With AS, when $\tau = 1$, i.e., a linear transducer, the prediction is a straight line, whereas for PS it is bow-shaped. However, bow-shaped functions are obtained for both AS and PS when $\tau > 1$. This is important, as one might be tempted to reject AS on the grounds that the function is bow-shaped, whereas one can only reject linear summation on this basis.

7.2.4.3 Working with Actual Psychometric Function Data

Figure 7.11 shows hypothetical data for the 2AFC detection of two stimuli, A and B, when presented separately and in combination, assuming a Fixed Attention Window scenario, i.e., for when the trials for A, B, and A + B are randomly interleaved. As can be seen the psychometric function for A + B is, as expected, shifted to the left of the A-alone and B-alone functions. The question is whether the improvement in thresholds for the A + B stimulus implicates PS or AS. One of the potential complicating factors in the analysis is that the units of stimulus A and stimulus B might not be the same. This would happen for example if stimulus A was a chromatically defined pattern and stimulus B a luminance-defined pattern and there was no agreed upon metric for both types of stimuli. Fortunately, the only problem that arises with the use of different metrics, or different intensities for A and B, concerns the units for plotting the data. Different researchers will adopt different solutions to the data-plotting problem. However, from the standpoint of applying a summation model to the data, the actual units for A and B do not matter because A and B can have different stimulus gains and different transducer exponents (see Boxes 7.4 and 7.5).

BOX 7.4

USING THE MULTIPLE-FIT SUMMATION ROUTINE IN PALAMEDES

The Palamedes multifit summation function `PAL_SDT_Summ_MultiplePFML_Fit` is designed to fit both PS and AS models to multiple psychometric functions measured for the detection of combined stimuli. As we go to press the routine is configured to model only the Fixed Attention Window scenario, that is when the number of monitored channels Q is fixed for all stimulus combinations. A future update will configure the routine to also model the Matched Attention Window scenario.

In the following example, we use the routine to simultaneously fit three psychometric functions, one for the detection of stimulus A, one for stimulus B, and one for stimulus A + B. The task is 2AFC and the three stimulus conditions, A, B, and A + B, are assumed to be interleaved, implying a Fixed Attention Window scenario, with the number of monitored channels Q fixed at 2 throughout. The hypothetical data are shown in [Figure 7.11](#). We first fit the psychometric functions with a PS model and estimate four parameters: g_A , g_B , τ_A , and τ_B —these are the gains g and transducer exponents τ for stimuli A and B. Bear in mind that these are PS model estimates; if we were to find that an AS model gave a better fit to the data (see below), then the AS model parameter estimates would be the better ones. First we designate the values M and Q :

```
>>M = 2;
>>Q = 2;
```

Next, we define a $3 \times 2 \times N$ array for the stimulus levels, where N is the number of levels for each stimulus. In order to model the Fixed Attention Window scenario, we specify stimulus levels for both A and B in all three psychometric functions, using zeros if the stimulus is absent. The first dimension specifies the psychometric function (A, B, A + B) and the second the stimulus (A, B). For example, type and execute

```
>>StimLevels(1,1,:) = [1 2 3 4 5 6 7 8];
>>StimLevels(1,2,:) = [0 0 0 0 0 0 0 0];
>>StimLevels(2,1,:) = [0 0 0 0 0 0 0 0];
>>StimLevels(2,2,:) = [1.5 2.5 3.5 4.5 5.5 6.5 7.5 8.5];
>>StimLevels(3,1,:) = [1.2 2.2 3.2 4.2 5.2 6.2 7.2 8.2];
>>StimLevels(3,2,:) = [1.5 2.5 3.5 4.5 5.5 6.5 7.5 8.5];
```

Next we fill a matrix with the number of correct responses for each stimulus level. Rows are for A, B, and A + B, and columns are for the number correct for each stimulus level. For example

```
>>NumPos(1,:) = [44 49 53 60 64 72 73 78];
>>NumPos(2,:) = [43 46 56 58 68 70 74 79];
>>NumPos(3,:) = [44 57 59 68 76 76 79 79];
```

and the matrix for the number of trials for each stimulus level

BOX 7.4 (*cont'd*)

```
>>OutOfNum(1,:) = [80 80 80 80 80 80 80 80];
>>OutOfNum(2,:) = [80 80 80 80 80 80 80 80];
>>OutOfNum(3,:) = [80 80 80 80 80 80 80 80];
```

Next we specify the summation function we want to use. Because the stimulus levels for A and B are not the same we use `PAL_SDT_PS_uneqSLtoPC`. Type and execute

```
>>SummFunc = @PAL_SDT_PS_uneqSLtoPC;
```

Now we specify initial guesses for the four parameter estimates. In the routine the gains g_A and g_B are contained in one vector and transducer exponents τ_A and τ_B in another. Let our guesses for g_A and g_B both be 0.3 and for τ_A and τ_B both 1.5. The transducer exponents are termed in the code "p," so one could for example name the vectors `PSgParams` and `PSpParams`. Thus type and execute

```
>>PSgParams = [0.3 0.3];
>>PSpParams = [1.5 1.5];
```

We are now ready to run the fitting routine: Type and execute

```
>>[PSgParams, PSpParams, PSnegLL] = ...
PAL_SDT_Summ_MultiplePFML_Fit(StimLevels, PSgParams,PSpParams,
NumPos, OutOfNum,SummFunc,M,Q)
```

The output should be

```
PSgParams =
0.2898    0.2508

PSpParams =
1.2499    1.4779

PSnegLL =
829.1145
```

The output gives the estimates of the four parameters and the negative of the log-likelihood of the model fit. The log-likelihood can be used to compare the PS and AS models. To obtain the parameter estimates and log-likelihood for the AS model we designate `SummFunc` as

BOX 7.4 (*cont'd*)

PAL_SDT_AS_uneqSLtoPC and name our parameter vectors ASgParams and ASpParams. The output this time should be

```
ASgParams =
0.2795    0.2408

ASpParams =
1.4267    1.7321

ASnegLL =
837.1332
```

To determine whether PS or AS is the better model we calculate the difference in their AIC as follows:

```
>>PSaic = -2*(-PSnegLL)+2*4;
>>ASaic = -2*(-ASnegLL)+2*4;
>>AICdiff = ASaic - PSaic;
```

where the number 4 corresponds to the number of free parameters (see Chapter 9 for details). The AIC difference value returned is 16.037. With AIC, lower AIC values indicate better fitting models, so in this example the positive AIC difference means that the PS model is preferred (given that both models have the same number of free parameters, 4, the preferred model is simply the one with the larger log-likelihood). Given that the PS model gives the better fit, then the parameter estimates from that model should be the favored ones.

The bootstrap errors and goodness-of-fit *p*-values for the parameter estimates can be obtained respectively using

PAL_SDT_Summ_MultiplePFML_BootstrapParametric and
 PAL_SDT_Summ_MultiplePFML_GoodnessOfFit, which take the same inputs as the multiple fit routine but using the fitted parameter estimates rather than their guesses. However, beware that these routines are generally slow to execute, depending on how many simulations are specified.

In what follows we describe two approaches to modeling the data in Figure 7.11, termed the “individual-fit” and “multiple-fit” methods. The individual-fit method is as follows:

1. Fit the psychometric functions for A and B separately using the PS_{SDT} and AS_{SDT} functions, with M set to 2, Q to 2, and n to 1. The fits for each model produce estimates of g_A , g_B , τ_A , and τ_B .

BOX 7.5

RUNNING THE MULTIPLE-FIT AND SUMMATION-SQUARE SUMMATION DEMO SCRIPTS

Palamedes currently has two scripts for demonstrating the multiple-fit method. The first deals with the three-psychometric function situation detailed in [Box 7.4](#) and for the data shown in [Figure 7.11](#). When you run the script you will be prompted for the number of simulations for the bootstrap errors and the number of simulations for the goodness-of-fit tests. If you only want the estimates of g_A , g_B , τ_A , and τ_B , then set the number of simulations to 2. The parameters are fit using the routine `PAL_SDT_Summ_MultiplePFML_Fit`. The script compares the PS and AS model fits obtained from this routine by calculating the difference in the AIC for each model (see [Box 7.4](#) for details).

For the summation-square data shown in [Figure 7.12](#) the script is `PAL_SDT_PSVAS_SummSquare_Demo`. The script uses the same multiple-fit method as for the three-psychometric function data but this time fitting the five psychometric functions, whose Weibull thresholds are shown in the figure, again to obtain estimates of g_A , g_B , τ_A , and τ_B , their bootstrap errors, the model AIC differences, and goodness-of-fit p -values.

2. Using g_A , g_B , τ_A , and τ_B , make a prediction for the A + B proportion correct data using PS_{SDT} and AS_{SDT} , again with M set to 2 and Q to 2, but this time with n set to 2.
3. Perform a goodness-of-fit test between the predicted and actual A + B data separately for the AS and PS models. The model giving the better goodness-of-fit is the better model.

Although in principle this method is fine, the preferred method here is the multiple-fit method. In the multiple-fit method, we simultaneously fit all three psychometric functions, “in one go.” One of the reasons why we prefer this method is because the data from all three psychometric functions is used to fit each model, rather than just the A-alone and B-alone psychometric functions, as in the individual-fit method. The fitting procedure with the multiple-fit method again produces estimates of g_A , g_B , τ_A , and τ_B for each model. The resulting fits are the continuous lines shown in [Figure 7.11](#). [Table 7.3](#) shows the estimates of the four parameters for the better-fitting model, which for this example is PS, together with their bootstrap errors. For the multiple-fit method an appropriate measure for comparing the two

TABLE 7.3 Parameter estimates and bootstrap errors for the PS model fit to the data in [Figure 7.11](#) using the multiple-fit method; the PS model gives the better fit

	g_A	g_B	τ_A	τ_B
Value	0.29	0.25	1.25	1.48
SE	0.025	0.018	0.18	0.2

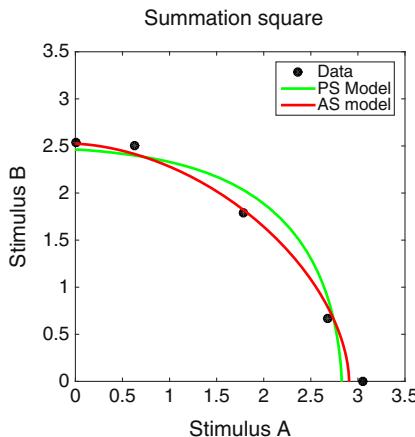


FIGURE 7.12 Hypothetical summation square data from a 2AFC task using five ratio combinations of stimuli A and B. The dots are thresholds obtained by fitting each psychometric function with a Logistic function. The probability (PS) and additive (AS) model predictions are obtained by simultaneously fitting all five psychometric functions with each model, then using the parameter estimates to generate synthetic psychometric functions whose Logistic-fitted thresholds are the continuous lines. See text for further details.

models is Akaike's Information Criterion (AIC), which is explained in Chapter 9. The difference between the AS and PS model AICs for the data in Figure 7.11 is 16.0. Since this value is positive, the PS model is favored.

The multiple-fit method also lends itself to the analysis of psychometric function data from a summation square experiment. Figure 7.12 shows hypothetical data together with the AS and PS model predictions. The black dots are Logistic thresholds estimated from the fits to five psychometric functions, each obtained from various intensity ratios of the A and B stimuli. Fitting the data with a Logistic (or Weibull, etc.) is not part of the method proposed here for evaluating whether PS or AS is the best model but is in keeping with common practice for displaying summation square data. The AS and PS models were first fit to the raw psychometric function data to produce parameter estimates for each model. These parameter estimates were then used to simulate psychometric functions for a range of stimulus ratios, each then fitted with a Logistic to produce thresholds that are the continuous lines in the graph. The important methodological feature of the multiple-fit approach is that each model fit to the data is obtained by simultaneously fitting all five psychometric functions.

The difference in AIC between the AS and PS models for the summation square data is -8.32 , which in this case favors the AS model. The parameter estimates and bootstrap errors for this model are given in Table 7.4.

TABLE 7.4 Parameter estimates and bootstrap errors for the AS model fit to the data in Figure 7.12 using the multiple-fit method; the AS model gives the better fit

	g_A	g_B	τ_A	τ_B
Value	0.41	0.47	1.7	1.74
SE	0.022	0.027	0.22	0.21

7.3 PART B: SUMMATION MODELED UNDER HIGH-THRESHOLD THEORY (HTT)

Although we have argued that HTT is not the best framework for modeling summation data, it still enjoys wide usage, in particular for modeling PS. It is important, therefore, to understand its theoretical basis. Moreover, there is always the possibility that evidence comes to light supporting HTT as the framework for some types of detection task. To our knowledge there are no detailed expositions of HTT in the literature—though a brief one is provided in [Graham \(1989\)](#)—so we have attempted to provide one here. Since it is the HTT predictions for PS rather than AS that are most widely employed, we will first consider how HTT models PS.

7.3.1 Probability Summation under HTT

7.3.1.1 A Simple Coin Tossing Exercise

Suppose you have two coins, and you want to know the probability that at least one of them will come up heads. There is more than one way that at least one head can come up from two coin tosses, so the simplest method of doing the calculation is to consider the probability that neither coin will come up heads—this is the only “negative” outcome—and subtract the result from 1. Suppose the probability of obtaining a head H is $p(H)$. The probability of a tail is therefore $1 - p(H)$ and the joint probability of obtaining tails from both coins is $(1 - pH)^2$. So the probability that at least one of the two coins will come up heads, which we designate $p(H|2)$, is simply one minus this joint probability, i.e.,

$$p(H|2) = 1 - [1 - p(H)]^2 \quad (7.29)$$

Thus, if $p(H) = 0.5$, then for two coin tosses $p(H|2) = 1 - (1 - 0.5)^2 = 0.75$. Note that the probability of obtaining at least one head from two coin tosses, 0.75, is greater than the probability of obtaining a head from just one coin toss, 0.5, simply because there are more chances of obtaining a head from two coin tosses compared to one. As you increase the number of coin tosses you increase the chance of obtaining at least one head. For example, with three coin tosses $p(H|3) = 1 - (1 - 0.5)^3 = 0.875$. Thus, with n coin tosses the probability of obtaining at least one head is

$$p(H|n) = 1 - [1 - p(H)]^n \quad (7.30)$$

The principle also works if you have coins that do not all have the same $p(H)$. Suppose for example your coins are twisted and as a result have different probabilities of coming up heads. If $p(H)_i$ is the probability of the i th coin coming up heads, then using the product notation, the probability of at least one head coming up from n coins is

$$p(H|n) = 1 - \prod_i^n [1 - p(H)_i] \quad (7.31)$$

What is the connection here with the HTT framework for modeling PS? According to HTT, the only way an observer can fail to detect a stimulus is if every channel fails to detect it. Since the outputs of the various channels are independent, the probability that all of them will fail

to detect the stimulus is the product of the probabilities that each will fail. The probability of detection is thus 1 minus this value. This is what is calculated in [Eqns \(7.30\) and \(7.31\)](#).

7.3.1.2 Proportion Correct in Forced-Choice Tasks under HTT

In a forced-choice experiment we do not measure directly the probability p that a channel detects the stimulus. Rather, we measure the proportion of correct detections, P_c . The observer will always be correct so long as one interval/alternative produces a detect state, because according to HTT, the internal noise in the other, blank interval never exceeds the threshold and hence never produces a false alarm. However, if the stimulus intensity is insufficient to produce a detect state in the target interval, neither interval will produce a detect state, and the observer must guess. Thus, P_c embodies not only the probability that the stimulus will be detected by a channel but also the contribution from guessing when the stimulus is undetectable.

Suppose we conduct a forced-choice experiment and measure proportion correct P_c for two stimuli, call them A and B, presented alone. What would we expect P_c to be when both stimuli are presented together, i.e., A + B, assuming they are detected independently, according to HTT? Let the guessing rate in the forced-choice task be γ , and let p_A and p_B be the probabilities that the A and B channels will detect the stimuli. The P_c s that will result purely from guessing will be $\gamma(1 - p_A)$ for A and $\gamma(1 - p_B)$ for B. Therefore, the expected P_c for each stimulus presented on its own will be

$$P_{cA} = p_A + \gamma(1 - p_A); \quad P_{cB} = p_B + \gamma(1 - p_B) \quad (7.32)$$

The same form of expression also holds for the A + B compound stimulus, i.e.,

$$P_{cAB} = p_{AB} + \gamma(1 - p_{AB}) \quad (7.33)$$

Now the probability of detection of the A + B stimulus (which as stated above is not the same as P_c) is

$$p_{AB} = 1 - (1 - p_A)(1 - p_B)$$

If we substitute this into [Eqn \(7.33\)](#) and simplify, we obtain

$$P_{cAB} = 1 - (1 - \gamma)(1 - p_A)(1 - p_B)$$

However, all we have from our experiment are proportions correct, so we need to obtain P_{cAB} from P_{cA} and P_{cB} , not from p_A and p_B . To do this we first rearrange the equations in [Eqn \(7.32\)](#) to put them in terms of p_A and p_B , i.e.,

$$p_A = \frac{P_{cA} - \gamma}{1 - \gamma}; \quad p_B = \frac{P_{cB} - \gamma}{1 - \gamma}$$

then substitute these into the equation for P_{cAB} and simplify. The result is

$$P_{cAB} = 1 - \frac{1}{(1 - \gamma)}(1 - P_{cA})(1 - P_{cB}) \quad (7.34)$$

[Equation \(7.34\)](#) is the basic equation for calculating the expected proportion correct in a forced-choice task when two stimuli are combined by PS under HTT. Take for example a

2AFC task, i.e., one in which the guessing rate $\gamma = 0.5$. Suppose Pc_A and Pc_B are both 0.75. Then $Pc_{AB} = 1 - 2 \times 0.25 \times 0.25 = 0.875$. Note that had we substituted the Pcs of 0.75 into the earlier Eqn (7.31), which does not take into account the guessing rate, the result would be $1 - 0.25 \times 0.25 = 0.9375$, a higher Pc . The difference is due to the contribution of guessing. Taking another example, suppose we have a 4AFC task, i.e., $\gamma = 0.25$, again with Pcs for A and B alone of 0.75. Now $Pc_{AB} = 1 - 1.333 \times 0.25 \times 0.25 = 0.917$.

We can generalize Eqn (7.34) to n independently detected stimuli using the product notation

$$Pc_{cmpd} = 1 - \frac{1}{(1 - \gamma)^{n-1}} \prod_i^n (1 - P_{ci}) \quad (7.35)$$

where Pc_{cmpd} stands for proportion correct for the compound stimulus.

7.3.1.3 Summation Psychometric Functions under HTT

From Chapter 4, the general form of the psychometric function that relates stimulus intensity x to proportion correct Pc is

$$Pc(x; \alpha, \beta, \gamma, \lambda) = \gamma + (1 - \gamma - \lambda)F(x; \alpha, \beta) \quad (7.36)$$

where F stands for a particular psychometric function, e.g., Logistic, Weibull, Quick, etc. with four parameters: α threshold, β slope, γ guess rate, and λ lapse rate.

In principle, Pc in both Eqns (7.34) and (7.35) can be replaced by the full expressions of any psychometric function F . By “full” this means including the part of the expression that involves the guess and lapse rates γ and λ . However, when dealing with PS under HTT, the two forms of psychometric function that lend themselves most easily to mathematical analysis are the Weibull and its close relative the Quick.

Take the Weibull. If we assume a lapse rate of zero, then proportion correct Pc for a single stimulus is given by

$$Pc = \gamma + (1 - \gamma) \left[1 - \exp\left(-\left(\frac{x}{\alpha}\right)^\beta\right) \right]$$

which can be slightly simplified to

$$Pc = 1 - (1 - \gamma) \exp\left(-\left(\frac{x}{\alpha}\right)^\beta\right) \quad (7.37)$$

Suppose we have a compound stimulus comprising two independently detected stimuli and assume that the psychometric function for each stimulus alone yields proportions correct Pc_A and Pc_B , based on parameters α_A, β_A and α_B, β_B , respectively. If we substitute the Weibull equation for each stimulus into Eqn (7.34), the result simplifies to

$$Pc_{AB} = 1 - (1 - \gamma) \exp\left(-\left(\frac{x}{\alpha_A}\right)^{\beta_A}\right) \exp\left(-\left(\frac{x}{\alpha_B}\right)^{\beta_B}\right) \quad (7.38)$$

Note that the prefix $1 - (1 - \gamma)$ comes out to be the same here as for the preceding single stimulus Weibull. This is a fortuitous result that is a property of the Weibull. However, the reader must not assume that one can simply transfer the prefix when going from single to

compound stimuli: the full equation of the psychometric function must always be inserted into Eqn (7.34).

We can generalize Eqn (7.38) for n different independently detected stimuli:

$$P_{C_{cmpd}} = 1 - (1 - \gamma) \prod_i^n \exp\left(-\left(\frac{x}{\alpha_i}\right)^{\beta_i}\right) \quad (7.39)$$

If all the stimuli are identical, this becomes

$$P_{C_{cmpd}} = 1 - (1 - \gamma) \exp\left(-n\left(\frac{x}{\alpha}\right)^{\beta}\right) \quad (7.40)$$

One can see from this equation that the slope of the predicted psychometric function β for the compound stimulus is the same irrespective of the number of stimuli n . This is the first key property of PS under HTT, and as we shall see it also holds for AS.

7.3.1.4 Many versus One with Probability Summation under HTT

Let us set x to its threshold value α . For a single stimulus, P_{c_T} (T = threshold) is given by

$$P_{c_T} = 1 - (1 - \gamma) \exp(-1)$$

For n stimuli, each having the same threshold α , and with the compound stimulus threshold denoted by α_{cmpd} , we have

$$P_{c_T} = 1 - (1 - \gamma) \exp\left(-n\left(\frac{\alpha_{cmpd}}{\alpha}\right)^{\beta}\right)$$

Since the right-hand parts of the two expressions above are by definition equal, we can solve for α_{cmpd}/α and obtain

$$\frac{\alpha_{cmpd}}{\alpha} = \left(\frac{1}{n}\right)^{\frac{1}{\beta}} \quad (7.41)$$

To see how α varies with n we take logarithms of both sides and rearrange:

$$\frac{\log\left(\frac{\alpha_{cmpd}}{\alpha}\right)}{\log n} = -\frac{1}{\beta} \quad (7.42)$$

This reveals that the slope of the function relating thresholds to the number of stimuli n , when plotted on log–log axes, falls with a slope of $-1/\beta$. This is the second key property of PS under HTT.

7.3.1.5 Quick Pooling Formula for Probability Summation under HTT

In Section 7.2.2.3 we described how the exponent m in the Minkowski formula was inversely related to the amount of summation and thus could be used as a general measure of summation, i.e., irrespective of whether summation is probability, additive, linear, or whatever. However, when m is replaced with the slope β of the Weibull psychometric function, the Minkowski formula is constrained to express a particular amount of summation, which equals, as it turns out, the amount predicted by PS under the assumptions of HTT. This was first demonstrated by Quirk (1974), and for this reason the version of the

Minkowski formula with β as the exponent is termed the Quick pooling model of PS (e.g., Wilson, 1980; Meese and Williams, 2000).

To see how Quick's formula serves to model PS under HTT when incorporating β , we take Eqn (7.41) and put it in Minkowski form:

$$\frac{1}{\alpha_{cmpd}} = \left[n \left(\frac{1}{\alpha} \right)^\beta \right]^{1/\beta}$$

Since sensitivity S is defined as the reciprocal of threshold, the above expression can be written as

$$S_{cmpd} = \left[n(S)^\beta \right]^{1/\beta} \quad (7.43)$$

This is the Quick pooling model, as applied to equal strength stimuli. However, one must always remember that in order for the formula to constitute a model of PS under HTT, the exponent must be the slope β of the Weibull (or Quick) psychometric function.

7.3.2 Additive Summation under HTT

The AS prediction of P_c under HTT is straightforward. If all n stimuli are equal, we simply multiply the stimulus intensities x by n to obtain the equation for the compound psychometric function:

$$P_{cmpd} = 1 - (1 - \gamma) \exp \left(- \left(\frac{nx}{\alpha} \right)^\beta \right) \quad (7.44)$$

Note that with AS, as with PS, the slope of the psychometric function β is unaffected by n .

7.3.2.1 Many versus One with Additive Summation under HTT

If we set equal the equation for P_{cmpd} with the equation for $n = 1$ and simplify, we obtain

$$\frac{\alpha_{cmpd}}{\alpha} = \frac{1}{n} \quad (7.45)$$

To see how α varies with n we take logarithms of both sides, rearrange, and obtain

$$\frac{\log \left(\frac{\alpha_{cmpd}}{\alpha} \right)}{\log n} = -1 \quad (7.46)$$

In other words, the slope of the function relating thresholds to the number of stimuli n , when plotted on log–log plots, falls with a slope of -1 . This is the third defining property of (additive) summation under HTT.

Figure 7.13 summarizes the three aforementioned defining properties of summation under HTT by showing how both α and β vary with n for both PS and AS.

Note finally that the results for summation under HTT are the same irrespective of whether one assumes a Fixed or Matched Attention Window scenario.

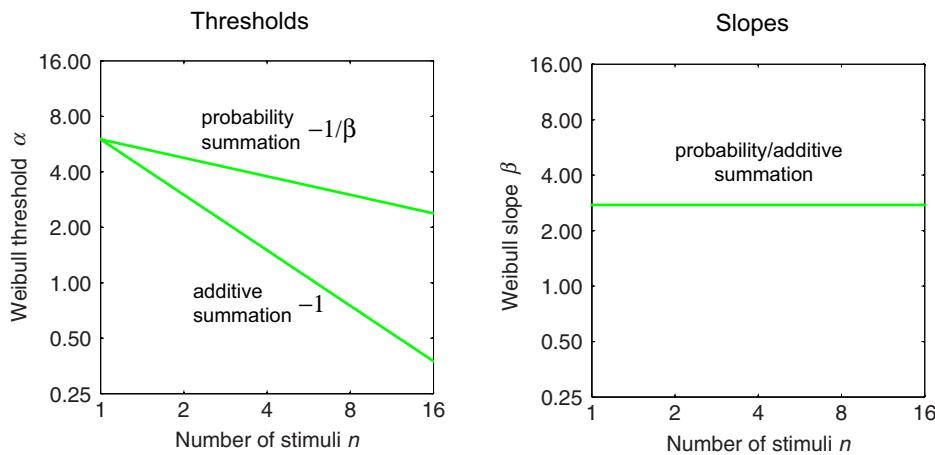


FIGURE 7.13 Predictions for PS and AS under HTT. Weibull thresholds α (left) and slopes β (right) are plotted as a function of the number of stimuli n in a summation experiment. The log–log slopes of the functions relating α to n are given in the left-hand figure.

FURTHER READING

Expositions of summation at detection threshold that are both readable and thorough are unfortunately few and far between. The reader, however, is encouraged to delve into the many articles referenced in this chapter.

References

- Bell, J., Badcock, D.R., 2008. Luminance and contrast cues are integrated in global shape detection with contours. *Vision Res.* 48 (21), 2336–2344.
- Blake, R., Sloane, M., Fox, R., 1981. Further developments in binocular summation. *Percept. Psychophys.* 30 (3), 266–276.
- Chen, C.C., Tyler, C.W., 1999. Accurate approximation to the extreme order statistics of Gaussian samples. *Commun. Stat-Simul. C.* 28, 177–188.
- Dickinson, J.E., Han, L., Bell, J., Badcock, D.R., 2010. Local motion effects on form in radial frequency patterns. *J. Vis.* 10 (3), 20.1–20.15.
- Graham, N., Robson, J.G., Nachmias, J., 1978. Grating summation in fovea and periphery. *Vision Res.* 18, 815–825.
- Graham, N., Sutter, A., 1998. Spatial summation in simple (Fourier) and complex (non-Fourier) texture channels. *Vision Res.* 38, 231–257.
- Graham, N.V.S., 1989. Visual Pattern Analyzers. Oxford University Press, Oxford.
- Green, D.A., Swets, J.A., 1988. Signal Detection Theory and Psychophysics. Krieger, Huntington, NY.
- Heeger, D.J., 1991. Nonlinear model of neural responses in cat visual cortex. In: Landy, M., Movshon, J.A. (Eds.), Computational Models of Visual Processing. MIT Press, Cambridge, MA, pp. 119–133.
- Kingdom, F.A.A., Baldwin, A.S., Schmidtmann, G., 2015. Modelling probability and additive summation for the detection across multiple mechanisms under the assumptions of signal detection theory. *J. Vis.* 15 (5).
- Kingdom, F.A.A., Prins, N., 2010. Psychophysics: A Practical Introduction. Academic Press: An Imprint of Elsevier, London.
- Laming, D., 2013. Probability summation—a critique. *J. Opt. Soc. Am. A* 30, 300–315.
- Legge, G.E., Foley, J.M., 1980. Contrast masking in human vision. *J. Opt. Soc. Am. A* 70, 1458–1471.

- Loffler, G., Wilson, H.R., Wilkinson, F., 2003. Local and global contributions to shape discrimination. *Vision Res.* 43 (5), 519–530.
- Meese, T.S., 2010. Spatially extensive summation of contrast energy is revealed by contrast detection of micro-pattern textures. *J. Vis.* 10 (8), 14.1–14.21.
- Meese, T.S., Baker, D.H., 2011. Contrast summation across eyes and space is revealed along the entire dipper function by a "Swiss cheese" stimulus. *J. Vis.* 11 (1), 23.1–23.23.
- Meese, T.S., Georgeson, M.A., Baker, D.H., 2006. Binocular contrast vision at and above threshold. *J. Vis.* 6, 1224–1243.
- Meese, T.S., Summers, R.J., 2007. Area summation in human vision at and above detection threshold. *Proc. R. Soc. B* 274, 2891–2900.
- Meese, T.S., Summers, R.J., 2009. Neuronal convergence in early contrast vision: binocular summation is followed by response nonlinearity and linear area summation. *J. Vis.* 9 (4), 7.1–7.16.
- Meese, T.S., Summers, R.J., 2012. Theory and data for area summation of contrast with and without uncertainty: evidence for a noisy energy model. *J. Vis.* 12 (11), 9.1–9.28.
- Meese, T.S., Williams, C.B., 2000. Probability summation for multiple patches of luminance modulation. *Vision Res.* 40, 2101–2113.
- Nachmias, J., 1981. On the psychometric function for contrast detection. *Vision Res.* 21, 215–223.
- Pelli, D.C., 1985. Uncertainty explains many aspects of visual contrast detection and discrimination. *J. Opt. Soc. Am. A* 2, 1508–1532.
- Prins, N., Kingdom, F.A.A., 2009. Palamedes: Matlab routines for analyzing psycho-physical data. <http://www.palamedestoolbox.org>.
- Quick, R., 1974. A vector-magnitude model of contrast detection. *Kybernetik* 16 (2), 65–67.
- Robson, J.G., Graham, N., 1981. Probability summation and regional variation in contrast sensitivity across the visual field. *Vision Res.* 21 (3), 409–418.
- Schmidtmann, G., Kennedy, G.J., Orbach, H.S., Loffler, G., 2012. Non-linear global pooling in the discrimination of circular and non-circular shapes. *Vision Res.* 62, 44–56.
- Shimozaki, S.S., Eckstein, M.P., Abbey, C.K., 2003. An ideal observer with channels versus feature-independent processing of spatial frequency and orientation in visual search performance. *J. Opt. Soc. Am. A* 20, 2197–2215.
- Simmons, D.R., Kingdom, F.A.A., 1994. Contrast thresholds for stereoscopic depth identification with isoluminant and isochromatic stimuli. *Vision Res.* 34, 2971–2982.
- Tan, K.W.S., Dickinson, J.E., Badcock, D.R., 2013. Detecting shape change: characterizing the interaction between texture-defined and contour-defined borders. *J. Vis.* 13 (14), 12.
- Tanner, W.P., Swets, J.A., 1954. A decision-making theory of visual detection. *Psychol. Rev.* 61 (6), 401–409.
- Tyler, C.W., Chen, C.-C., 2000. Signal detection theory in the 2AFC paradigm: attention, channel uncertainty and probability summation. *Vision Res.* 40, 3121–3144.
- Wilson, H., Wilkinson, F., Asaad, W., 1997. Concentric orientation summation in human form vision. *Vision Res.* 37, 2325–2330.
- Wilson, H.R., 1980. A transducer function for threshold and suprathreshold human vision. *Biol. Cybern.* 38, 171–178.

Scaling Methods*

Frederick A.A. Kingdom¹, Nicolaas Prins²

¹McGill University, Montreal, Quebec, Canada; ²University of Mississippi, Oxford, MS, USA

OUTLINE

8.1 Introduction	225	8.3.2 MLDS Applied to Paired Comparisons	243
8.2 Discrimination Scales	227	8.3.3 MLDS and Internal Noise	243
8.2.1 Fechner's Integration of Weber's Law	228	8.4 Partition Scaling	244
8.2.2 The Dipper Function	229	Further Reading	245
8.2.3 Limitations of Discrimination Scales	231	Exercise	246
8.3 Maximum Likelihood Difference Scaling (MLDS)	232	References	246
8.3.1 How MLDS Works	232		

8.1 INTRODUCTION

Perceptual scales, sometimes termed “psychological scales,” “sensory scales,” or “transducer functions,” describe the relationship between the perceived and physical magnitudes of a stimulus. Example perceptual scales are the relations between perceived contrast and physical contrast, perceived depth and retinal disparity, perceived velocity and physical velocity, and perceived transparency and physical transparency. Section 3.3.2 in Chapter 3 summarizes many of the methods available for measuring perceptual scales, and we recommend reading this section before proceeding.

*This chapter was primarily written by Frederick Kingdom.

In most cases perceptual scales characterize stimulus appearance, since they are derived from procedures with no correct and incorrect responses, in other words they are Type 2 procedures according to the taxonomy outlined in Chapter 2. For some scaling tasks this might seem counterintuitive. Take the method of paired comparisons, in which the observer is required on each trial to choose the stimulus from a pair with the greater perceived magnitude along the dimension of interest. If the growth of perceived magnitude is a monotonically increasing function of stimulus magnitude, one can legitimately argue that the observer's judgment is "correct" when the chosen stimulus is the one with the higher physical magnitude and "incorrect" when the chosen stimulus is the one with the lower physical magnitude. The argument does not hold, however, for scales that are not monotonic, for example the circle of colors that comprise a color wheel. Moreover, for scaling methods involving comparisons of stimulus differences, such as the methods of triads or quadruples, it is meaningless to consider the observer's responses in terms of being correct or incorrect unless the scale is perfectly linear, which in most cases it is not.

Not all perceptual scales, however, are derived from appearance-based judgments. Fechnerian or discrimination scales, of which more will be said shortly, are derived by integrating just-noticeable-differences, or JNDs (specifically increment thresholds), across the stimulus range and are therefore performance-based.

To illustrate the general principle of a perceptual scale consider Figure 8.1. In this hypothetical example perceived stimulus magnitude, denoted by ψ , is described by a power function of physical stimulus magnitude S , i.e.,

$$\psi = aS^\tau \quad (8.1)$$

where a is an arbitrary scaling factor and τ is the exponent of the power function. Stanley Smith Stevens (1906–1973) showed that certain perceptual scales were best described as power functions, and the ones that do are said to obey "Steven's Law". If $\tau < 1$ the function is bow-shaped or "compressive," whereas if $\tau > 1$ the function accelerates or is "expansive."

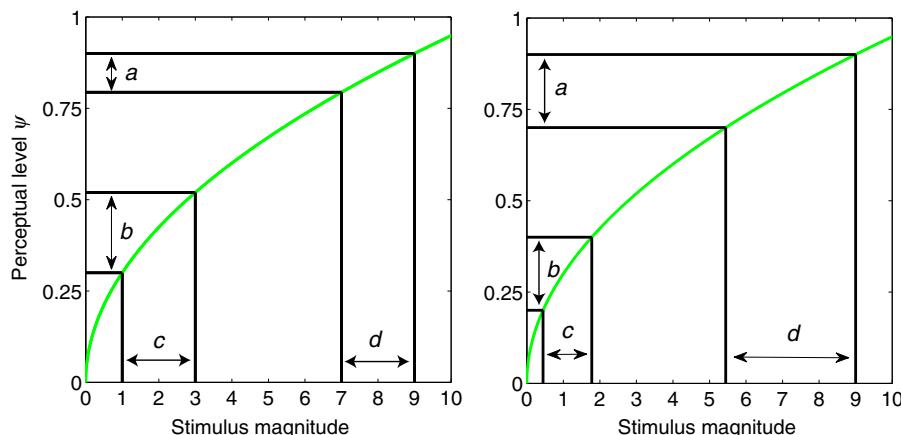


FIGURE 8.1 Hypothetical perceptual scale. Left: two pairs of stimuli (c, d) with the same physical difference (b, a) produce different values of perceived difference (b, a). Right: two pairs of stimuli with different physical difference (c, d) produce equal perceptual differences (b, a).

In [Figure 8.1](#) the exponent $\tau = 0.5$, producing a compressive function. In most scaling methods, observers make judgments about combinations of stimuli selected from various parts of the stimulus range. For example, in the method of quadruples, observers are shown on each trial two pairs of stimuli and must choose the pair that appears most similar (or most different). The left graph in [Figure 8.1](#) illustrates how two pairs of stimuli, with magnitudes of 1 and 3, and 7 and 9, respectively, will differ in their perceived similarity (or difference) owing to the compressive nature of the scale. In this example the 7,9 pair appears to be more similar than the 1,3 pair. Conversely, the graph on the right shows how two pairs of stimuli that differ in physical difference may appear equal in perceived difference.

In the rest of this chapter we concentrate on three types of perceptual scale: performance-based discrimination scales and appearance-based difference and partition scales. In the difference scale section we describe in some detail the principles and practice of a scaling method known as Maximum Likelihood Difference Scaling, or MLDS ([Maloney and Yang, 2003](#)). MLDS is a relatively new method and has a number of attractive features. It avails itself to forced-choice procedures, uses state-of-the-art computer optimization algorithms for parameter estimation, and is robust to how the observer's internal noise changes with stimulus magnitude. The importance of this last property will emerge during the discussion of discrimination scales.

8.2 DISCRIMINATION SCALES

Intuitively, the simplest method for constructing a perceptual scale is from discrimination thresholds, or JNDs, obtained across the full range of the stimulus dimension in question. The thought experiment goes something like this. Start with a low, baseline stimulus level, S_1 , and measure the JND from this baseline, call this ΔS_1 . Now set the second baseline, S_2 , to the first baseline plus ΔS_1 , i.e., $S_2 = S_1 + \Delta S_1$, and measure a new JND, ΔS_2 . Now set the third baseline S_3 to the second baseline plus ΔS_2 , i.e., $S_3 = S_2 + \Delta S_2$, and measure the next JND and so on. Eventually you will end up with a set of baseline S s separated by ΔS s, covering the whole stimulus range. Next, create a graph with an ordinate of equally spaced points for the perceptual levels $\psi_1 \dots \psi_N$, and on the abscissa put the corresponding measured baselines $S_1 \dots S_N$. Now join up the points and you will have a discrimination scale. [Figure 8.2](#) shows the principle applied to the Power Law example, though in practice one would normally need far more JNDs to cover the stimulus range than is shown, since JNDs are usually very small.

Of course, if all you had recorded in your experiment were the JNDs, then you could instead derive your perceptual scale by adding them one at a time, in other words by integration. Thus if ΔS_n represents the JND for the n th baseline stimulus level, then the corresponding perceptual scale value ψ_n is given by

$$\psi_n = \sum_{i=1}^n \Delta S_i \quad (8.2)$$

In practice an experiment along these lines would be problematic for a number of reasons. First, since each baseline S can only be determined after the JND from the previous baseline has been measured, the JNDs have to be measured in strict order of increasing (or decreasing)

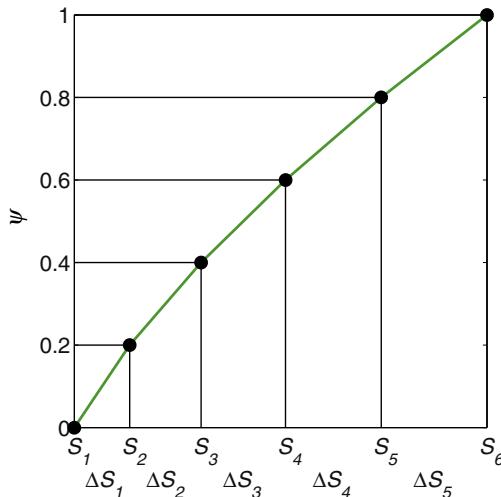


FIGURE 8.2 Constructing a discrimination scale by summing JNDs. $S_{1\dots 6}$ are baselines and $\Delta S_{1\dots 5}$ are discrimination thresholds.

S. If there were practice effects, the JNDs at the end of the scale would be smaller than otherwise, biasing its shape. Second, the errors associated with each JND will tend to accumulate as one progresses to higher baselines, causing the perceptual scale estimates to stray increasingly from their “true” values. Finally, if the JNDs are very small one would need to measure a large number of them in order to construct a scale spanning the full stimulus range.

An alternative approach that gets around these problems is to preselect a relatively small number of baseline S s spaced linearly or geometrically across the whole stimulus range. They can then be presented in random order. The JNDs can then be plotted as a function of the baseline levels and a function fitted to the plot. The discrimination scale can then be derived by mathematical integration of the function. The origin of this approach to deriving discrimination scales lies with Gustav Fechner, who famously derived a scale from JNDs that obeyed a function known as Weber’s Law. Weber’s Law states that JNDs are proportional to the baseline stimulus magnitude. Examples of Weber’s Law abound in all sensory modalities. In vision, the best-known example is to be found in the dimension of luminance: increment thresholds for patches on a uniform background at photopic light levels are proportional to background luminance. Fechner showed that if JNDs obeyed Weber’s Law, the underlying perceptual scale could be approximated by a logarithmic function (Fechner, 1860/1966; Gescheider, 1997). Since Fechner’s approach can be generalized to any type of scale (e.g., Kingdom and Moulden, 1991), it is worth working through his reasoning in detail.

8.2.1 Fechner’s Integration of Weber’s Law

According to Weber’s Law, illustrated on the left in Figure 8.3, ΔS is proportional to stimulus magnitude S , i.e.,

$$\Delta S = kS \quad (8.3)$$

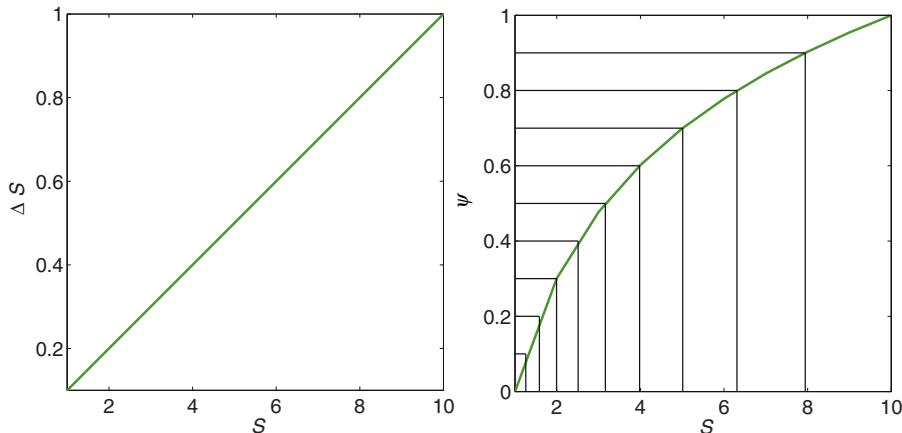


FIGURE 8.3 Left: Weber’s Law: ΔS is proportional to S . Right: Fechner’s Law: ψ is proportional to the logarithm of S . On the right, the intervals on the abscissa between S s or ΔS s increase proportionately with S . When these are mapped onto equal perceptual intervals via the horizontal lines the function mapped out is logarithmic.

or

$$\frac{\Delta S}{S} = k$$

where k is the constant of proportionality. We assume that each JND produces a constant difference in the perceptual response, call this $\Delta\psi$. Thus

$$\Delta\psi = k' \frac{\Delta S}{S} \quad (8.4)$$

where k' is the new constant. In the limit, i.e., for infinitesimally small values of $\Delta\psi$ and ΔS this becomes

$$d\psi = k' \frac{1}{S} dS \quad (8.5)$$

Integrating both sides of this equation, i.e.,

$$\int d\psi = k' \int \frac{1}{S} dS$$

gives

$$\psi = k' \ln S + C \quad (8.6)$$

where C is the constant of integration and can be assumed to be 0. The result is Fechner’s Law, illustrated on the right in Figure 8.3.

8.2.2 The Dipper Function

In visual psychophysics, “dipper functions” describe a well-known class of discrimination threshold function (see review by Solomon, 2009). They are best associated with

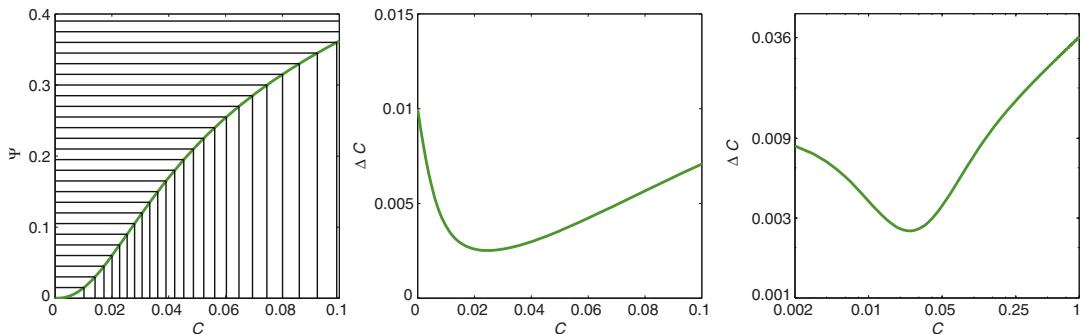


FIGURE 8.4 Left: Perceptual scale derived from Eqn (8.7) for the contrast range $C = 0\text{--}0.1$, i.e., one-tenth of the full contrast range. The black horizontal lines show the scale divided into equal intervals, the vertical lines the corresponding intervals in contrast. Middle: the contrast intervals from the left figure, ΔC , plotted as a function of C . Right: same as the middle figure except plotted over the entire range of C , i.e., $0\text{--}1$ in log-log coordinates.

discrimination thresholds for contrast. If one measures JNDs as a function of baseline, or “pedestal” contrast, say for a grating patch, you obtain a function that invariably looks like the figure on the right of Figure 8.4. The “dipper” refers to the finding that as you increase pedestal contrast from zero, JNDs first decline to a minimum—the “dipper”—then rise steeply. Although there is more than one explanation as to why one obtains the dipper (Solomon, 2009), a popular explanation is that it reflects an accelerating transducer function near threshold (Nachmias and Sansbury, 1974). In the previous section we showed how one can infer the shape of a perceptual scale from the discrimination function known as Weber’s Law. Here we will do the opposite: we will begin with the perceptual scale and see how it predicts the dipper function.

According to Legge and Foley (1980) the perceptual scale for contrast can be described by

$$\psi(C) = \frac{C^p}{z + C^q} \quad (8.7)$$

where C is Michelson contrast, and p , q , and z are constants. The left-hand graph in Figure 8.4 shows the function for the first 10% of contrast values, i.e., that range from 0 to 0.1, with p set to 2.4, q to 2.0, and z to 0.001. Note that this function differs from Steven’s Power Law in that as contrast increases from zero the function first accelerates and then decelerates, whereas with Steven’s Law it only decelerates. The acceleration tends to occur only over the first few percent of the contrast range and hence is easily missed by a scaling method that only coarsely samples the available contrast range. The degree of acceleration and deceleration in Eqn (8.7) is controlled by the parameters p and q , while z controls the point at which one switches to the other. The horizontal black lines in the figure represent threshold differences and are hence equal intervals in ψ . When the horizontal lines are mapped onto the X axis by the vertical lines, one can see that the intervals between them, which correspond to discrimination thresholds ΔC , first become smaller and then become bigger. This is the dipper. If one plots the ΔC values from this graph as a function of C one obtains the middle graph in the figure. The right-hand graph, which will be more familiar to readers, shows the same result extended to full contrast and plotted in log-log-spaced units (e.g., as in Wilson, 1980).

8.2.3 Limitations of Discrimination Scales

A fundamental limitation to discrimination scales as representing perceptual scales is that the pattern of JNDs is not only determined by the shape of the perceptual scale but also by the way the observer's internal noise varies with stimulus magnitude. Figure 8.5 illustrates the problem using the power function example. The observer's internal noise is shown as a Gaussian distribution centered on each point on the ordinate. On the left of the figure the standard deviation σ of the internal noise distribution is the same at all points—this is termed “additive” noise. Formally, the addition of additive noise to $\psi(S)$ can be expressed by the following equation:

$$\psi(S) = \alpha S^\tau + N(\sigma) \quad (8.8)$$

where the additional term $N(\sigma)$ is normally distributed noise around a mean of 0 and a standard deviation σ . Assume that each JND, measured according to some criterion level of performance (for example 0.75 proportion correct detections), is determined by the signal-to-noise ratio d' . Here, $d' = \Delta\psi / \sigma$ where $\Delta\psi$ is the criterion difference in internal response. The resulting JNDs for two points on the perceptual scale are the points on the abscissa labelled a and b . Because the function is bow-shaped (or compressive), the JNDs will increase with stimulus magnitude as shown. Note that for illustrative purposes the JNDs are much larger than would be expected assuming that the abscissa spans the full range of the stimulus.

Now consider the figure on the right of Figure 8.5. Here the perceptual scale is linear, not bow-shaped, and the internal noise σ s are proportional to stimulus magnitude, termed “multiplicative” noise. In this case σ is not a constant but is proportional to bS^τ , where b scales the growth of the noise with stimulus magnitude. With multiplicative noise $\Delta\psi$ must increase with stimulus magnitude in order to maintain the criterion ratio of $\Delta\psi$ to σ . However, because the function is linear, not bow-shaped, the resulting JNDs end up the same as in the figure on the left. In other words a compressive perceptual scale combined with additive noise can result in

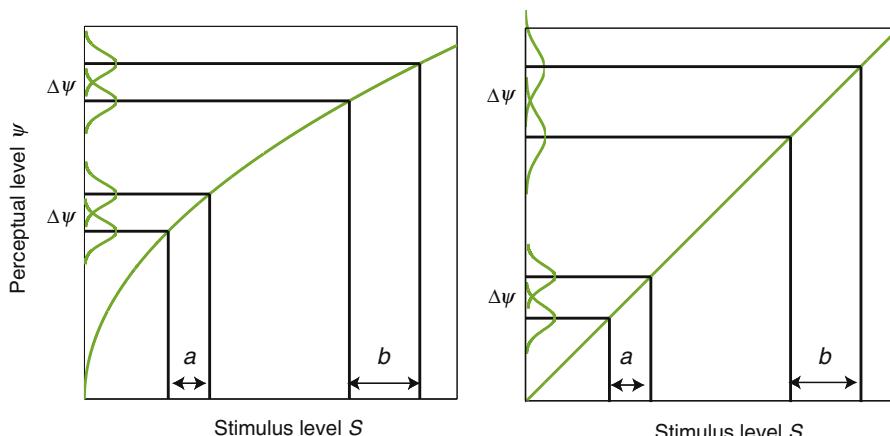


FIGURE 8.5 The impact of both the shape of the perceptual scale and the level of internal noise on JNDs. Note that the JNDs a and b are much larger than would normally be found for an abscissa that spans the full range of the stimulus dimension, for example for contrast ranging from 0 to 1. See text for details.

the same pattern of JNDs as a linear perceptual scale combined with multiplicative noise. It follows that it is impossible to derive the shape of the underlying perceptual scale from JNDs unless one knows how internal noise changes (or not) with stimulus magnitude. Put another way, if one were to assume that internal noise was additive, whereas in fact it was multiplicative, the perceptual scale estimated by integrating JNDs would not be valid. Of course internal noise is not necessarily one or other of additive or multiplicative; it may lie somewhere in between or even be a nonmonotonic function, but either way the problem exists.

The caveat to this conclusion is that if the purpose of a discrimination scale is only to define a function that predicts the pattern of JNDs, then by definition it is valid irrespective of whether the internal noise is additive or multiplicative. To conclude this section, if one wants to represent the true shape of the underlying perceptual scale, a discrimination scale is only valid if internal noise is additive.

8.3 MAXIMUM LIKELIHOOD DIFFERENCE SCALING (MLDS)

8.3.1 How MLDS Works

MLDS (Maloney and Yang, 2003) generates “interval” perceptual scales. To recap, an interval scale captures the perceived differences between scale values rather than the perceived values themselves. Suppose, for example, that a stimulus that moves with speeds of 2, 4, and 6 deg/s (degrees per second) is represented on an interval scale by values 1, 5, and 6. This would capture the fact that for this stimulus the perceived difference between 2 and 4 deg/s, a difference of four perceptual units, is four times greater than the perceived difference between 4 and 6 deg/s, a difference of one perceptual unit. The same speeds could just as well be represented by scale values of 4, 12, and 14, since these embody the same difference-relations as 1, 5, and 6. As we noted in Chapter 3, an interval scale can be transformed without loss of information by the equation $aX + b$, where X is the scale value, and a and b are constants. In Figure 8.1 the perceptual scale ranges from 0 to 1 but could be rescaled to 0–100, or 1–10, or any other range for that matter.

Although MLDS was developed for use with the method of quadruples, it can also be applied to the method of triads, and as shown in the next section the principle also applies to the method of paired comparisons. Let us then remind ourselves of these three methods, in reverse order (and see Figure 3.3). With paired comparisons the observer is presented on each trial with two stimuli, call them A and B, drawn from a larger set, and chooses the stimulus whose perceived magnitude is greater or “further along” the dimension of interest. With the method of triads the observer is presented with three stimuli on each trial, call them A, B, and C, and chooses the pair, AB or BC, with the larger (or smaller) perceived difference. With the method of quadruples the observer is presented with four stimuli on each trial, call them A, B, C, and D, and chooses the pair, AB or CD, with the larger (or smaller) perceived difference. All three methods are thus two-alternative forced-choice (2AFC), but remember that because they are appearance-based methods there are no correct or incorrect responses.

In order to use MLDS with any of these methods the stimulus space must be sampled in such a way that for any particular stimulus combination not every trial produces the same response. In other words, for any given stimulus combination, we want the observer to

choose one of the two alternatives only a proportion of times. If a particular stimulus combination always resulted in the same response, MLDS would only produce an ordinal scale.

We begin with MLDS applied to the method of quadruples. Suppose we have a set of stimulus magnitudes $S_1, S_2, S_3, \dots, S_N$. We denote their corresponding perceptual scale values $\psi_1, \psi_2, \psi_3, \dots, \psi_N$. MLDS treats the set of $\psi_2, \psi_3, \dots, \psi_{(N-1)}$ scale values as free parameters that have to be estimated from the data, with ψ_1 and ψ_N being fixed at 0 and 1, respectively. On each trial of the experiment there are four different stimulus magnitudes presented to the observer in two pairs, and the observer decides which pair is more different (or more similar). Suppose on trial one, the two pairs in the quadruple are S_1S_2 and S_3S_4 , and the observer responds that the first pair, S_1S_2 , is the more different. For a given test set of perceptual scale values $\psi_1 \dots \psi_{(N)}$, MLDS calculates the probability that the hypothetical observer characterized by $\psi_1\psi_2\psi_3\psi_4$ will respond that S_1S_2 has the larger perceived difference. The result is the likelihood associated with $\psi_1\psi_2\psi_3\psi_4$ for this one trial. The calculation is then repeated for the next trial; for example, the quadruple might be $S_1S_6S_2S_4$ with associated scale values $\psi_1\psi_6\psi_2\psi_4$ from the test set, and so on until the likelihoods of all trials have been calculated. The likelihoods of all the trials are then multiplied to obtain the across-trials likelihood. The entire procedure is then repeated for a different test set of $\psi_1, \psi_3, \dots, \psi_{(N-1)}$. After searching through the parameter space in this way the test set that gives the maximum across-trials likelihood is the one chosen as the estimated perceptual scale.

Let us work through the first trial example in more detail. We start with initial guesses $\psi_1 = 0.5$, $\psi_2 = 0.7$, $\psi_3 = 0.2$, and $\psi_4 = 0.3$ and assume an internal decision noise level $\sigma_D = 0.1$. We then calculate the probability that the observer will respond “ S_1S_2 more different” given those values. First we compute a value D that corresponds to the difference-between-the-difference-between scale values. This is

$$D = |\psi_2 - \psi_1| - |\psi_4 - \psi_3| = |0.7 - 0.5| - |0.3 - 0.2| = 0.1 \quad (8.9)$$

To convert D into a probability, we first convert it to a z -score by dividing by σ_D , which for this example gives a value of 1. The area under the normal distribution below this value is then calculated, which is 0.8413. Thus we can say that for the first trial, the likelihood of the response “ S_1S_2 more different,” given the test values of ψ_1, ψ_2, ψ_3 , and ψ_4 and given a noise σ_D of 0.1, is 0.8413.

Using the same set of ψ_s and the same σ_D the algorithm proceeds similarly to calculate the likelihoods for each of the other trials, which will include all other quadruples. For example, on those trials in which the response to the $S_1S_2S_3S_4$ quadruple is “ S_3S_4 more different” the likelihood will be $1 - 0.8413 = 0.1587$ (the two likelihoods must sum to unity). Once the likelihoods have been calculated for all trials, we multiply them out to obtain their joint probability, i.e., across-trials likelihood. However, as with the calculation of likelihoods in Chapter 4, rather than multiply out the individual likelihoods across trials and then compute the logarithm of the result, we take the logarithm of each likelihood and sum across trials. Formally expressed, this is

$$LL(\psi_1, \psi_2 \dots \psi_{(N)}, \sigma_d | \mathbf{r}) = \sum_{k=1}^T \log_e p(r_k | D_k; \psi_1, \psi_2 \dots \psi_{(N)}, \sigma_D) \quad (8.10)$$

where r_k is the response (0 or 1) and D_k is the value of D on the k th trial, \mathbf{r} is the full set of responses across all trials, and T is the number of trials. The whole procedure is then repeated for other parameter sets of ψ and σ_D , not by a brute force method but by an iterative search that “homes in” on the best-fitting parameter set (see Box 4.7 for details). We then select the set that gives the largest across-trials likelihood. The result is the maximum likelihood estimates of the parameters for $\psi_1, \psi_2, \psi_3, \dots, \psi_N$. These parameters then define the perceptual scale when plotted as a function of stimulus magnitude.

Typically, with MLDS algorithms, rather than calculating the log-likelihood on a trial-by-trial basis, as described above, the scores for each quadruple are first pooled and the log-likelihood calculated for each pooled quadruple and then summed across quadruples. **Table 8.1** tabulates the results of a hypothetical experiment in which the scores have been pooled in this way. In the example there are seven stimulus levels (1...7). The set of quadruples chosen for the experiment are shown in the first column (don’t forget that the two pairs in each quadruple are presented in random order). Column r gives the number of trials out of N in which the hypothetical observer responded that the first pair in the quadruple was the more different. The scores were generated by an algorithm in which the underlying perceptual scale was a logistic, i.e., a sigmoidally-shaped function. The rest of the table shows how the log-likelihoods for each quadruple are calculated based on an initial “guess” perceptual scale of linearly spaced values, i.e., $\psi_1 = 0, \psi_2 = 0.167, \psi_3 = 0.333, \psi_4 = 0.5, \psi_5 = 0.667, \psi_6 = 0.833$, and $\psi_7 = 1.0$. The value of D is calculated as in [Eqn \(8.9\)](#), and p_D gives the probability of obtaining the response “first pair more different” given the value of D and a guess decision noise level σ_D of 0.3.

TABLE 8.1 MLDS calculations for one test set of perceptual scale values, applied to the responses to 36 quadruples in a hypothetical experiment

Q	r	N	D	p_D	LL_Q
1234	14	50	-0.000	0.500	-34.66
1235	1	50	-0.167	0.289	-17.97
1236	0	50	-0.333	0.133	-7.15
1237	1	50	-0.500	0.048	-5.44
1245	11	50	0.000	0.500	-34.66
1246	6	50	-0.167	0.289	-22.47
1247	3	50	-0.333	0.133	-12.77
1256	20	50	-0.000	0.500	-34.66
1257	19	50	-0.167	0.289	-34.15
1267	24	50	0.000	0.500	-34.66
1345	18	50	0.167	0.711	-45.84

TABLE 8.1—cont'd

<i>Q</i>	<i>r</i>	<i>N</i>	<i>D</i>	<i>p_D</i>	<i>LL_Q</i>
1346	6	50	-0.000	0.500	-34.66
1347	7	50	-0.167	0.289	-23.37
1356	21	50	0.167	0.711	-43.14
1357	25	50	-0.000	0.500	-34.66
1367	32	50	0.167	0.711	-33.25
1456	45	50	0.333	0.867	-16.51
1457	42	50	0.167	0.711	-24.26
1467	43	50	0.333	0.867	-20.26
1567	50	50	0.500	0.952	-2.45
2345	14	50	0.000	0.500	-34.66
2346	9	50	-0.167	0.289	-25.16
2347	6	50	-0.333	0.133	-18.39
2356	26	50	-0.000	0.500	-34.66
2357	17	50	-0.167	0.289	-32.36
2367	31	50	0.000	0.500	-34.66
2456	46	50	0.167	0.711	-20.67
2457	44	50	0.000	0.500	-34.66
2467	44	50	0.167	0.711	-22.47
2567	50	50	0.333	0.867	-7.15
3456	34	50	-0.000	0.500	-34.66
3457	34	50	-0.167	0.289	-47.64
3467	38	50	0.000	0.500	-34.66
3567	49	50	0.167	0.711	-17.97
4567	42	50	0.000	0.500	-34.66
<i>LL = -951.38</i>					

Q = quadruple; *r* = number of "1st pair more different" responses; *N* = total number of quadruple presentations; *D* = the difference of the difference between the associated test perceptual scale values; *p_D* = likelihood of "1st pair more different" given the test perceptual scale values; *LL_Q* = log-likelihood of responses given the test perceptual scale values.

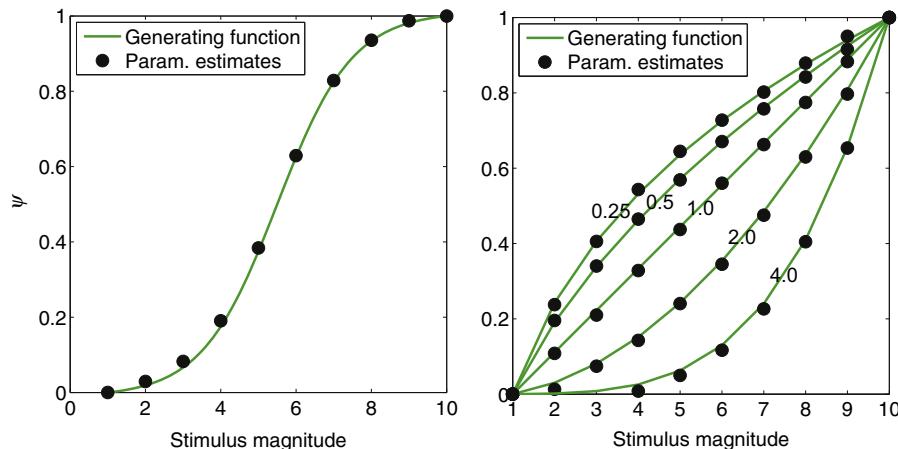


FIGURE 8.6 MLDS fitted to hypothetical logistic-shaped (left) and power-function-shaped (right) perceptual scales, the latter for five different power function exponents τ (given with each plot). The continuous green lines are NOT the fits to the data but the functions used to generate the hypothetical data that are then fitted. The fits to the data are the filled circles and represent the set of estimates of ψ .

The log-likelihood for each quadruple LL_Q is calculated as

$$LL_Q = [r \log_e p_D + (N - r) \log_e (1 - p_D)] \quad (8.11)$$

For example, for the quadruple 1247, LL_Q is calculated as $[3\log_e 0.133 + 47\log_e 0.867] = -12.77$. The log-likelihood for the set of ψ_s is calculated as the sum of the log-likelihoods across all quadruples, which for our example is -951.38 . Remember that this is the log-likelihood for just one test set of ψ_s . The algorithm then searches through different test sets of ψ_s until it finds the set producing the highest log-likelihood. The principle of maximum likelihood fitting is explained in detail in Chapter 4.

The whole of the above procedure also applies to the methods of triads D for, say, the triad $S_1S_2S_3$, which is given by

$$D = |\psi_2 - \psi_1| - |\psi_3 - \psi_2| \quad (8.12)$$

Note that ψ_2 , which corresponds to the stimulus S_2 , is common to both sides of the equation.

Figure 8.6 shows the results of applying MLDS to quadruples data generated by two different types of hypothetical perceptual scale: one logistic-shaped and the other power-function-shaped, in the case of the latter for five different exponents. The green lines are the functions used to generate the hypothetical data sets, while the filled circles are the estimates of ψ_s obtained from the MLDS fitting procedure. Full details of the Palamedes routines and demonstration scripts that implement MLDS are given in Box 8.1.

BOX 8.1**MAXIMUM LIKELIHOOD DIFFERENCE SCALING
(MLDS) WITH PALAMEDES**

The best way to understand the Palamedes MLDS routines is through the use of simulated data sets. Our simulated data consists of responses that hypothetical observers would be expected to make if their judgments were determined by an underlying perceptual scale that we specify. We can then see how MLDS, which makes no assumptions about the shape of the underlying perceptual scale, reconstructs the scale from the data. So we begin with the Palamedes routines that generate hypothetical data.

Generating Stimulus Sets for MLDS

The first step is to generate the stimulus set, and the routine that does this is `PAL_MLDS_GenerateStimList`. Note that this routine is useful not only for helping to demonstrate MLDS but also for generating stimulus lists for use in actual scaling experiments. The routine is executed as follows:

```
>> StimList = PAL_MLDS_GenerateStimList(N, NumLevels, ...
    MaxDiffBetweenLevels, NumRepeats);
```

The argument `N` defines the number of stimuli per trial and should be set to 2, 3, or 4, depending on whether one wishes to generate pairs, triads, or quadruples. `NumLevels` is the number of different stimulus magnitudes or levels. `MaxDiffBetweenLevels` is a very useful parameter that precludes the generation of stimulus combinations that are “too far apart” and that would tend to result in identical observer responses across trials. The precise meaning of the parameter depends on whether one is dealing with pairs, triads, or quadruples. With pairs, setting `MaxDiffBetweenLevels` to 3 precludes stimulus pairs that are different by more than 3 stimulus levels. So, for example, if there are 10 stimulus levels, the pairs 6 and 9, 1 and 2, and 8 and 10 will appear in the list, but the pairs 2 and 7, 5 and 9, and 3 and 10 will not. With triads, `MaxDiffBetweenLevels` sets an upper limit for the difference-between-the-difference-between stimulus levels. For example, if `MaxDiffBetweenLevels` is again set to 3, the triad 1, 6, and 9 would be allowed since $|6 - 1| - |9 - 6| < 3$ and the triad 2, 7, and 9 would be allowed since $|7 - 2| - |9 - 7| = 3$; however, the triad 2, 8, and 9 would be precluded since $|8 - 2| - |9 - 8| > 3$. The principle for quadruples is the same as for triads. Finally, the argument `NumRepeats` sets the number of repeat trials for each pair/triad/quadruple. The list of pairs/triads/quadruples generated by the routine is stored in the output matrix `StimList`. As an example, type and execute the following:

```
>> StimList = PAL_MLDS_GenerateStimList(3,6,3,1);
```

Now type and execute

```
>> StimList
```

Continued

BOX 8.1 (*cont'd*)

and the output should be:

```
StimList =
1 2 3
1 2 4
1 2 5
1 2 6
1 3 4
1 3 5
1 3 6
1 4 5
1 4 6
1 5 6
2 3 4
2 3 5
2 3 6
2 4 5
2 4 6
2 5 6
3 4 5
3 4 6
3 5 6
4 5 6
```

Confirm for yourself that the combinations listed are permissible given the value for `MaxDiffBetweenLevels`. It must be remembered that `StimList` only lists the stimulus combinations that are to be used in the experiment. The *order* in which they are presented to the observer must of course be randomized, as must also be the order of presentation of the stimuli in each combination.

How many pairs/triads/quadruples will be generated? If all possible combinations are allowed (to achieve this one simply sets `MaxDiffBetweenLevels` to $\geq \text{NumLevels}-1$), and each combination is listed in only one order (as in the routine here), the binomial coefficient provides the answer. If the number of stimulus levels is S and the number of stimuli per combination is N , the total number of unique combinations T is given by:

$$T = \frac{S!}{N!(S-N)!} \quad (\text{B8.1})$$

(remember $S! = S \times (S-1) \times (S-2) \times \dots \times 1$). Thus, in the present example where $S = 10$ and $N = 2$, T is calculated to be 45. With pairs ($N = 2$) a simpler formula that gives the same result is $(S^2 - S)/2$.

BOX 8.1 (*cont'd*)

Try generating other stimulus sets for $N = 2, 3$, and 4 . Check the number of combinations generated (use `length(StimList)`) against the number calculated using the above equation (don't forget to take into account `NumRepeats` if set to greater than 1). Then try varying `MaxDiffBetweenLevels` and observe the effect on the number of stimulus combinations.

For the next step in our hypothetical experiment, we will again use triads as an example. Execute `PAL_MLDS_GenerateStimList` with arguments 3, 10, 3, and 30. Use `StimList` to type out the list of stimulus triplets. Note that each triplet is repeated 30 times. This should enable a sufficient number of responses to be simulated for the MLDS fitting routine.

Simulating Observer Responses for MLDS

Having generated the stimulus list, the next step is to simulate the hypothetical observer's responses. Let us suppose that the underlying shape of the perceptual scale is a Logistic function, which we have seen in Chapter 4 has a sigmoidal shape. First we need to set up the hypothetical perceptual scale values that will determine the simulated responses. Type the following command:

```
>>PsiValuesGen = PAL_Logistic([5 1 0 0],[1:10]);
```

The first argument is a vector of four parameters that defines the shape of the Logistic function (see Chapter 4), and the second argument is a vector defining the stimulus levels. The output `PsiValuesGen` is a vector containing the hypothetical perceptual scale values that correspond to each stimulus level, given the perceptual scale's logistic shape. Next, we need to define a vector `OutOfNum` that lists, for each of the entries in `StimList`, how many trials are to be simulated. Since each entry in `StimList` corresponds to one trial, we fill it with 1 s.

```
>>OutOfNum = ones(1,size(StimList,1));
```

We can now generate hypothetical responses using `PAL_MLDS_SimulateObserver`. Every response is either "0" or "1," according to the following rules. For pairs, the response is "1" if the first member of each pair is perceived (hypothetically) to be of greater magnitude, otherwise the response is "0." For triads and quadruples, the response is "1" if the first pair is perceived to be more different than the second pair (or the second pair more similar than the first pair) and "0" otherwise. Execute the routine by typing:

```
>>Response = PAL_MLDS_SimulateObserver(StimList, OutOfNum, ...
PsiValuesGen, 0.3);
```

The last argument specifies the hypothetical noise standard deviation of the decision process σ_D and for the present example can be set to 0.3. This is essential. If there were no internal decision noise, our hypothetical observer's responses would be determined solely by the physical differences between stimuli on each trial and MLDS could not be used.

Continued

BOX 8.1 (*cont'd*)

Next we need to combine responses across repeat trials using the `PAL_MLDS_GroupTrialsbyX` routine. Type and execute

```
>>[StimList NumPos OutOfNum] = .....  
PAL_MLDS_...GroupTrialsbyX(StimList, Response, OutOfNum);
```

The summed responses are contained in the output parameter `NumPos`. The output parameter `OutOfNum` gives the number of trials for each stimulus combination. You might like to look at the results. To view the summed responses for each pair type and execute the following:

```
>>Results = [StimList(:,1),StimList(:,2),...  
StimList(:,3),NumPos',OutOfNum']
```

If you had generated pairs you would need to type:

```
>>Results = [StimList(:,1),StimList(:,2), NumPos',OutOfNum']
```

and if quadruples:

```
>>Results = [StimList(:,1),StimList(:,2),...  
StimList(:,3),StimList(:,4),NumPos',OutOfNum']
```

Don't forget the inverted commas after `NumPos` and `OutOfNum`, as these are needed to transpose the vectors from rows into columns. Having simulated our experiment, we can now proceed to fitting the data using MLDS.

Fitting the Data with MLDS

An important feature of MLDS is that it makes no assumptions as to the shape of the underlying perceptual scale. The parameters fitted by MLDS are not parameters of a predefined function shape, as when fitting a psychometric function (see Chapter 4). Instead, the parameters fitted by MLDS are the perceptual scale values that correspond to each stimulus level and that collectively define the perceptual scale. MLDS essentially finds the best weights for the scale values that correspond to each stimulus level (except the first and last, which are not free parameters and are set to 0 and 1). MLDS also fits a value for the decision noise. The decision noise is the error associated with each trial decision.

As with most fitting procedures, one has to make initial guesses for the free parameters. Probably the best guess is that the perceptual scale parameters are linearly spaced, although in many instances a compressive function such as the power function described earlier in the chapter will be a better guess. To make a linear scale of guesses between 0 and 1 execute the following:

```
>>PsiValuesGuess = [0:1/(NumLevels-1):1];
```

with `NumLevels` set to 10.

BOX 8.1 (*cont'd*)

We are now ready to run the MLDS fitting routine `PAL_MLDS_Fit`. It has the form:

```
>>[PsiValues SDnoise LLexitflag output] = PAL_MLDS_ ....
Fit(StimList, NumPos, OutOfNum, PsiValuesGuess, SDnoiseGuess);
```

The last new argument is the initial guess for the decision noise standard deviation σ_D . You can set this again to 0.3. The function returns a vector `PsiValues`, which contains the list of fitted parameters. The number of parameters in `PsiValues` corresponds to the number of stimulus levels, but remember that the first and last of these have already been set to 0 and 1. `SDnoise` is the estimate of the σ_D . `LL` is the log-likelihood, `exitflag` is 1 if the routine converged, 0 if it did not, and `output` is a structure that contains some details regarding the iterative search.

Finally, to obtain estimates of the errors associated with each of the estimated scale parameters, we perform a bootstrap analysis using `PAL_MLDS_Bootstrap` by typing and executing:

```
>>[SE_PsiValues SE_SDnoise] = PAL_MLDS_Bootstrap(StimList,Out ...
OfNum, PsiValues,SDnoise,400);
```

`PsiValues` and `SDnoise` contain the values that resulted from the MLDS fit. The last parameter sets the number of bootstrap iterations.

Both `PAL_MLDS_Fit` and `PAL_MLDS_Bootstrap` use an iterative search procedure. You can deviate from the default search parameters—see the help sections of relevant routines. In `PAL_MLDS_Bootstrap` the routine might fail to fit a simulated dataset. If this is the case, the routine will issue a warning and the standard errors it returns should not be used. The problem may be helped by having the routine try the fit a few more times, starting with different initial guesses for the parameters. Note that fits to some simulated datasets may never converge. This might happen especially when an experiment consists of relatively few trials or when the value of `SDnoise` is high. Box 4.7 provides a discussion about optimization issues in Palamedes.

As elsewhere in Palamedes the search algorithm employed to choose the set of parameters that produces the greatest log-likelihood is the Nelder–Mead Simplex method (see Box 4.7 for more information on the Nelder–Mead Simplex algorithm).

Plotting the Results of MLDS

First plot a green line for the Logistic function used to generate the artificial data set.

```
>>StimLevelsGenPlot = [1:0.1:9];
>>PsiValuesGenPlot = PAL_Logistic([5 1 0 0],StimLevelsGenPlot);
>>plot(StimLevelsGenPlot, PAL_Scale0to1(PsiValuesGenPlot),'g-');
>>hold on
```

Continued

BOX 8.1 (*cont'd*)

And now add in the MLDS-fitted perceptual scale values and the associated standard errors that were derived by bootstrapping:

```
>>plot(1:NumLevels, PsiValues, 'k-s');
>>for i = 2:length(SE_PsiValues)-1
    line([i i], [PsiValues(i)-SE_PsiValues(i) PsiValues(i) +...
    SE_PsiValues(i)], 'color', 'k');
end
```

The result should look something like the graph in Figure B8.1. It is important to remember that the green line in the figure is the function used to *generate* the responses in the simulated triads task and is *not* a fit to the data. The fits to the data are the open squares.

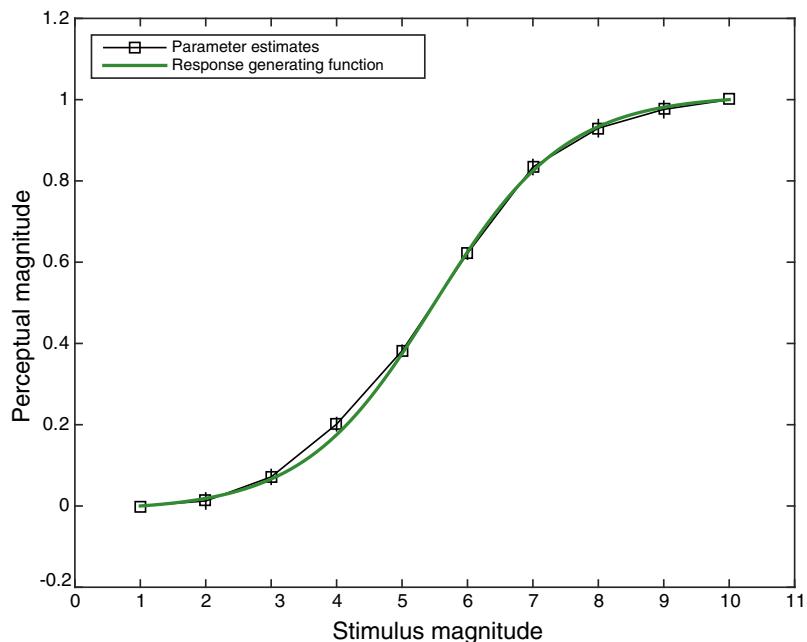


FIGURE B8.1 Output of MLDS for a simulated perceptual scale in the form of a Logistic function. The simulation used the method of triads. The continuous green line is the Logistic generating function. Open squares are the perceptual scale parameters fitted by MLDS. Error bars on each square are bootstrap errors.

Running the MLDS Demonstration Program

The various steps above can be run together as a complete sequence using the following demonstration routine in Palamedes:

```
>>PAL_MLDS_Demo
```

BOX 8.1 (*cont'd*)

The script prompts the user for the type of method (pairs, triads, or quadruples), the number of stimulus levels, the number of repeats of each stimulus combination, and the hypothetical observer's internal noise level. Thus, if these arguments are set to 3, 10, 30, and 0.3, the routine will output a graph that should look something like [Figure B8.1](#). The program outputs the number of trials the experiment simulates, which for our example should be 2820. However, this number will be different if the `MaxDiffBetweenLevels` parameter, which is set to 3 inside the program is changed.

8.3.2 MLDS Applied to Paired Comparisons

The method of paired comparisons is not strictly speaking a type of MLDS, since it is not based on a comparison of stimulus *differences*. Remember that with paired comparisons the task is to choose from a single pair of stimuli the one with the greater perceived magnitude along the dimension of interest. However, from a computational point of view, paired comparisons is simply one end of the continuum of quadruples, triads, and pairs. In terms of the procedure described above, the measure D , say for the S_1S_2 pair, is

$$D = \psi_2 - \psi_1 \quad (8.13)$$

and the set of $\psi_2, \psi_3, \dots, \psi_{(N-1)}$ scale values can be estimated in the same way using MLDS as for quadruples and triads. Since these perceptual scale estimates are obtained from proportions of response judgments, MLDS applied to paired comparisons has a formal similarity to [Thurstone's \(1927\)](#) classic method of paired comparisons (see also [Gescheider, 1997](#)).

It is important to remember, however, that with paired comparisons the number of discreet stimuli N that are needed is necessarily much larger than that required using quadruples or triads, at least if the goal is a perceptual scale that covers the whole stimulus range. This is so because the stimuli chosen for each paired comparison must differ by less than a JND, and a very large number of JNDs are typically required to cover the stimulus range.

8.3.3 MLDS and Internal Noise

Given our earlier reservations about the use of discrimination scales, the question arises as to whether MLDS is robust to whether internal noise is additive or multiplicative. Remember that the fitting procedure in MLDS not only fits a set of ψ_s' but also $\sigma_{D'}$, the internal error associated with making judgments about each pair, triad, or quadruple. In reality this error term is the sum of a number of different internal noise components. First, there is the internal noise associated with each ψ on the perceptual scale; this is the σ on the ordinate of [Figure 8.5](#). Second, there is the internal noise associated with judging the difference between stimulus levels, in other words the noise associated with $\Delta\psi_s$. With paired comparisons $\Delta\psi_s$ is the decision

variable, whereas with triads and quadruples it is an intermediate term with its own noise level. Given that perceptual distance judgments tend to be Weber-like, this second noise term is likely to be proportional to $\Delta\psi_s$, i.e., multiplicative. A third internal noise component is associated with judging the “difference-between-the-difference-between” stimulus levels, in other words the noise associated with $\Delta\Delta\psi_s$. $\Delta\Delta\psi_s$ is the decision variable for triads and quadruples and again its associated noise term will likely be multiplicative.

The extent to which MLDS is vulnerable to incorrect assumptions about these three noise components is best answered by simulation. Our own simulations reveal that with triads and quadruples MLDS is robust to whether the internal noise level associated with ψ_s and/or $\Delta\psi_s$ is additive or multiplicative, provided the internal noise levels are not implausibly large. For the internal noise associated with the decision variable $\Delta\Delta\psi_s$, [Maloney and Yang \(2003\)](#) have shown that with quadruples MLDS is similarly robust to whether the noise is additive or multiplicative. Therefore, with triads and quadruples, MLDS appears to be robust to whether all three noise components are additive or multiplicative. That is, the assumption implicit in MLDS, namely that all the noise components are additive, does not result in a misestimation of the shape of the perceptual scale if any or all of the noise components are in fact multiplicative. On the other hand, with paired comparisons our simulations show that MLDS is not robust to whether the noise added to each $\Delta\psi_s$ is additive or multiplicative. Therefore, we recommend that paired comparisons should only be used with MLDS if one can safely assume that the internal noise associated with each $\Delta\psi_s$ is additive, not multiplicative.

In the next section we will argue that the partition scaling methods described in Chapter 3 are also robust to whether the noise associated with either $\Delta\psi_s$ or $\Delta\Delta\psi_s$ is additive or multiplicative.

8.4 PARTITION SCALING

Chapter 3 described various types of partition scaling. Each involved observers adjusting the magnitude of a stimulus until it was perceptually midway between two “anchor” stimuli. Here we argue that partition scaling is a good method because like MLDS it is robust as to whether the internal noise associated with each ψ_s is additive or multiplicative.

Consider [Figure 8.7](#). In the figure, perceptual magnitude ψ , not stimulus magnitude, is shown on the abscissa. In this hypothetical example the internal noise levels are multiplicative, i.e., they increase with stimulus magnitude, as can be seen by the different σ s for the two anchor ψ_s , ψ_L for the lower anchor and ψ_U for the upper anchor. On a given trial the observer’s setting will be a point midway between ψ_L and ψ_U , plus an error ψ_e . The error term here is a combination of the internal noise associated with the partition stimulus plus a computational noise component that will likely be proportional to the perceptual distance between the anchors. Thus, the distribution of ψ_p for the partition stimulus will be determined by the σ s of the two anchor distributions and a σ associated with the partition setting. Let the distance between the means of the two anchor distributions be d . If we set the mean value of the lower anchor distribution to be 0, the setting on a given trial will be

$$\psi_p = \frac{[\psi_L + \psi_U]}{2} + \psi_e \quad (8.14)$$

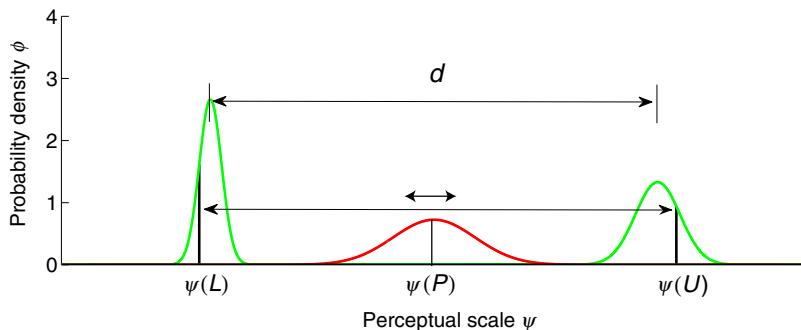


FIGURE 8.7 Effect of internal noise in a partition scaling experiment. Perceived stimulus magnitude ψ is shown on the abscissa. The green curves describe the distributions of ψ in response to the two anchors L (lower) and U (upper) as well as the distribution of ψ for the partition settings. d is the distance between the means of the anchor distributions.

One can see intuitively from Eqn (8.14) that if ψ_L , ψ_U , and ψ_P are random variables from three normal distributions, the mean of ψ_P will be $d/2$, irrespective of the variance of each distribution. This follows from the rule that the mean difference between two normal distributions is equal to the difference between their means, irrespective of their variances. This will only be true, however, if the noise is symmetric, e.g., normally distributed, but this would seem to be a reasonable assumption in most cases.

Partition scaling should therefore produce an interval perceptual scale that is robust to whether the internal noise is additive or multiplicative. By “robust” we do not mean unaffected. The reliability of each partition setting will be dependent on the amount of internal noise associated with each stimulus level as well as the computational noise associated with each partition judgment. However, it would appear that with partition scaling the derived shape of the perceptual scale will not be *systematically* shifted from its “true” shape, even if internal noise is multiplicative.

Are there any advantages to partition scaling over MLDS? MLDS requires a large number of trials, especially when using a large number of stimulus levels, a necessary requirement if the number of discriminable steps across the stimulus range is itself large, as for example with contrast (e.g., [Kingdom and Whittle, 1996](#) estimated that for periodic patterns the number of discriminable steps across the full contrast range was between 40 and 90, depending on the observer and the particular stimulus). Under these circumstances partition scaling methods might prove to be more efficient.

FURTHER READING

An excellent and user-friendly discussion of psychological scaling procedures can be found in Chapters 9 and 10 of [Gescheider \(1997\)](#). A more detailed discussion can be found in the classic text on scaling by [Torgerson \(1958\)](#). MLDS is described in [Maloney and Yang \(2003\)](#). An excellent discussion on Thurstonian scaling methods can be found in [McNicol \(2004\)](#). Multidimensional scaling techniques are described in [Borg and Groenen \(2005\)](#).

EXERCISE

1. Use Palamedes to explore the relative merits of using pairs, triads, and quadruples to establish a perceptual scale. Simulate experiments with pairs, triads, and quadruples using the same-shaped scale for generating the hypothetical observer responses, the same number of stimulus levels, the same number of trials, and the same levels of observer decision noise. Then fit the results using MLDS. Is there a difference between pairs, triads, and quadruples in how close the MLDS-fitted scale values are to the generator scale? Is there a difference in the size of the bootstrap errors?

References

- Borg, I., Groenen, P.J.F., 2005. Modern Multi-dimensional Scaling. Springer, New York, NY.
- Fechner, G.T., 1860/1966. Elements of Psychophysics. Holt, Rinehart & Winston, Inc., New York, NY.
- Gescheider, G.A., 1997. Psychophysics: The Fundamentals. Lawrence Erlbaum Associates, Mahwah, NJ.
- Kingdom, F., Moulden, B., 1991. A model for contrast discrimination with incremental and decremental test patches. *Vision Res.* 31, 851–858.
- Kingdom, F.A.A., Whittle, P., 1996. Contrast discrimination at high contrasts reveals the influence of local light adaptation on contrast processing. *Vision Res.* 36, 817–829.
- Legge, G.E., Foley, J.M., 1980. Contrast masking in human vision. *J. Opt. Soc. Am.* 70, 1458–1471.
- Maloney, L.T., Yang, J.N., 2003. Maximum likelihood difference scaling. *J. Vis.* 3, 573–585.
- McNicol, D., 2004. A Primer of Signal Detection Theory. Lawrence Erlbaum Associates, Mahwah, NJ.
- Nachmias, J., Sansbury, R., 1974. Grating contrast: discrimination may be better than detection. *Vision Res.* 14, 1039–1042.
- Solomon, J.A., 2009. Tutorial review: the history of dipper functions. *Atten. Percept. Psychophys.* 71 (3), 435–443.
- Torgerson, W.S., 1958. Theory and Methods of Scaling. Wiley, New York, NY.
- Thurstone, L.L., 1927. A law of comparative judgment. *Psychol. Rev.* 34, 273–286.
- Wilson, H.R., 1980. A transducer function for threshold and suprathreshold human vision. *Biol. Cybern.* 38, 171–178.

Model Comparisons*

Frederick A.A. Kingdom¹, Nicolaas Prins²

¹McGill University, Montreal, Quebec, Canada; ²University of Mississippi, Oxford, MS, USA

OUTLINE

9.1 Introduction	247	9.3.5 A Note on Failed Fits	295
9.2 Section A: Statistical Inference	249	9.3.6 Some Cautionary Words Regarding the Interpretation of p-Values	296
9.2.1 Standard Error Eyeballing	249		
9.2.2 Model Comparisons	252		
9.2.3 Other Model Comparisons	262	9.4 Some Alternative Model Comparison Methods	302
9.2.4 Goodness-of-Fit	264	9.4.1 Information Criteria: AIC and BIC	302
9.2.5 More than Two Conditions	268	9.4.2 Bayes Factor and Posterior Odds	304
9.3 Section B: Theory and Details	275	Further Reading	305
9.3.1 The Likelihood Ratio Test	275	Exercises	305
9.3.2 Simple Example: Fairness of Coin	275	References	306
9.3.3 Composite Hypotheses	278		
9.3.4 Specifying Models Using Reparameterization	280		

9.1 INTRODUCTION

As in any field of behavioral science, statistical tests are often required to make inferences about data. Ideally, psychophysical data would “speak for itself,” but in reality differences between psychophysical measurements obtained under different conditions are often subtle, and the consensus (particularly strong among reviewers of research articles) is that one needs criteria to judge whether the differences are “real” or not.

*This chapter was primarily written by Nicolaas Prins.

The theory of statistical testing and its application to psychophysical data is an extensive and complex topic. This chapter is not intended to be a general introduction to it or a summary of the gamut of statistical tests available for analyzing psychophysical data. That would require a book (actually several books) in itself. Rather, we explain the logic behind the likelihood ratio test, which is a statistical test that has a very general application, but we use examples taken from a particular context, namely that of testing models regarding psychometric functions (PFs). We also discuss the Palamedes toolbox (Prins and Kingdom, 2009) routines that implement the tests. Finally, we present some alternative approaches to model selection. Much of what follows is concerned with PFs, so the reader is encouraged to read at least Section A of Chapter 4 before tackling the present chapter.

Let's say you are interested in determining whether some variable X affects performance on some task. An example would be whether adaptation, such as from prolonged viewing of a stimulus, affects the visual system's sensitivity to another stimulus. The presence or absence of adaptation would be manipulated by the researcher and is considered to be the independent variable, and you would be interested in its effects on performance in a detection task, which is considered the dependent variable. We use adaptation/no-adaption as an example of an independent variable, but many others could be used, e.g., the stimuli could be fast-moving versus slow-moving, red-colored versus green-colored, large versus small, etc. Of course, the independent variable can have more than two values. For example, one may use different periods of adaptation. In order to determine whether an effect of adaptation exists, you could measure an observer's performance twice using a 2AFC paradigm, once without adaptation and once with adaptation. Let's say you use the method of constant stimuli with five stimulus contrasts and 100 trials at each stimulus contrast in both conditions, and you obtain the results shown in Table 9.1. These are hypothetical data and we have chosen to use whole numbers for the log contrast values. These numbers would be unrealistic in a real experiment but will be convenient in this and other examples.

Using the procedures discussed in Chapter 4 and implemented in the Palamedes function `PAL_PFML_Fit`, the conditions can be fitted individually with Logistic functions using a maximum likelihood criterion. The guessing rate parameter is fixed at 0.5 and the lapse rate parameter is fixed at 0, but the threshold and slope parameters are free to vary. Figure 9.1 displays the two fitted functions as well as the "raw" proportions correct for both of the conditions. It appears that the threshold estimates are quite different between the conditions. This is obvious from Figure 9.1 in that the fitted function for the "no adaptation" condition lies some way to the left of that for the adaptation condition. Specifically, the value for the

TABLE 9.1 Number of correct responses out of 100 trials (or 200 when combined) in a hypothetical experiment investigating whether adaptation to a stimulus affects the sensitivity to another stimulus

	log Contrast				
	-2	-1	0	1	2
No adaptation	61	70	81	92	97
Adaptation	59	59	67	86	91
Combined	120	129	148	178	188

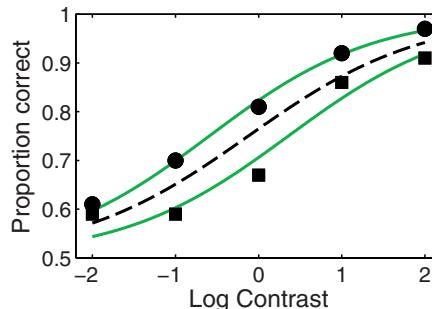


FIGURE 9.1 Proportions correct for the adaptation condition (square symbols) and no adaptation condition (round symbols), along with best-fitting logistic functions.

threshold estimate in the no-adaptation condition is -0.5946 , and in the adaptation condition it is 0.3563 . The estimates for the slope parameters, on the other hand, are very close: 1.0158 and 0.9947 for the no-adaptation and adaptation conditions, respectively. Indeed, in Figure 9.1, the two functions appear approximately equally steep.

It is tempting to conclude from these results that adaptation indeed affects performance. After all, the two fitted functions are not identical, especially with respect to the value of the threshold parameter. The problem with that logic, however, is that this may simply be due to “sampling error.” In Chapter 4 we discussed how the parameter estimates derived from experimental data are exactly that—estimates. They will not be equal in value to the “true” parameter values but rather will vary between repeated experiments due to sampling error, even if the experiments are conducted identically. In other words, the finding that the parameter estimates in the above experiment are not identical across conditions is not a surprise at all, and does not, in and of itself, mean that the underlying, true parameters have different values. This chapter will discuss procedures that are used to answer questions about the true underlying parameter values when all we have are their estimates obtained from the limited set of data from an experiment. Such procedures are commonly referred to as inferential statistics, since they deal with making inferences about parameter values from experimental data. Note that many research questions do not concern the exact value of a threshold or slope parameter per se but rather ask whether parameter values differ as a function of some independent variable. In our example above, we are not interested in the absolute level of performance with or without adaptation per se; rather, we are interested in whether this performance differs between the adaptation conditions.

9.2 SECTION A: STATISTICAL INFERENCE

9.2.1 Standard Error Eyeballing

In Chapter 4 we discussed the standard error of a parameter estimate. The proper interpretation of a standard error is somewhat complicated and is discussed in Chapter 4. For our current purposes we may loosely think of a standard error as the expected difference between the true parameter value and our estimate, based on the results of our experiment. The standard

TABLE 9.2 Parameter estimates along with their standard errors (SE) based on the raw data shown in Table 9.1 and Figure 9.1

	Threshold	SE	Slope	SE
No adaptation	-0.5946	0.2174	1.0158	0.1814
Adaptation	0.3563	0.2207	0.9947	0.2167

errors for the parameter estimates in the above experiment were estimated using the Palamedes function `PAL_PFML_BootstrapParametric` (Chapter 4). Table 9.2 lists the four parameter estimates (a threshold and slope estimate for each of the two conditions) with their standard errors.

Figure 9.2 shows the threshold parameter estimates (left panel) and the slope parameter estimates (right panel) for the two conditions in the above experiment. The vertical lines shown with each of the estimated parameters are standard error bars. Standard error bars extend from one standard error below the parameter estimate to one standard error above the parameter estimate. For example, the standard error bar of the threshold in the no-adaptation condition covers the interval -0.8120 ($-0.5946 - 0.2174$) to -0.3772 ($-0.5946 + 0.2174$).

The standard errors tell us something about the reliability of the parameter estimate. As a rule of thumb, we can be fairly confident that the value of the underlying true parameter will be within the range delineated by the standard error bars. Assuming the sampling distribution of parameter estimates is approximately normal in shape (a reasonable assumption in most practical situations) the standard error bars delineate the 68% confidence interval of the parameter estimate. Confidence intervals are discussed in some detail in Chapter 4. Briefly, the idea behind a confidence interval is that it makes explicit the notion that the parameter estimate is indeed only an estimate. It also expresses the degree of uncertainty regarding the value of the parameter in a manner that has some intuitive appeal. We say that we can be 68% confident that the true threshold in the no adaptation condition has a value between one standard error below its estimate and one standard error above it. Note that this does not mean that the probability that the value of the underlying parameter has a value within this range is 68%. The distinction between “probability” and “confidence” is explained in Chapter 4.

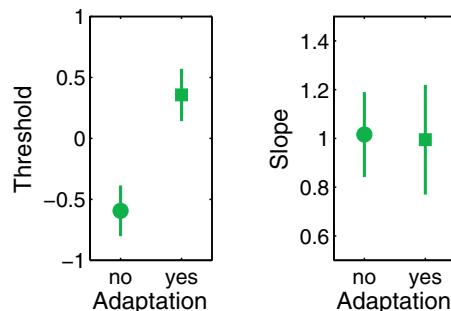


FIGURE 9.2 Graphical representation of the threshold and slope estimates and standard errors shown in Table 9.2. Standard error bars represent parameter estimate ± 1 standard error.

Given a graph that displays parameter estimates with their standard error bars, we can eyeball whether it is reasonable to attribute an observed difference between parameter estimates to sampling error alone. Remember that to say that the difference between parameter estimates is due to sampling error alone is to say that the parameter estimates were derived from identical true underlying parameters. Consider the left panel in [Figure 9.2](#), which shows the threshold estimates in the two experimental conditions with their standard error bars. Given the “confidence” interpretation of standard error bars outlined above, we can be 68% confident that the true threshold in the no adaptation condition has a value within the range delineated by its standard error bar. Similarly, we can be 68% confident that the true threshold in the adaptation condition has a value within the range delineated by its standard bar. Combining these two pieces of information, it seems reasonable to be confident that the underlying parameter values in the conditions are not equal to each other and thus that adaptation affects performance.

A very general rule of thumb to adopt is to check whether the standard error bars show any overlap between conditions. If there is no overlap, as is the case for the threshold estimates in the left panel of [Figure 9.2](#), it is usually considered reasonable to conclude that the underlying true parameters are not identical. This example would lead us to conclude that adaptation does increase the detection threshold. Consider now the right panel of [Figure 9.2](#), which shows the estimates of the slope parameters with their standard errors for the two conditions. The slope estimates differ somewhat between the conditions, but the standard error bars show a great deal of overlap. This would lead us to conclude that the observed difference in slope estimates might very well be due to sampling error and gives us little reason to suspect that the underlying true parameters are different between conditions.

Note that whereas it is considered acceptable to conclude that parameter values are “different” (as we did here with regard to the threshold parameters) we can never conclude that parameter values are “identical.” For example, we cannot conclude that the true slope parameters are identical here; it is very possible that the slope parameter in the no adaptation condition has a true value near 1.05 and in the adaptation condition a true value near 0.95. Note how we worded our conclusions regarding the slope parameters above. We never concluded that the slope parameters were the same; instead, we concluded that the observed difference in their estimates could very well have arisen by sampling error alone. We also stated that we were given little reason to suspect that the true slope parameters were different. In other words, with respect to the slope parameters, we simply do not know whether the difference between estimates is due to sampling error, to a difference in the underlying parameter values, or a combination of the two.

Many researchers will report the parameter estimates with their standard errors either in a table (as in our [Table 9.2](#)) or in a graph (as in our [Figure 9.2](#)) without any further statistical analysis. The reader is left to his or her own devices to draw conclusions as to whether an effect of some experimental manipulation is “real” (i.e., due to differences in the underlying true parameter values) or whether it could have resulted from sampling error alone. In many cases, it is quite clear whether one can reasonably conclude that parameters differ between conditions or whether such a conclusion is unwarranted. For example, given the results in [Figure 9.2](#), few will argue with the conclusion that the difference in threshold estimates is probably real, whereas the difference in slope estimates could easily be attributed to sampling

error alone. In other situations, it can be quite an art to eyeball whether an effect is “real” or might be due to sampling error.

Under some circumstances it might be useful to display not the standard errors of a parameter estimate in a figure but rather some multiple of the standard error. One that is often encountered is 1.96 (or simply 2) standard error bars. In other words, the error bars extend from 1.96 standard errors below the parameter estimate to 1.96 standard errors above the parameter estimate. If we assume that the sampling distribution of the parameter estimate is normal in shape, such error bars delineate the 95% confidence interval. Within the classical hypothesis testing framework, convention allows us to conclude that the true parameter value is not equal to any particular value outside of the 95% confidence interval. So, one might prefer “1.96 standard error bars” in case one wants to show that the parameter value is different from some particular fixed value, for example the value 0 in case the parameter represents a difference between conditions. Figure captions should always be clear as to what exactly the error bars represent.

9.2.2 Model Comparisons

This section describes the underlying logic behind the likelihood ratio test. This is a more formal procedure for determining whether differences between the PFs in different conditions are substantial enough to allow us to conclude that the parameters of the underlying true PFs are different. The problem is the same as before: even if the experimental manipulation between conditions in actuality has no effect, we still expect differences in the results between the two conditions due to random factors. So the mere existence of differences in the results between conditions does not necessarily mean that the true underlying PFs are different. The logic underlying the traditional (or “frequentist” or “Fisherian”) solution to this problem is the same for any statistical test you come across that results in a “*p*-value.” This *p*-value is the ultimate result of any frequentist statistical test. It serves as the criterion for our decision as to whether we can reasonably conclude that an experimental manipulation affects performance. Using the example experiment described above, this section will go through one of several comparisons we might make in order to explain the concept of the *p*-value and will then extend the logic to some other comparisons we could perform.

9.2.2.1 *The Underlying Logic*

We need to decide whether the observed differences among the PFs in the two conditions are real or whether they may be accounted for by sampling error alone. The specific research question is again whether adaptation affects performance on our task. Another way of looking at the question is that we aim to decide between two candidate models. One model states that adaptation does not affect performance. According to this model any observed differences in the results between the experimental conditions do not reflect a difference between the true underlying PFs but rather are due to sampling error. In other words, performance in both conditions is governed by identical underlying PFs. Let us call this model the 1 PF model. The second model states that adaptation does affect performance. Thus, differences in the results between conditions reflect differences between conditions in the performance of the underlying sensory mechanism. There are thus two different underlying true PFs, one for each condition. Let us call this model the 2 PF model.

A different way to think about the two models is that they differ in the assumptions they make. Let us list these assumptions explicitly. The 2 PF model assumes that the probability of a correct response is constant for a given stimulus level in a given condition. What this means is that the model assumes that, as the experiment progresses, the participant does not improve or get worse (due to learning or fatigue, perhaps). We called this assumption the “assumption of stability” in Chapter 4. The model also assumes independence between trials: whether the observer gets the response on a trial correct does not affect the probability that he or she will get the response on the next trial (or any other trial) correct. We called this the “assumption of independence” in Chapter 4. The assumptions of stability and independence allow us to treat all 100 trials in a particular condition and at a particular contrast identically (and combine them as we did in [Table 9.1](#) and [Figure 9.1](#)). The 2 PF model also assumes that the probability of a correct response in the no adaptation condition varies as a function of stimulus intensity in the form of a PF with a particular shape (we assumed a Logistic function on log-transformed stimulus intensities). The model assumes that, in the adaptation condition also, the probability correct varies with log-transformed stimulus levels according to a Logistic function. The Logistic functions in the two conditions do not necessarily have equal thresholds or slopes according to the 2 PF model. Let us further have the model assume that in both conditions the lapse rate equals 0 and the guess rate equals 0.5 (see Chapter 4).

The 1 PF model makes all the assumptions that the 2 PF model makes, along with some additional assumptions; therefore, it is a bit more restrictive. The additional assumptions are that the true underlying thresholds of the PFs in both conditions are identical, and that the slopes are also identical. This is a crucial characteristic of model comparisons: one of the candidate models needs to make the same assumptions as the other model and at least one additional assumption. The statistical model comparison is used to decide whether the extra assumptions that the more restrictive model makes are reasonable. Note that when we use the term “assumption” we mean a restrictive condition. One could argue that the 2 PF model makes an assumption that the 1 PF model does not, namely that the PFs are not identical between conditions. However, if we reserve the term assumption for restrictive conditions only it is the 1 PF model that makes the additional assumption. Here, we refer to the more restrictive model as the “lesser” model and the less restrictive model as the “fuller” model.

In order to perform the statistical comparison, we start by fitting the data from both conditions twice: once under the assumptions of one of the models and once under the assumptions of the other model. Let us first consider the 1 PF model, which claims that adaptation does not affect performance. Under the assumptions of this model, true performance is equal between conditions. In order to estimate the parameters of this single underlying function we should combine the trials across the conditions and fit a single PF to the results. [Table 9.1](#) shows the number of correct responses combined across conditions. Of course, the number correct is now out of 200 trials per stimulus level. We can use `PAL_PFML_Fit` to fit a PF to these data using a maximum likelihood criterion. We use a Logistic function and assume a value of 0.5 for the guess rate and a value of 0 for the lapse rate. The resulting best-fitting function is shown by the broken line in [Figure 9.1](#). It has a threshold estimate equal to -0.1251 and a slope estimate equal to 0.9544 .

The 2 PF model claims that adaptation does affect performance; thus, under the assumptions of this model, the underlying PFs for the two conditions will be different, and we should

fit each condition individually. We have already fitted this model above (Section 9.1), and the fitted functions are shown in Figure 9.1 by the green lines.

Now, which is the better model? Remember that we used the “likelihood” (Chapter 4) as the metric in which to define “best-fitting.” It might seem that all we need to do is determine which of the two models has the higher likelihood and conclude that it is that model which is the better one. That is a nice idea, but it will not work. To appreciate why, consider the following. Under the 2 PF model we fit the conditions separately, each with its own PF. The 2 PF model will fit identical PFs in the two conditions in the (extremely unlikely) case in which the proportions correct in condition 1 are identical to those in condition 2. In this case (and this case only), the fit of the 2 PF model would be identical to that of the 1 PF model, as would the likelihoods associated with the models. In case any difference in the pattern of results exists between the two conditions, be it due to a real effect or sampling error, the 2 PF model has the opportunity to fit different PFs in the two conditions in order to increase the likelihood. The 1 PF model, on the other hand, does not. It is constrained to fit a single PF to the two conditions. Thus, the 2 PF model can mimic the 1 PF model if the results in the conditions are identical but can improve its fit when the conditions are different. Another way of thinking about this is that the 1 PF model is a special case of the 2 PF model; namely, the case in which the 2 PFs of the 2 PF model happen to be identical. As a result, the likelihood under the 2 PF model will always be greater than or equal to the likelihood of the 1 PF model.

Table 9.3 shows the parameter estimates and likelihoods for both models for this example. The likelihood under the 1 PF model is 1.0763×10^{-215} , while under the 2 PF model it is 5.1609×10^{-213} . In other words, the likelihood under the single PF model is only a fraction, equal to $1.0763 \times 10^{-215} / 5.1609 \times 10^{-213} = 0.0021$, of the likelihood under the two PF model. This ratio is known as the “likelihood ratio” and is a measure of the relative fit of the two models. In cases where the results in the two conditions are exactly identical, the two model fits will also be identical, and the likelihood ratio would equal 1. In cases where the results differ between conditions, the 2 PF model will result in a higher likelihood as compared to the 1 PF model, and the likelihood ratio will be less than 1. The smaller the likelihood ratio, the worse is the fit of the 1 PF model relative to that of the 2 PF model.

The 1 PF model would have you believe that the relatively small value of the likelihood ratio in our experimental data can be explained entirely by sampling error. That is, according to this model the underlying true PFs in the two conditions are identical, and the differences

TABLE 9.3 Model fits to experimental data and to data from the first simulation

	α No adaptation	β No adaptation	α Adaptation	β Adaptation	Likelihood	LR
EXPERIMENTAL DATA						
1 PF:	-0.1251	0.9544	-0.1251	0.9544	1.0763×10^{-215}	0.0021
2 PF:	-0.5946	1.0158	0.3563	0.9947	5.1609×10^{-213}	
SIMULATION 1						
1 PF:	-0.2441	0.9224	-0.2441	0.9224	2.0468×10^{-211}	0.6326
2 PF:	-0.2082	1.0454	-0.2768	0.8224	3.2355×10^{-211}	

LR is the likelihood ratio

in actual observed performance between conditions are due to random factors only. The question is whether that is a reasonable explanation of the low value of the likelihood ratio. In other words, is it possible for an observer whose true PFs are identical between conditions to generate data resulting in such a low value of the likelihood ratio? One way to answer this question is simply to try it out. We simulate an observer who responds according to the 1 PF model and run this hypothetical observer many times through the same experiment that our human observer participated in. For every repetition we calculate the likelihood ratio based on the simulated results, and we see whether any of them are as small as that obtained from our human observer.

Specifically, in this situation, we test a simulated observer in an experiment that uses the same stimulus intensities as the experiment that our human observer participated in. The responses are generated in accordance with the 1 PF model, which describes the performance of our human observer best (i.e., under both conditions the responses are generated by the PF shown by the broken line in Figure 9.1). Table 9.4 shows the results of the first such simulated experiment.

These simulated results are plotted in Figure 9.3 together with the best-fitting PFs under the 2 PF model (green lines) and the best-fitting PF under the 1 PF model (broken line). Table 9.3 shows the parameter estimates, likelihoods, and the likelihood ratio alongside the same information for the experimental data for both models. It is important to stress that these simulated data were generated by a hypothetical observer whose responses were known to be governed by the 1 PF model. Quite clearly, the results are much more similar

TABLE 9.4 Results generated by a simulated observer acting according to the 1 PF model, which fits the data shown in Table 9.1 best and is displayed by the broken line in Figure 9.1

	log Contrast				
	-2	-1	0	1	2
Condition 1	61	63	77	89	96
Condition 2	59	71	75	87	94

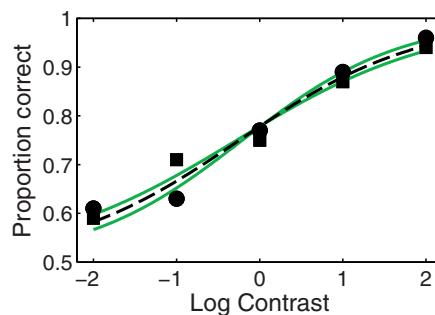


FIGURE 9.3 Data and fits of simulated experiment in which responses were generated according to the 1 PF model that was fit to the results of the human observer (i.e., the broken line in Figure 9.1).

between the two conditions as compared to those produced by our human observer. The separate PFs of the 2 PF model hardly differ from each other or from the single PF of the 1 PF model. Not surprisingly then, the likelihood ratio for the simulated data is 0.6326, much closer to one as compared to the likelihood ratio we obtained from the data of our human observer. Of course, a single simulated data set resulting in a much higher likelihood ratio than our human data does not allow us to conclude much. However, we repeated the simulation a total of 10,000 times. Figure 9.4 shows the results of the first 11 of these 10,000 simulations and the best-fitting PFs to the individual conditions. The results and fits to the experimental data are also shown again. Each of the graphs also shows the corresponding likelihood ratio.

Note from Figure 9.4 that the likelihood ratio varies systematically with the similarity between the fitted PFs in the two conditions. For example, in simulation 9 the two PFs are nearly identical and the value of the likelihood ratio is very near in value to 1. On the other hand, in simulation 3 the PFs appear quite different and the likelihood ratio is only 0.0039. However, none of the 11 simulated likelihood ratios shown in Figure 9.4 are as small as that of our experimental data. As a matter of fact, of the 10,000 simulations, only 24 resulted

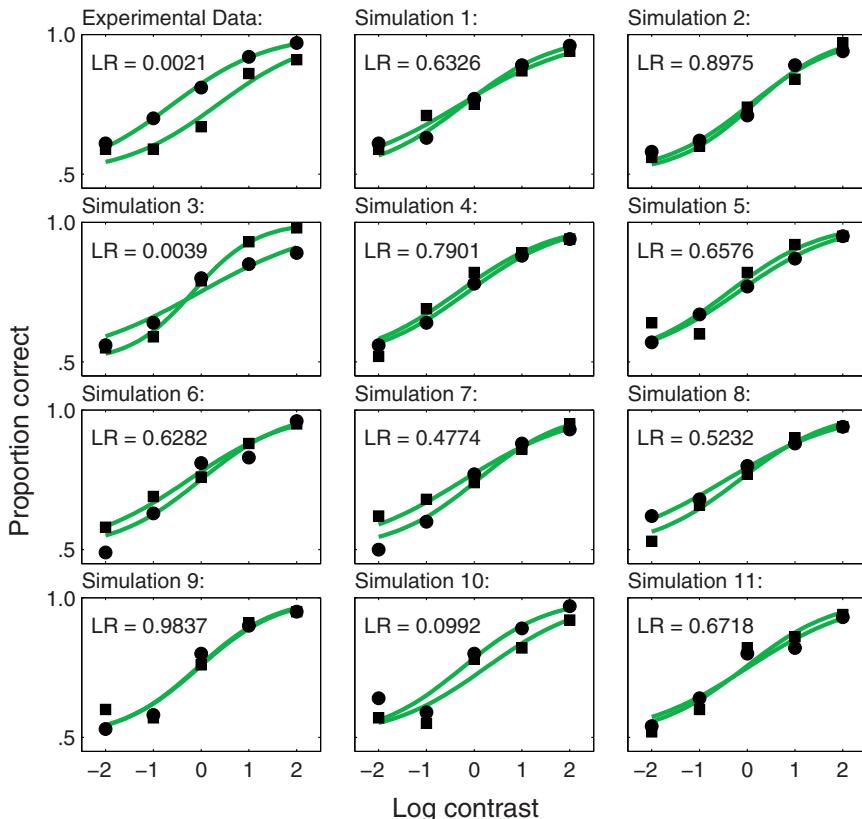


FIGURE 9.4 Experimental results as well as the results of the first 11 simulated experiments. The simulated observer acted according to the 1 PF model. Also shown are the likelihood ratios associated with each graph.

in a smaller likelihood ratio than that based on our experimental data. Apparently, it is very unlikely (24 out of 10,000 gives $p = 0.0024$) that an observer who acts according to the 1 PF model would produce a likelihood ratio as small or smaller than that produced by our human observer. It seems reasonable to conclude, then, that our human observer did not act according to the 1 PF model. The simulations indicate that we simply would not expect such a small likelihood ratio had the observer acted in accordance with the 1 PF model.

The p -value of 0.0024 we derived above is analogous to a p -value obtained from any classical, frequentist Null Hypothesis (NH) test that the reader might be more familiar with (be it a z -test, regression, ANOVA, or whatever). Any of these tests can be phrased in terms of a comparison between two models. One of the two models is always a more restrictive, “special case” version of the other. The p -value that results from such a test and is reported in journal articles always means the same thing. Roughly speaking, it is the probability of obtaining the observed data if the more restrictive model were true. If this probability is small (no greater than 5% by broad convention), we may conclude that the assumptions that the more restrictive model makes (but the alternative model does not) are incorrect. [Box 9.1](#) explains how the model comparison can be performed in the Palamedes toolbox.

BOX 9.1

MODEL COMPARISONS IN PALAMEDES

The function `PAL_PFLR_ModelComparison` in the Palamedes toolbox performs model comparisons using the likelihood ratio test. We use the example introduced in [Section 9.1](#) to demonstrate the use of the function. We specify the stimulus intensities in a matrix that has as many rows as there are experimental conditions and as many columns as there are stimulus intensities in each condition. For our example we would specify

```
>>StimLevels = [-2:1:2; -2:1:2];
```

A second matrix specifies the number of trials used at each of the stimulus levels in each condition:

```
>>OutOfNum = [100 100 100 100 100; 100 100 100 100 100];
```

A third matrix specifies the number of correct responses for each stimulus level and condition:

```
>>NumPos = [61 70 81 92 97; 59 59 67 86 91];
```

We need to specify the form of the PF we wish to use as a MATLAB® inline function:

```
>>PF = @PAL_Logistic;
```

We also create a matrix that contains values for the parameter values. For both conditions we provide the initial guesses for the free parameters and specific values to use for the fixed parameters. Unless we specify otherwise, the fuller model in the model comparison will fit separate PFs to each of the conditions in the experiment, using free threshold and slope

BOX 9.1 (*cont'd*)

parameters and fixed guess and lapse rate parameters. The lesser model will constrain threshold and slope parameters to be equal across conditions and, like the fuller model, uses fixed values for the guess and lapse rate parameters.

```
>>paramsValues = [0 1 .5 0; 0 1 .5 0];
```

Finally, we create a variable that specifies the number of simulations we wish to perform to derive our statistical *p*-value.

```
>>B = 10000;
```

We are now ready to call our function:

```
>>[TLR pTLR paramsL paramsF TLRSim converged] = ...
    PAL_PFLR_ModelComparison (StimLevels, NumPos, ...
    OutOfNum, paramsValues, B, PF);
```

pTLR will contain the proportion of simulated likelihood ratios that were smaller than the likelihood ratio obtained from the human data.

```
>>pTLR
pTLR =
0.0024
```

Note that the exact value for *pTLR* might vary a bit when we run the function again, due to the stochastic nature of the simulations. *TLR* is a transformation of the likelihood ratio of the model comparison. The transformation serves to transform the likelihood ratio into a statistic that has a known asymptotic distribution (see Section 9.3.3). *paramsL* and *paramsF* are the parameter estimates under the 1 PF and the 2 PF model, respectively.

```
>>paramsL
paramsL =
-0.1251 0.9544 0.5000 0
-0.1251 0.9544 0.5000 0
```

Note that the estimates are identical for the two conditions, which was an assumption made by the 1 PF model. Note also that these estimates are identical to those derived above by combining the results across conditions and fitting a single PF to the combined results.

```
>>paramsF
paramsF =
-0.5946 1.0158 0.5000 0
0.3563 0.9947 0.5000 0
```

BOX 9.1 (*cont'd*)

Note that these results are also identical to those derived by fitting the conditions separately ([Table 9.2](#)). `TLRsim` is a vector of length B , which contains the values of the TLRs resulting from each of the simulated experiments.

Alternative model comparisons can be performed by using optional arguments. Under the default settings, `PAL_PFLR_ModelComparison` performs the test above, which compares the 2 PF model to the 1 PF model. In other words, in the fuller model threshold parameters can take on any value in any of the conditions, as can the slope parameters. In the lesser model threshold parameters are constrained to be equal across conditions, as are the slope parameters. In both the fuller and lesser models, the guess rate and lapse rate parameters are fixed. As explained in the text, it is the assumptions that the lesser model makes but the fuller does not that are tested by the model comparison. Thus this model comparison tests whether the thresholds and slopes are equal in the conditions or whether differences exist.

In `PAL_PFLR_ModelComparison`, the statistical test to be performed is defined by specifying the two models to be compared. This approach gives the user much more flexibility than a number of predefined specific tests (e.g., one test that tests the equivalence of thresholds, another that tests the equivalence of slopes, etc.) would give. Even in this simple scenario with only two conditions there are multitudes of possible tests that may be performed. The test above determines whether there is an effect on thresholds or slopes. In the text we discuss two more specific tests: "Do the thresholds differ significantly?" and "Do the slopes differ significantly?". However, the possibilities are virtually limitless. For example, in all of the tests mentioned so far we assumed fixed values for the guess rates and lapse rates. This is not necessary. For example, we can perform all these tests while using a free lapse rate for the lapse rate of each of the conditions for both the fuller and lesser model or while using a single, free lapse rate parameter but constraining this lapse rate to be equal in the two conditions. We could test whether the thresholds differ between conditions while both the fuller and lesser models assume that the slopes are equal in the two conditions. We could perform a test as to whether the lapse rates are significantly different between the conditions. We could do so assuming equal slopes in the conditions or allowing slopes to differ. Etc., etc.

`PAL_PFLR_ModelComparison` allows the user to specify for both the fuller and the lesser models and for each of the PF's four parameters (threshold, slope, guess rate, lapse rate) whether they should be assumed to be equal to a fixed value ("fixed"), to be a free parameter that is constrained to be equal between conditions ("constrained"), or to be free to vary in each condition individually ("unconstrained"). For example, in the model comparison in [Section 9.2.2.1](#), both the fuller and lesser models assumed that guess rates and lapse rates were fixed. The fuller model assumed that the thresholds as well as the slopes were unconstrained. The lesser model assumed that thresholds and slopes were constrained.

Continued

BOX 9.1 (*cont'd*)**Testing whether Thresholds Differ**

In order to perform the model comparison that only tests whether the thresholds are equal between conditions, we need to change the default settings of `PAL_PFLR_ModelComparison`. This comparison is identical to the 2 PF versus 1 PF model test above and [Section 9.2.1](#), except that now in the lesser model only the thresholds should be constrained to be identical between conditions, but the slopes should be allowed to vary between the conditions. We can make such changes in either the fuller or lesser model's set of assumptions by providing `PAL_PFLR_ModelComparison` with optional arguments. These optional arguments come in pairs. The first argument in the pair indicates which setting we wish to change, and the second indicates the new value of the setting. The options and their default settings are given below.

Default settings:	<code>lesserThresholds: constrained</code> <code>lesserSlopes: constrained</code> <code>lesserGuessRates: fixed</code> <code>lesserLapseRates: fixed</code> <code>fullerThresholds: unconstrained</code> <code>fullerSlopes: unconstrained</code> <code>fullerGuessRates: fixed</code> <code>fullerLapseRates: fixed</code> .
-------------------	--

Our fuller model (the 2 PF model of [Section 9.2.3](#)) corresponds to the default settings. Our lesser model, however, differs from the default lesser model in that in the default lesser model the slopes are constrained while we wish them to be unconstrained. Thus, for our lesser model we need to free the slope estimates such that they can take on different values in the two conditions. We set up our variables as before, but now we call the function as follows:

```
>>[TLR pTLR paramsL paramsF TLRsim converged] = ...
    PAL_PFLR_ModelComparison (StimLevels, NumPos, OutOfNum, ...
    paramsValues, B, PF, 'lesserSlopes', 'unconstrained');
```

When we inspect the parameter estimates under the lesser model, we note that the threshold estimates are identical in value under this model, but the slope estimates are not, as we specified

```
>>paramsL
paramsL =
-0.1906 1.1560 0.5000 0
-0.1906 0.7824 0.5000 0
```

The value of `pTLR` is once again very small:

```
>>pTLR
pTLR = 0.0015
```

BOX 9.1 (*cont'd*)**Testing whether Slopes Differ**

In order to test whether slopes are affected by adaptation we call the function as follows:

```
>>[TLR pTLR paramsL paramsF TLRSim converged] = ...
    PAL_PFLR_ModelComparison (StimLevels, NumPos, OutOfNum, ...
    paramsValues, B, PF, 'lesserThresholds','unconstrained');
```

When we inspect the parameter estimates of the lesser model we note that the slope estimates are indeed equal in value between the two conditions, but the thresholds are not.

```
>>paramsL
paramsL =
-0.6001 1.0071 0.5000 0
0.3605 1.0071 0.5000 0
```

As expected, *pTLR* now has a much larger value:

```
>>pTLR
pTLR = 0.9337
```

Other optional arguments for the function *PAL_PFLR_ModelComparison* are '*lapseLimits*', '*guessLimits*', '*lapseFits*', '*gammaE0lambda*', '*searchOptions*' (all of the above are used as in *PAL_PFML_Fit*, see Box 4.2), '*maxTries*', and '*rangeTries*'. The latter two may be used to retry fits that have failed since the use of a 'search grid' (see Boxes 4.2 and 4.7) is not supported in *PAL_PFLR_ModelComparison*. In this particular example, a small percentage of fits will fail on the first try and the *maxTries* and *rangeTries* options will remedy this. *PAL_PFLR_Demo* demonstrates how to use the optional arguments. Type '*help PAL_PFLR_ModelComparison*' for information on how to use any of the optional arguments.

There are two output arguments that we haven't talked about yet, *TLRSim* and *converged*. *TLRSim* is a vector of length *B* and lists all *B* TLR values that resulted from the *B* simulations; *converged* is a vector of length *B* whose entries contain a 1 for each simulated experiment that was fit successfully and a 0 for each simulation that was not fit successfully.

In addition to using the above three options ('*fixed*', '*constrained*', and '*unconstrained*') in order to define models, the user may also specify more complex, custom models (refer to [Section B](#) of this chapter).

We apply the same logic on a regular basis in our daily lives. Compare our logic to these two examples: "If he had remembered that today is our anniversary he would probably have said something by now. Since he has not said something by now, he has probably forgotten that today is our anniversary;" or "If it was my dog Brutus that roughed up your cat Fifi,

Brutus would have probably had his face all scratched up. Since Brutus does not have his face all scratched up, it probably was not Brutus that roughed up Fifi." Compare these to our logic: "If the simpler model were true, the likelihood ratio probably would not have been as small as it was. Since it did come out as small as it was, the simpler model is probably not true." Note that, despite its tremendous intuitive appeal, the logic is in fact flawed, as a Bayesian thinker would be quick to point out. That is, we are making a statement about the probability that a model is true given our experimental results, but we do so based on the probability of obtaining our experimental results given that the model is true (See Section 4.3.3.2.1 in Chapter 4 for a more elaborate version of the Bayesian argument).

9.2.3 Other Model Comparisons

Note that the 1 PF and the 2 PF models in [Section 9.2.1](#) differ with respect to the assumptions they make about the thresholds as well as the slopes in the two conditions. We ended up deciding that the 2 PF model fit the data significantly better than the 1 PF model. What we do not know is whether this is because the thresholds, the slopes, or perhaps both differ between conditions. The 2 PF model would also have a much better fit if, for example, only the thresholds were very different between conditions, but the slopes were very similar. This, in fact, seems to be the case in the example above based on the eyeball method described in [Section 9.2.1](#) applied to [Figure 9.2](#). However, we may perform model comparisons that target more specific research questions.

Specifically, we can perform any comparison in which one of the models is a special case of the other model. In the comparison of the 1 PF and the 2 PF models we argued that the 2 PF model can always match the likelihood of the 1 PF model. It would only do so in case the results were exactly identical between the two conditions. In that case, the 2 PF would fit identical PFs to the conditions, and the likelihood under the 1 PF and 2 PF models would be identical. In case any differences exist between conditions, the 1 PF model is constrained to fit a single PF to both conditions, but the 2 PF model can accommodate the differences between conditions by fitting different PFs to the conditions. When a model is a special case of a second model, we say that it is "nested" under the second model. The likelihood ratio test is appropriate only when one of the models to be compared is nested under the alternative model.

We will now discuss two more model comparisons. Both tackle more specific research questions as compared to the above 1 PF versus 2 PF comparison. Both model comparisons compare a lesser and a fuller model where the lesser model is nested under the fuller model. The first model comparison tests whether the threshold parameters differ between conditions, and the second model comparison tests whether the slope parameters differ between conditions. Keep in mind that there are many more model comparisons that we could perform. As long as one of the models is nested under the other, we can use the likelihood ratio test to compare them.

9.2.3.1 Effect on Threshold

Perhaps you are not interested in whether adaptation has an effect on the slopes of the PF but are only interested in the effect on the thresholds. In this case you could compare a model that assumes that thresholds and slopes differ between conditions (this is the 2 PF model

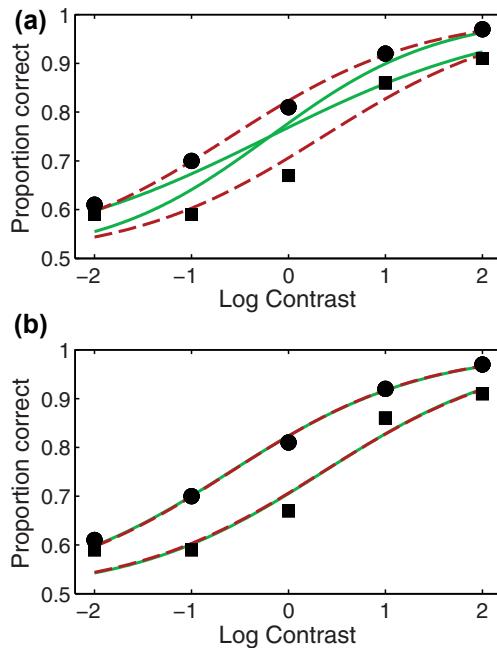


FIGURE 9.5 Two model comparisons. In (a), two models for the data in Table 9.1 are shown. The red, broken lines show the fuller model in which thresholds and slopes are free to take on any value in both of the conditions. The green, continuous lines show a lesser model that constrains thresholds to be equal in the two conditions but allows different slopes in the two conditions. (b) As in (a) except that the lesser model constrains slopes in the two conditions to be equal but allows thresholds to take on different values.

from above) to a lesser model that is identical except that it constrains only the thresholds to be identical. Note that the lesser model is once again a special case version of the fuller model. That is, under the 2 PF model, it is possible to fit PFs with equal thresholds to the two conditions. Thus, the fuller 2 PF model can do everything the lesser model can do and more.

Figure 9.5(a) shows the fuller and lesser model for this model comparison. The fuller model is the 2 PF model and is shown in red, while the lesser model is shown in green. Note that in the lesser model the functions cross at the threshold (i.e., threshold values are equal) but have different slopes. Note also that the fuller model seems to do a much better job of describing the data. When we perform the model comparison using the likelihood ratio test, we find that the p -value equals 0.0015. Thus, we conclude that the difference in threshold estimates between the conditions reflects a real difference in the underlying threshold parameters, and that adaptation does appear to affect the threshold. In Box 9.1 we explain how this model comparison can be performed in Palamedes.

9.2.3.2 Effect on Slope

Similarly, we can test whether the slopes differ significantly. Our fuller model is once again the 2 PF model, which allows thresholds and slopes to differ between conditions. Now, our lesser model constrains the slopes but allows the thresholds to vary between conditions.

Figure 9.5(b) shows the fuller and lesser model for this model comparison. The fuller model is again shown in red, and the lesser model is shown in green. Note that the lesser model is almost indistinguishable from the fuller model. They are different, however. The values for the slopes in the best-fitting fuller model were very nearly identical (1.0158 and 0.9947 for the no-adaptation and adaptation conditions, respectively). In the lesser model they are exactly identical (as the model specifies) at a value of 1.0071. Not surprisingly, the *p*-value for this model comparison is found to be 0.9337. Thus, the difference in slopes between the two conditions in the data of our human observer could easily have arisen by sampling error alone in case the lesser model was true; i.e., we have no reason to suspect that the slopes of the underlying true PFs differ between conditions. In Box 9.1 we explain how this model comparison can be performed in Palamedes.

Overall, so far, it appears based on the model comparisons that the assumption that the slopes are equal between conditions is reasonable. The assumption that the thresholds are equal, however, does not appear to be reasonable. Going back to our original research question, then, we may reasonably conclude that adaptation affects the threshold, but not the slope, of the PF.

9.2.4 Goodness-of-Fit

In the adaptation experiment example, it appears that the slopes may be equal between conditions but that the thresholds are likely not. However, remember that both models in each of the comparisons made additional assumptions. The assumptions made by all of the models above is that the probability of a correct response is constant for a particular stimulus level in a particular condition (assumptions of stability and independence), and that this probability is a function of log stimulus intensity by way of the Logistic function with a guess rate equal to 0.5 and a lapse rate equal to 0. It is very important to note that the procedure we followed to make our conclusions regarding the equality of the threshold and slope parameters is valid only insofar as these assumptions are valid. The assumptions of stability and independence are rarely made explicit in research articles, and their validity is rarely tested. The other assumptions are often explicitly verified by way of a specific model comparison that is commonly referred to as a “goodness-of-fit test.”

Although no different from any of the tests we performed above in any fundamental sense, such a model comparison is referred to as a goodness-of-fit test for reasons we hope to make evident below. A goodness-of-fit test is performed in order to test whether a particular model provides an adequate fit to some data. We briefly mentioned goodness-of-fit tests in Chapter 4, and we are now ready to discuss them in more detail. The general logic behind a goodness-of-fit test is the same as described above. A goodness-of-fit test also compares two models. Here again, one of the models is nested under the other model.

By way of example, let us determine the goodness-of-fit of the model which, so far, appears to do a good job of fitting the data obtained in our two-condition experiment above. This is the model that assumes that the slopes are identical between conditions but the thresholds are not. For the sake of brevity, we will refer to this model as the “target model.”

The target model assumes stability, independence, Logistic functions with a guess rate equal to 0.5 and a lapse rate equal to 0, and equal slopes between conditions. A goodness-of-fit test is used to test the validity of all these assumptions of the target model

simultaneously, except for the assumptions of stability and independence. It does so by comparing the target model against a model that makes only the assumptions of stability and independence. The model that assumes only stability and independence is termed the “saturated model.” In the saturated model, the parameters corresponding to the probabilities of a correct response are not constrained at all. That is, for each stimulus intensity in each of the conditions, the estimate of the probability of a correct response is free to take on any value, entirely independent of the probabilities of correct responses at other stimulus intensities or conditions. Thus, the saturated model requires the estimation of the probability of a correct response for each particular stimulus intensity in each condition. Note that the target model is nested under the saturated model. That is, under the saturated model the probabilities of a correct response are free to take on any value, including those that would collectively conform exactly to the target model. As such, the target model could not possibly produce a better fit (as measured by the likelihood) as compared to the saturated model, and thus we can perform the likelihood ratio test.

Now that we have identified our two models and made sure that one is nested under the other, we proceed exactly as we did in the tests we performed above. We simulate the experiment many times using a hypothetical observer, which we programmed to respond in accordance with the more restrictive, or lesser, target model. We fit the data of each simulated experiment twice: once under the assumptions of the target model and once under the assumptions of the saturated model. Under the saturated model, the fitting consists of finding the probability of a correct response for each condition and stimulus intensity that maximizes the likelihood. Note that estimating the probability of a correct response is much like estimating the probability that a coin will land on heads on any flip. We discussed this in Section 4.3.3.1.1 in Chapter 4 where we found that the maximum likelihood estimate of this probability corresponds to the observed proportion of heads in the experiment. Here too, the maximum likelihood estimate of the probability correct for any condition and stimulus intensity will simply be the observed proportions of correct responses. For each simulated experiment we calculate the likelihood ratio based on the fitted target model and saturated model. If the likelihood ratio computed from our experimental data seems to be similar to those obtained from the simulated experiments, it seems reasonable to conclude that our human observer acted like the target model (i.e., the target model fits the data well). If the likelihood ratio obtained from our experimental data is much lower than those typically obtained from the simulated observer, we decide that at least one of the assumptions made by the target model, but not by the saturated model, is invalid (i.e., the target model does not fit the data well).

Using 10,000 simulations, the goodness-of-fit model comparison for the target model above resulted in a p -value equal to 0.9167 (Box 9.2 demonstrates how to perform the model fit, determine standard errors on the parameters, and perform the goodness-of-fit test in Palamedes). Thus, the target model provides an excellent fit to the experimental data (after all, our experimental data produced a higher likelihood ratio than 92% of the data sets that were actually produced according to the target model).

Note that when we defined the target model we did not specify it to have the particular parameter values that we estimated from our data. That is, for each of the simulations, the parameter estimates of the lesser model were determined from the simulated data themselves. This is crucial, because if we were to force the lesser model for all the simulated data sets to have the specific parameter estimates that we derived from our data, our resulting

BOX 9.2

FITTING PFS TO MULTIPLE CONDITIONS, AND DETERMINING STANDARD ERRORS AND GOODNESS-OF-FIT IN PALAMEDES

The Palamedes function `PAL_PFML_FitMultiple` can be used to fit PFs to multiple conditions simultaneously. Its use is in many ways similar to `PAL_PFML_Fit` (Box 4.2). We will demonstrate here how to fit the model that we suspected in Section 9.2.3.2 to provide a good fit to the data. This model allows thresholds to vary between the conditions but assumes that slopes are equal in the two conditions. It further assumes fixed values for the guess and lapse rates. The syntax of `PAL_PFML_FitMultiple` is as follows:

```
[paramsValues LL exitflag output funcParams numParams] = ...
    PAL_PFML_FitMultiple(StimLevels, NumPos, OutOfNum, ...
    paramsValues, PF,{optional arguments})
```

We specify `StimLevels`, `NumPos`, `OutOfNum`, `paramsValues`, and `PF` as we did for `PAL_PFLR_ModelComparison` in Box 9.1. We can specify the model to be fitted using the optional arguments. This we accomplish in a manner similar to that used in Box 9.1. For each of the four parameters of the PF ('thresholds', 'slopes', 'guessrates', 'lapsesrates') we specify which of the three options to use: 'fixed', 'constrained', or 'unconstrained'. The option 'fixed' specifies that Palamedes should not estimate a value but instead fix the parameter values at the value provided by the user in `paramsValues`. The option 'constrained' specifies that a single, constrained value should be estimated. The option 'unconstrained' specifies that the parameter should be independently estimated for each of the conditions. In case we do not explicitly specify a choice, default choices will be used. These are 'unconstrained' for the thresholds and slopes and 'fixed' for the guess and lapse rates. Thus, we perform our fit using the following call:

```
>>[paramsValues LL exitflag output funcParams numParams] = ...
    PAL_PFML_FitMultiple(StimLevels, NumPos, OutOfNum, ...
    paramsValues, PF,'slopes','constrained')
```

The values returned in `paramsValues` are identical to those of the lesser model in the model comparison that tested whether the slopes were significantly different (Box 9.1). `LL` is the log-likelihood associated with the model; `exitflag` indicates whether the fit was successful (see Box 4.7); `output` gives some information on the Nelder–Mead search (see Box 4.7); `funcParams` is obsolete when `PAL_PFML_fitMultiple` is used, as it is in this box (but see Box 9.6); and `numParams` gives the number of free parameters in the model. For this fit, `numParams` is set to 3, since there are three free parameters (two thresholds and one shared slope).

Standard errors on the parameter estimates can be determined through bootstrap analysis using the function `PAL_PFML_BootstrapParametricMultiple`. Its syntax is as follows:

```
[SD paramsSim LLSim converged SDfunc funcParamsSim] = ...
    PAL_PFML_BootstrapParametricMultiple(StimLevels, OutOfNum, ...
    paramsValues, B, PF,{optional arguments})
```

BOX 9.2 (*cont'd*)

All common input arguments are as above, except that `paramsValues` now needs to contain the fitted parameter estimates (i.e., as returned by `PAL_PFML_FitMultiple`). `B` specifies the number of bootstrap simulations to perform. We will use 400 simulations and call the function

```
>>B = 400;
>>[SD paramsSim LLSim converged] = ...
    PAL_PFML_BootstrapParametricMultiple(StimLevels, OutOfNum, ...
    paramsValues, B, PF,'slopes','constrained');
```

`SD` is an array of the same size as `paramsValues` and contains the standard errors of the parameter estimates:

```
>> SD
SD =
0.2022 0.1355 0 0
0.2016 0.1355 0 0
```

Note that the standard errors for the slopes are equal. This is to be expected, of course, since the slope estimates for the two conditions are constrained to be equal in all simulations, as specified by the model. `paramsSim` contains the parameter estimates for all `B` simulations, `LLSim` contains the log-likelihood values for the fits of all `B` simulations, and `converged` is a vector of length `B` that indicates for all `B` simulations whether the fit converged.

The Palamedes function that performs a goodness-of-fit test when we have more than one condition is `PAL_PFML_GoodnessOffFitMultiple`. Its syntax is as follows:

```
[Dev pDev DevSim converged] = ...
    PAL_PFML_GoodnessOffFitMultiple(StimLevels, NumPos, ...
    OutOfNum, paramsValues, B, PF,{optional arguments});
```

All input arguments are as above and we are ready to call the function

```
>>[Dev pDev DevSim converged] = ...
    PAL_PFML_GoodnessOffFitMultiple(StimLevels, NumPos, ...
    OutOfNum, paramsValues, B, PF,'slopes','constrained');
```

`Dev` is the value of the Deviance (the term used for the TLR when the model comparison is a goodness-of-fit test), `pDev` is the proportion of simulated Deviances that were larger than `Dev` (i.e., this is the *p*-value for the goodness-of-fit test), `DevSim` lists the values of the Deviances for the simulated fits, and `converged` lists whether each of the `B` simulated data sets were fitted successfully. Note that if one wishes to derive a *p*-value by comparing `Dev` against the theoretical χ^2 distribution (Section 9.3.3), one may set `B = 0`, in which case no simulations will be run.

Continued

BOX 9.2 (*cont'd*)

`PAL_PFML_FitMultiple`, `PAL_PFML_BootstrapParametricMultiple`, and `PAL_PFML_GoodnessOfFitMultiple` all have some additional optional arguments, many of which are the same as those of `PAL_PFLR_ModelComparison` (see Box 9.1). Use the `help` command followed by the function name to learn more about the optional arguments.

On occasion, some of the fits to simulated datasets might not converge. Such situations might be remedied by having the routine try the fits repeatedly using initial parameter values that are randomly drawn from a range that we can specify using the optional arguments `maxTries` and `rangeTries`. Note, however, that some datasets might not be fittable at all because no local maximum exists in the likelihood function (see Box 4.7 in Chapter 4). The function `PAL_PFLR_Demo` in the PalamedesDemos folder demonstrates the use of `maxTries` and `rangeTries`.

p-value would be hard to interpret. The problem is that the model we specify should be a model of the actual underlying process. However, the parameter estimates that we derive from our data are only estimates. Their exact values are tailored to the particular set of responses we collected from our observer. If we were to test our observer again, these estimates would have different values. Thus, to include the specific values of parameter estimates in the definition of the model we wish to test is inappropriate. A general rule of thumb to follow is that the target model you wish to test should be specified before the experimental data are collected. Of course, before the data are collected there is no way to predict what the best-fitting estimates to the data will turn out to be. Thus, we do not specify their exact values in our target model.

The transformed likelihood ratio (TLR) derived in the context of a goodness-of-fit test is known as "Deviance." It is important to keep in mind, however, that despite this difference in terminology, a goodness-of-fit test is not in any fundamental sense different from any other likelihood ratio test. A goodness-of-fit test is simply a test in which the fuller model is the saturated model, which is assumption-free except for the assumptions of stability and independence. Note that the test described here is not Pearson's χ^2 test for goodness-of-fit, a test which is included in virtually every introductory statistics text used in the social sciences. Box 9.3 explains how the goodness-of-fit test based on the Deviance is related to Pearson's χ^2 test for goodness-of-fit.

9.2.5 More than Two Conditions

The likelihood ratio test may be used to compare models involving any number of conditions, just as long as one of the models is nested under the other. Imagine you wish to expand on the above experiment by testing how thresholds vary with the duration of adaptation. You repeat the experiment, now using four different durations of the adaptation period: 0, 4, 8, and 12 s. Thus, you now have four conditions. As before, in each condition you use stimulus

BOX 9.3

PEARSON'S CHI-SQUARE TEST FOR GOODNESS-OF-FIT

Most readers will be familiar with Pearson's chi-square test for goodness-of-fit. It can be used to determine whether an observed frequency distribution deviates significantly from some hypothesized distribution. Pearson's chi-square goodness-of-fit test may, for example, be used to analyze the coin-flipping experiment that we analyzed using the likelihood ratio test in [Section 9.3.2](#). There we flipped a coin 10 times in order to test the hypothesis that the coin is fair. Six of the 10 flips resulted in heads (H) and the other four resulted in tails (T). We introduced the likelihood ratio, which in essence is a measure of the relative merit of the model that states that the coin is fair as compared to the model that states that the coin is not fair. We created a sampling distribution of possible likelihood ratios that might result from an experiment in which a fair coin is flipped 10 times. This sampling distribution gave us the probability that a result as deviant as ours would be obtained from a fair coin. This probability was quite high ($p = 0.7540$). In other words, the result of our experiment could easily result from 10 flips of a fair coin, and thus we have no reason to believe that our coin is not fair. In [Section 9.3.3](#) we further introduced the TLR, which is equal to -2 times the natural log of the likelihood ratio. The greater the TLR, the more the data favor the model that states that the coin is not fair. For the experimental results obtained the TLR equals 0.4027.

Pearson's chi-square is an alternative measure by which to quantify how much better the model is that states that the coin is not fair as compared to the model that states that the coin is fair. The equation for Pearson's χ^2 is

$$\chi^2 = \sum_{k=1}^K \frac{(o_k - e_k)^2}{e_k}$$

where, for this example, k enumerates the possible categorical outcomes of an individual coin flip (H or T), K is the total number of possible outcomes (i.e., $K = 2$), o_k is the number of coin flips observed to have resulted in outcome k , and e_k is the number of coin flips expected to result in outcome k if the coin were fair. Thus, if 6 of the 10 flips were observed to result in H,

$$\chi^2 = \frac{(6 - 5)^2}{5} + \frac{(4 - 5)^2}{5} = 0.4.$$

The greater the value of χ^2 , the more the data favor the model that states that the coin is not fair.

Note that since χ^2 is a measure of how well some model fits the data, we can use it to find the model that would best fit the data. This would be the model that minimizes the value of χ^2 . For this simple example it is quite clear that the model that states that $p(H) = 0.6$ would minimize the value of χ^2 . Under this model, the expected frequencies would correspond to the observed frequencies and χ^2 would equal 0. The process of fitting a model by minimizing Pearson's χ^2 is termed minimum χ^2 fitting (e.g., [Berkson, 1980](#)).

Continued

BOX 9.3 (*cont'd*)

So, the TLR and Pearson's χ^2 are merely two different measures or statistics of the relative merit of one model as compared to another model. As it turns out, both statistics are asymptotically distributed as the continuous χ^2 distribution with degrees of freedom equal to the difference in the number of free parameters in the two models to be compared. In the context of our example, this means that with an increasing number of coin flips on which the statistic is based, the more the sampling distribution of TLR and Pearson's χ^2 will tend to resemble the theoretical χ^2 distribution with one degree of freedom. This also implies that with an increasing number of coin flips, values of TLR and Pearson's χ^2 will tend to converge. The TLR and Pearson's χ^2 obtained from our experiment were close in value (0.4027 and 0.4, respectively). Note that this would not be the case for all possible outcomes of our modest experiment. For example, had the experiment resulted in nine heads, the TLR would have been equal to 7.3613, while Pearson's χ^2 would have been equal to 6.4.

Let us now consider how Pearson's χ^2 may be extended to the fitting of a PF and determining the goodness-of-fit. Consider Figure B9.3.1. It shows the results of a hypothetical experiment that used 10 trials at each of five different stimulus levels. The green curve shows an entirely arbitrary logistic PF (it is neither the generating PF nor the best-fitting PF) that may serve as a model. It has a threshold (α) value equal to 0, slope (β) equal to 1, and guess rate (γ) as well as lapse rate (λ) equal to 0.02. According to this model, the probability of a positive response for the 10 trials presented at $x = 0$ equals 0.5. In other words, the model states that for $x = 0$, whether or not you get a positive response is like a flip of a fair coin. Given that 10 trials were used at this intensity, we would thus expect five positive responses. The observed

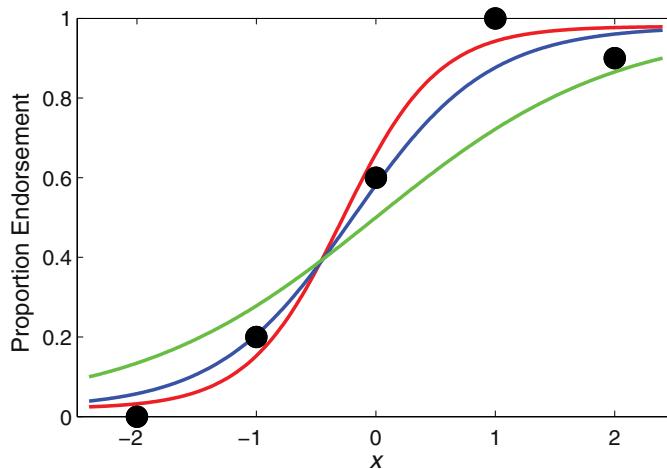


FIGURE B9.3.1 Some hypothetical data with best-fitting Logistic PF according to the minimum χ^2 criterion (blue) and the maximum likelihood criterion (red). The green function is an arbitrary curve.

BOX 9.3 (*cont'd*)

number of positive responses was actually six. Observing six positive responses out of 10 trials is like our example above: flipping a fair coin 10 times and observing six heads. We can calculate Pearson χ^2 value for just the data observed at $x = 0$, and of course it would equal 0.4 here, too. The entire experiment is like flipping five coins each 10 times. However, according to the model that corresponds to the green curve in [Figure B9.3.1](#), only the coin at $x = 0$ is a fair coin (i.e., $p(H) = 0.5$). For example, the model states that $p(H)$ for the $x = -2$ coin equals $\psi_{\text{logistic}}(x = -2; \alpha = 0, \beta = 1, \gamma = \lambda = 0.02) = 0.1344$. The expected frequency of positive responses (or heads) out of 10 trials equals $0.1344 \times 10 = 1.344$ and the observed frequency of positive responses equals 0. The expected frequency of negative responses equals 8.656 and the observed frequency equals 10. Pearson χ^2 for this coin then would be

$$\chi^2 = \frac{(0 - 1.344)^2}{1.344} + \frac{(10 - 8.656)^2}{8.656} = 1.5527$$

In this manner, we can calculate Pearson χ^2 for each of the five coins and get an overall χ^2 by summing the χ^2 values for the individual coins. This overall χ^2 for the arbitrary model shown as the green curve in [Figure B9.3.1](#) would equal 6.2134.

In order to find the threshold and slope value (while fixing the guess and lapse rates) for the best-fitting model using the minimum χ^2 method, we would search for the parameter values that minimize the overall χ^2 . The resulting logistic PF is shown in blue in [Figure B9.3.1](#).

The reader may be more familiar with the following equation that is algebraically identical to how we calculated the overall χ^2 just now:

$$\chi^2 = \sum_{x \in X} \frac{n_x (y_x - \psi_x)^2}{\psi_x (1 - \psi_x)}$$

where n_x is the number of trials used at stimulus intensity x , X is the set of all stimulus intensities, y_x is the observed proportion of positive responses at x , and ψ_x is the probability of a positive response according to the model.

The curve shown in red in [Figure B9.3.1](#) is the best-fitting logistic curve according to the maximum likelihood criterion (Chapter 4). Note that the two curves are quite different in this example. The two methods would generally not result in curves that are as different as they are here (we selected this specific outcome of the experiment because it results in quite distinct curves for the two methods). Minimum χ^2 fitting was made popular chiefly by Joseph Berkson (e.g., [Berkson, 1980](#)). History, however, has favored maximum likelihood fitting. The information criteria ([Section 9.4.1](#)) are based on the likelihood of models, and likelihood plays a large role in Bayesian fitting (Chapter 4) and Bayesian model comparison ([Section 9.4.2](#)).

When one uses a maximum likelihood criterion to fit a model, one should determine the goodness-of-fit using the goodness-of-fit test that is based on the likelihood ratio test ([Section 9.2.4](#)). It is important to realize that even though the test statistic that results from the likelihood ratio test (i.e., the TLR or Deviance if the test is a goodness-of-fit test) is asymptotically distributed as χ^2 ; this test should not be identified as the χ^2 test for goodness-of-fit. If there is

BOX 9.3 (*cont'd*)

such a thing as the χ^2 test for goodness-of-fit, it is Pearson's χ^2 test for goodness-of-fit. Some might argue that any test that is based on a statistic that is (asymptotically) distributed as χ^2 might be referred to as an χ^2 test. However, since there are multiple ways of assessing goodness-of-fit that could then be referred to as an χ^2 test for goodness-of-fit, doing so would not be very informative. If one uses the goodness-of-fit test based on the likelihood ratio, one should report it as such. One way in which to do this is to make explicit that the test is based on the Deviance (the term Deviance is used for the TLR when the model comparison is a goodness-of-fit test).

contrasts -2 , -1 , 0 , 1 , and 2 (in logarithmic units). You use 150 trials at each stimulus intensity in each condition for a total of 3000 trials (4 adaptation durations \times 5 stimulus contrasts \times 150 trials). **Table 9.5** shows the number of correct responses for the different conditions.

One might start off by fitting a PF to each condition separately. Let us say that we are confident that fixing the guess rate of the PFs at 0.5 is appropriate. However, we are aware that observers on occasion will lapse. As discussed in Chapter 4 (Box 4.6) lapses may have a large effect on threshold and slope estimates if it is assumed that the lapse rate equals 0. Thus, we wish to make the lapse rate a free parameter. We are interested in the effect on threshold, so threshold is made a free parameter. We also wish to estimate the slope of each PF. **Table 9.5** lists the parameter estimates derived by fitting the conditions individually with a Logistic function using a maximum likelihood criterion with threshold, slope, and lapse rate being free parameters. **Figure 9.6** plots the observed proportions correct and the fitted PFs.

We note a few problems. One is that one of the lapse rate estimates is negative, and we know that the true lapse rate cannot be negative, so this is clearly not a very good estimate. We may of course constrain the lapse rate to be nonnegative (see Chapter 4). The low slope

TABLE 9.5 Number of correct responses (of 150 trials) as a function of log contrast and adaptation duration; also shown are parameter estimates (α : threshold, β : slope, λ : lapse rate) for individually fitted conditions

Adaptation duration	log Contrast					Parameter estimates		
	-2	-1	0	1	2	α	β	λ
0 s	84	92	128	137	143	-0.57	1.89	0.050
4 s	67	85	103	131	139	0.12	1.98	0.063
8 s	73	85	92	125	143	0.61	1.68	0.002
12 s	82	86	97	122	141	0.93	1.02	-0.089

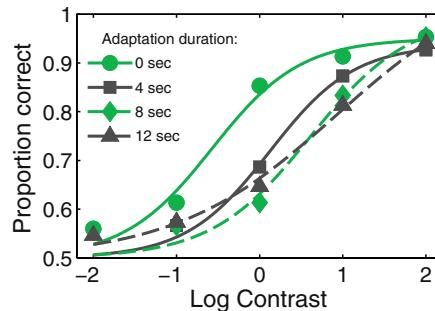


FIGURE 9.6 Proportion correct as a function of log contrast for each of four adaptation durations.

estimate in that same condition is also a bad estimate (directly related to the misestimate of the lapse rate). A second problem we face is that we cannot estimate standard errors by the bootstrap method because not all fits to simulated data sets will converge. The fundamental problem with our current strategy is that we are attempting to derive too many parameter estimates from too few observations (See Box 4.7). We need to decrease the number of free parameters or, alternatively, increase the number of observations (many-fold).

Let us try to reduce the number of free parameters. For example, the lapse rate may be assumed to be identical between conditions. Lapses occur when the probability of a correct response is independent of stimulus intensity (Section 4.3.1.1 in Chapter 4), for example when the stimulus presentation is missed altogether due to a sneeze and the observer is left to guess. There is little reason to suspect that lapse rates would vary with condition. Thus, we will constrain the lapse rate to be identical between conditions and effectively estimate a single lapse rate across all conditions. This has the obvious advantage that this single lapse rate will be based on four times the number of trials that a lapse rate fitted to an individual condition would be. We will also assume that the slope parameters are equal between conditions. Thus, we will constrain the slope parameter to be equal between conditions and estimate a single, shared, slope parameter. It is of course debatable whether it is reasonable to assume that slopes are equal between conditions, and we should consider whether this assumption seems reasonable on a case-by-case basis. Either way, we can test whether these kinds of assumptions are reasonable by performing a goodness-of-fit test (which we will do later).

We have now reduced the number of free parameters from 12 (4 thresholds, 4 slopes, 4 lapse rates) to 6 (4 thresholds, 1 slope, 1 lapse rate). The threshold estimates and their standard errors are shown in [Figure 9.7](#). When we apply the eyeball method of [Section 9.2.1](#) to [Figure 9.7](#), it seems quite clear that we may reasonably conclude that the true thresholds are not equal across all four conditions. In particular, the threshold in condition 1 is very low as compared to the others and, taking into consideration the standard errors, this appears to be a real effect (i.e., unlikely to occur by sampling error only).

Let us confirm this conclusion by performing a statistical model comparison. In order to do so, we need to define the appropriate lesser and fuller models. The fuller model is the model we have just fitted. Besides making the assumptions of stability and independence, it assumes that the underlying PFs are Logistic functions, that the slopes as well as the lapse rates are identical between conditions, and that the guess rate equals 0.5 for all four conditions.

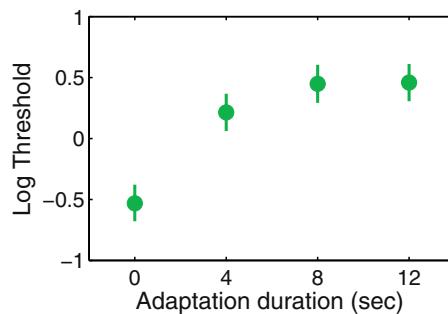


FIGURE 9.7 Plot of threshold estimates and standard errors as a function of adaptation duration (based on the hypothetical data shown in Table 9.5).

The lesser model is identical except that it makes the additional assumption that the thresholds are identical between conditions. Note that the lesser model is nested under the fuller model, which is a necessary condition for us to perform our model comparison by way of the likelihood ratio test.

When we performed the likelihood ratio test comparing these two models using 4000 simulations, the resulting p -value equaled 0. In other words, none of the 4000 experiments that were simulated in accordance with the lesser model (in which thresholds are equal) resulted in a likelihood ratio as low as that resulting from our experimental data. Of course, this is only an estimate of the actual p -value, which cannot be exactly equal to 0. It does seem that the actual p -value must be very small (likely, $p < 1/4000$). Thus, we conclude that our experimental data were not generated by an observer acting according to the lesser model. That is, adaptation duration does appear to affect threshold.

Note that the model comparison does not validate or invalidate any of the assumptions that are made by both models. These are the assumptions of stability and independence, the assumption that the true functions that describe probability of a correct response as a function of log contrast are Logistic functions, the assumption that the slopes and the lapse rates of the true PFs are identical between conditions, and the assumption that the guess rate equals 0.5. We can test all but the assumptions of stability and independence by performing a goodness-of-fit test. When we performed the goodness-of-fit test using 10,000 simulations, the p -value that resulted equaled 0.7218. In words, the data from our human observer produced a fit that was better than 72% of the data sets that were simulated in accordance with our model. Thus, we may conclude that all assumptions that our model makes, but the saturated model does not, seem reasonable. In other words, the model provides a good description of the behavior of our human observer. Once again, it is important to realize that our conclusions are only valid insofar as the assumptions that both models make (i.e., those of stability and independence) are valid.

Overall, then, our conclusion is that thresholds are affected by adaptation duration. Note that we may conclude only that not all underlying thresholds are equal (which is the assumption that the lesser model made but the fuller model did not). We may not conclude that they are all different. The situation is analogous to that which results when we reject the NH in an Analysis of Variance (ANOVA), with which the reader might be more familiar. For example, we cannot conclude that the threshold in condition 3 differs from that in condition 4.

Of course, we could perform what is often termed a “pairwise comparison.” This could simply be a model comparison between conditions 3 and 4, disregarding the other conditions (i.e., as in the two-group comparison in [Section 9.2.2](#)). There are six such pairwise comparisons that could be performed in this four-condition experiment. A disadvantage of performing pairwise comparisons in such a manner is that we lose the distinct benefit of being able to use all data in the experiment to estimate a single lapse rate or slope. Of course, we would still be able to estimate a single slope or lapse rate across the two conditions under consideration, but we would be disregarding the information that the conditions that are not part of the pair in the comparison hold regarding their values. We would also again be pushing the number of parameters that we are attempting to estimate from a relatively small amount of data.

There is a possibility to answer more specific questions about, for example, the thresholds, such as the question posed above (i.e., “Do we have reason to believe that the true threshold in condition 3 might differ from that in condition 4?”), without losing the advantage of basing our slope and lapse parameter estimates on all data collected in the experiment. Another research question that one might wish to consider is whether the decelerating nature of the thresholds as a function of adaptation duration is “real” or whether a linear trend suffices to describe the thresholds. Answering these more specific questions requires a bit of technical detail, and we will take up the issue and attempt to answer the above two questions in Section B of this chapter.

9.3 SECTION B: THEORY AND DETAILS

9.3.1 The Likelihood Ratio Test

All of the model comparisons in Section A were performed using what is known as the likelihood ratio test. The likelihood ratio test is a very flexible test. As long as one of the models is nested under the other and we can estimate model parameters by applying the maximum likelihood criterion, we can use the likelihood ratio test. In order to understand the details behind the likelihood ratio test, we will start off by considering a very simple example, that of coin flipping. We will then extend the logic to more complex situations.

9.3.2 Simple Example: Fairness of Coin

Let’s say that we have a particular coin that we suspect is biased. Here, we consider a coin to be biased when the probability that it will land on heads on any given flip does not equal 0.5. We perform a rather small-scale experiment, which consists of flipping the coin 10 times. The results of the 10 flips are, respectively,

HHTHTTHHTH (H: heads; T: tails).

Thus, we obtained six heads out of 10 flips. Do we have any reason to believe that the coin is not fair? In [Section 4.3.3.1.1](#) of Chapter 4 we performed the same experiment (with the same outcome) in order to illustrate the use of the likelihood function in parameter estimation. [Equation \(4.11\)](#) introduced the likelihood function:

$$L(a|\mathbf{y}) = \prod_{k=1}^N p(y_k|a) \quad (4.11)$$

where a is a potential value for our parameter of interest; $p(y_k|a)$ is the probability of observing outcome y on trial k , assuming value a for our parameter; and N is our total number of trials (here, $N = 10$).

Note again that the likelihood function is a function of a . In Chapter 4 we defined the maximum likelihood estimate of a (the parameter corresponding to the probability that the coin will land on heads on any flip) to be that value of a for which $L(a|y)$ attains its maximum value. For the results of the current experiment, the maximum likelihood occurs at $a = 0.6$, and this is the maximum likelihood estimate of α .

Currently, we are trying to decide whether the outcome of our experiment gives us any reason to believe that our coin is unfair. To put this a bit differently, we are trying to decide between two different models of the world. In one model the coin is fair, and in the other it is not. The first model is more restrictive as compared to the second because it assumes a particular value of α (0.5), whereas the second model allows α to assume any value. For this reason we refer to the models here as the lesser and fuller models, respectively. The likelihood ratio is the ratio of the likelihood under the lesser model to that of the likelihood under the fuller model, using the maximum likelihood estimate for the free parameter in the fuller model. Thus, the likelihood ratio is

$$\Lambda = \frac{L(a = 0.5|\mathbf{y})}{L(a = 0.6|\mathbf{y})} = \frac{0.5^{10}}{0.6^6 \times 0.4^4} = \frac{9.766 \times 10^{-4}}{1.194 \times 10^{-3}} = 0.8176$$

The interpretation of this value is that the probability that a fair coin would produce the exact outcome of the experiment as we observed it is a fraction, equal to 0.8176, of the probability that a coin characterized by $\alpha = 0.6$ would produce the same result. Because the lesser model is a more restrictive variant of the fuller model, the likelihood ratio must have a value in the interval between 0 and 1, inclusive. In our example, it would equal 1 in case we had flipped an equal number of heads and tails, an outcome which would have given us no reason whatsoever to suspect that our coin was unfair. The likelihood ratio equals 0 only when the outcome of the experiment is impossible under the lesser model but not the fuller model. Under our lesser model, no outcome would have been impossible. Only under two possible lesser models ($\alpha = 0$ and $\alpha = 1$) would the outcome of the experiment be an impossibility.

Note that the likelihood ratio will get smaller as the proportion of flips that land on heads in an experiment deviates more from the expected value under the lesser model. Specifically, Table 9.6 lists the six possible values of the likelihood ratio that may result from an experiment such as this, the outcomes that would result in these likelihoods, and the probabilities with which the six likelihood ratios would be obtained if one used a fair coin. Also listed for each possible outcome is the associated cumulative probability, i.e., the probability that a fair coin would produce a likelihood ratio equal to or smaller than that listed. For reasons to be discussed later, we often do not report the likelihood ratio, but rather a monotonic transformation of the likelihood ratio, which we refer to here as TLR ($TLR = -2 \log_e(\Lambda)$). Note that the likelihood ratio and TLR are functions of the results of our experiment and can thus be termed statistics. A distribution of the values of a statistic that might result from an experiment, together with the probabilities of obtaining these values and assuming a particular state of the world (here: the coin is fair or $\alpha = 0.5$), is termed a “sampling distribution” of that statistic.

TABLE 9.6 Sampling distribution and cumulative sampling distribution of the likelihood ratio (Λ_i , i enumerates possible outcomes of experiment) and TLR for an experiment consisting of 10 flips and assuming $\alpha = 0.5$

Number heads	Λ_i	TLR [$-2\log_e(\Lambda_i)$]	$p(\Lambda = \Lambda_i \alpha = 0.5)$	$p(\Lambda \leq \Lambda_i \alpha = 0.5)$
0 or 10	0.0010	13.8629	0.0020	0.0020
1 or 9	0.0252	7.3613	0.0195	0.0215
2 or 8	0.1455	3.8549	0.0879	0.1094
3 or 7	0.4392	1.6457	0.2344	0.3438
4 or 6	0.8176	0.4027	0.4102	0.7540
5	1	0	0.2461	1

As you can see from the cumulative sampling distribution of the likelihood ratio ($p(\Lambda \leq \Lambda_i | \alpha = 0.5)$) in Table 9.6 (and likely suspected by using common sense), obtaining a likelihood ratio as low as 0.8176 is not an unexpected outcome if the coin is fair. You would expect the likelihood ratio to be that low or lower on about three of every four (0.7540) similar experiments performed with a fair coin. As such, the outcome of our experiment gives us no reason to suspect that our coin is unfair. Had 9 of the 10 flips in our experiment come up heads, our likelihood ratio would have been 0.0252. Obtaining a likelihood as low as 0.0252 in the experiment is unlikely to happen ($p = 0.0215$) when one flips a fair coin. Thus, when a coin does produce nine heads out of 10 flips, it appears reasonable to doubt the fairness of the coin.

By convention, the outcome of the experiment needs to have a probability of less than 5% of occurring in case the lesser model is true before we may conclude that the lesser model is false. Note that we need to consider not the probability of the exact outcome but rather the probability of an outcome that is at least as different from that expected under the lesser model as the outcome that is actually obtained. One way in which to look at this is that we need to establish these probabilities regarding the outcome of an experiment before the experiment actually takes place. After the experiment has taken place there is no uncertainty regarding its outcome, and calculating the probability of the observed outcome is an absurdity. Before the experiment takes place we do not suspect that the coin will flip exactly nine heads (or any other specific number); rather, we suspect that it is unfair and thus expect it will flip a number of heads that is different from five.

As a bit of an aside, people nevertheless have a strong tendency to “guesstimate” the probability of seemingly improbable events that have already occurred. In case this number comes out low, they tend to rule out the possibility of the event being random and prefer other (usually incredulous, sometimes absurd) explanations. This reasoning has immense intuitive appeal but is nevertheless invalid. Consider the following story, the gist of which is true (the details have been forgotten and made up here). One of us once witnessed a presentation by a researcher who studied twins. He related to his audience a case of two identical twins who were separated shortly after birth and grew up far apart and unaware of each other’s existence. The researcher tracked down both twins and found that both twins were chiefs

of their respective county's fire departments, and both twins drove a 1987 blue Ford pickup truck. The researcher mentioned that he had performed a quick casual estimate of the probability that two randomly selected individuals would both be chiefs of their respective county's fire departments and would both drive 1987 blue Ford pickup trucks. That number was obviously extremely low. The audience was to infer, we presume, that one's choice of occupation as well as the color, year, and make of the car one drives are genetically determined.

Had the researcher predicted beforehand that both twins would be fire department chiefs and would both drive 1987 blue Ford pickup trucks you should be impressed. However, you should also consider that to be a rather odd prediction to make. A more sensible, but still quite odd, prediction would have been that both twins would have the same occupation (whatever it may turn out to be) and would drive similar vehicles. For the sake of argument, let's say the researcher had made this latter prediction before tracking down his separated twins. Being a much less specific prediction, the probability of it occurring by chance alone is much greater than that of a prediction which specifies particular occupations and particular vehicles. However, we imagine it would still be low enough to reject chance as an explanation if the prediction was correct (at least by the rules of classical hypothesis testing).

Unfortunately, on tracking down the twins he finds that one is an accountant who drives an older model red Geo Metro and the other is a physical therapist driving a brand new blue Nissan Pathfinder. However, as it turns out, both are passionate about building World War II model airplanes, and both own a German shepherd named Gustav. Seemingly impressive as that finding would be ("what are the odds!" right?), it is indeed only seemingly so. Apparently, we would have been impressed with the twins having any two things in common. To cover his bases then, the researcher should predict that his next pair of twins has any two things in common. He could then guesstimate the probability that two people have at least two things in common by pure chance alone and hope this guesstimate turns out low. He would then track down his next pair of long-separated twins and hope they have at least two things in common. If all that comes about, by the conventions of classical hypothesis testing, he could claim that chance can be ruled out as an explanation. The problem, of course, is that the probability that two random people will have at least two things in common is not low at all. We think it is quite likely, actually.

9.3.3 Composite Hypotheses

The lesser model in the above coin example states that the coin is fair, i.e., $\alpha = 0.5$. As such, the statistical properties of the coin according to the lesser model are completely specified. This allowed us to create the sampling distribution of the likelihood ratio in [Table 9.6](#). This distribution lists all likelihood ratios that could be obtained in the experiment and for each lists the probability with which it will be obtained in case the lesser model is true. All we needed to do to obtain this distribution was to go through all possible outcomes of the experiment, determine the likelihood ratio that would result from this outcome, and determine the probability with which it would result. The resulting sampling distribution is known as an "exact sampling distribution" (a test that uses an exact sampling distribution is known as an "exact test"). A lesser model that specifies the properties of the system completely is said to represent a "simple hypothesis."

Compare this to the example with which we started off this chapter ([Section 9.2.2.1](#)). There we wished to test whether adaptation affected sensitivity to some stimulus. The lesser model stated that adaptation does not affect sensitivity and thus that behavior in both conditions is determined by a single underlying PF. However, it did not specify this PF completely. It did make some assumptions about the PF, namely that the shape is that of a Logistic function, that the guess rate is equal to 0.5, and that the lapse rate is equal to 0. However, it did not specify the value of the threshold parameter or the value of the slope parameter. A model that does not specify the properties of the system completely is said to represent a “composite hypothesis.” In such a case, we cannot create an exact sampling distribution. In order to do so, we would have to go through all possible outcomes of the experiment (e.g., one possible outcome would be that all responses are incorrect, another would be that the response on trial 1 is correct but the responses on all other trials are incorrect, etc.). For each of these outcomes we would calculate the likelihood ratio that would result. Finally, we would have to calculate for each of these possible outcomes the probability with which the outcome would be obtained. It is the latter two that cannot be determined when the lesser model represents a composite hypothesis.

However, in case the parameter space of the lesser model is a subset of the parameter space of the fuller model (i.e., the lesser model is nested under the fuller model), TLR is asymptotically distributed as the χ^2 distribution that has degrees of freedom equal to the difference in the number of free parameters in the models. A bit more formally, let $\hat{\theta}_F$ be the maximum likelihood estimates of the parameter set θ_F of the fuller model given observations y . Similarly, let $\hat{\theta}_L$ be the maximum likelihood estimates of the parameter set θ_L of the lesser model given y . Furthermore, let $\theta_L \subset \theta_F$, such that the above condition is met. The likelihood ratio is

$$\Lambda = \frac{L(\hat{\theta}_L|y)}{L(\hat{\theta}_F|y)} \quad (9.1)$$

The TLR is given as

$$\text{TLR} = -2 \times \log_e(\Lambda) \quad (9.2)$$

In case the lesser model is correct, TLR will be asymptotically distributed as χ^2 with degrees of freedom equal to the difference in the number of free parameters in θ_F and θ_L .

To be asymptotically distributed as χ^2 means that, with increasing numbers of observations, the sampling distribution of TLR will tend more and more toward the theoretical and continuous χ^2 distribution. Unfortunately, the number of observations that are necessary to obtain an acceptable approximation to the sampling distribution depends heavily on the particular circumstances. In many realistic settings the χ^2 approximation is quite poor (e.g., [Wichmann and Hill, 2001](#)).

An alternative is to create an empirical sampling distribution. In order to do this, we simulate the experiment many times, generating the responses in accordance with the lesser model. Of course, in order to perform the simulations we need a fully specified lesser model that includes values for the threshold and slope parameters. However, as discussed, in our example of [Section 9.2.2.1](#), the lesser model is not fully specified. In order to be able to generate a sampling distribution, we use the maximum likelihood estimates for the free

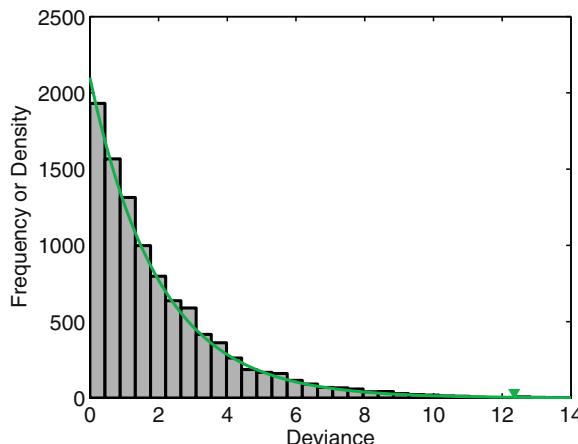


FIGURE 9.8 Empirical sampling distribution of TLR values for the model comparison discussed in Section 9.2.2.1 as well as the (appropriately scaled) theoretical χ^2 distribution with two degrees of freedom.

parameters (threshold and slope) that we obtained from our human observer to specify the behavior of the simulated observer completely. From each of the simulated experiments a TLR value is calculated in the same manner as we did for our human observer. The resulting distribution of TLR values will serve as our empirical sampling distribution. A distribution consisting of 10,000 simulated TLR values for the model comparison in Section 9.2.2.1 is shown in Figure 9.8 in the form of a histogram. Note that large TLR values indicate a poor fit of the lesser model as compared to the fuller model. This is opposite to likelihood ratio values where small likelihood ratio values are indicative of a poor fit of the lesser, relative to the fuller, model. The TLR value obtained from our experimental data was 12.35 (indicated by the green triangle in the figure). Only 24 of the 10,000 simulated TLR values were as large as that obtained from our experimental data. Thus, the human observer who generated the experimental data produced a TLR value that is quite unlikely to be obtained from an observer acting according to the lesser model. It seems reasonable to conclude, then, that our observer did not act according to the lesser model.

Note that the fuller model has four free parameters (two thresholds and two slopes) and the lesser has two free parameters (one threshold and one slope). Thus, the difference in the number of free parameters equals two. Also shown in Figure 9.8 is the (appropriately scaled) χ^2 distribution with two degrees of freedom. For this particular comparison, the $\chi^2(2)$ distribution is quite similar to our empirical sampling distribution. It may also be noted that $p(\chi^2(2) > 12.35) = 0.0021$, which is quite close to that derived from our empirical distribution (0.0024).

9.3.4 Specifying Models Using Reparameterization

Reparameterization is a tool that can be used to define models. In general terms, the idea is to use a transformation that turns the raw parameters (e.g., threshold parameters in different

conditions) into parameters that correspond to “effects.” For example, in the two-condition experiment introduced at the start of this chapter ([Section 9.1](#)) we have, of course, two threshold parameters, α_1 and α_2 . Neither of these thresholds can be used directly to investigate whether the experimental variable affects thresholds because this research question is not about the specific value of either of the thresholds. Rather, it is about whether there is a difference between the thresholds. In order to perform the model comparison that tests this question we need to fit a fuller model that allows the thresholds to differ as well as a lesser model that does not allow them to differ. The lesser model must specify that $\alpha_1 = \alpha_2$. The trick is to transform the two threshold parameters into two new parameters, one corresponding to the sum of the thresholds and the other to the difference between the thresholds. The fuller model allows both parameters to take on any value, while the lesser model allows only the sum parameter to take on any value while fixing the difference parameter at the value of 0. In [Section 9.3.4.1](#) we discuss how to transform parameters using so-called “contrasts” and provide two detailed examples of the use of contrasts. One disadvantage of contrasts is that they allow only linear transformations of parameters. In [Section 9.3.4.2](#) we use an example to illustrate how one may attain nonlinear parameter reparameterizations.

9.3.4.1 Linear Contrasts

Linear contrasts may be used to reparameterize model parameters. This allows for a convenient and flexible manner in which to specify models. As an example, let us consider the two-condition experiment discussed above. We may use the following matrix multiplication to accomplish the reparameterization:

$$\begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}$$

Each row in the matrix specifies a contrast that defines one θ parameter as a linear combination of the α parameters. θ_1 corresponds to the sum of the thresholds, whereas θ_2 corresponds to their difference. In the fuller model, both θ_1 and θ_2 are free to vary, allowing both thresholds to take on any value. In the lesser model we make θ_1 a free parameter again, but we fix θ_2 at 0. This will constrain the two thresholds to equal each other but, as a pair, to take on any value. In other words, the model comparison can also be thought of as testing whether θ_2 differs significantly from 0. [Sections 9.3.4.1.1](#) and [9.3.4.1.2](#) use examples to describe two kinds of special contrasts (“polynomial contrasts” and “Helmert contrasts”) that can each be used to address specific research questions. Essentially, these are contrasts that have been designed to fit models that can be used in model comparisons to test specific research questions. Other research questions will require different contrasts. [Box 9.4](#) gives a very brief introduction to the procedure of creating contrast matrices that can be used to target specific research questions. The reader should be aware that in this text we take some liberty by referring to the first row in the matrix above (and others below) as a contrast. Technically, it is not a contrast, since the coefficients of a proper contrast must sum to 0. A row consisting of 1s in a contrast matrix is often referred to as the “intercept,” especially in the context of the general linear model.

BOX 9.4**BRIEF INTRODUCTION TO MODEL DEFINITION
USING CONTRASTS**

Let's reiterate the simple example discussed in [Section 9.3.4](#). You obtained data in two conditions and want to determine whether there is reason to believe that the difference in threshold estimates you obtained is real or might simply be a sampling error. This comes down to a statistical comparison between two models: one in which the thresholds are forced to be identical and one in which they are free to take on different values in the two conditions. Contrasts help to define these models. Essentially, what they do is turn the two threshold parameters into two new parameters, one of which corresponds to the sum of the thresholds and the other to their difference. As discussed in [Section 9.3.4](#), the reparameterization that does this is

$$\begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}$$

The first row of the matrix is [1 1] and thus the first new parameter (let's call it θ_{sum}) is defined as $\theta_{\text{sum}} = 1 \cdot \alpha_1 + 1 \cdot \alpha_2$, which of course is the sum of the thresholds. The second row of the matrix is [1 -1] and thus the second new parameter (let's call it θ_{diff}) is defined as $\theta_{\text{diff}} = 1 \cdot \alpha_1 + -1 \cdot \alpha_2 = \alpha_1 - \alpha_2$, the difference between the thresholds. Thus if we specify thresholds to be parameterized using the full matrix (both rows) we can estimate θ_{sum} and θ_{diff} rather than α_1 and α_2 . If θ_{sum} and θ_{diff} are both free parameters, α_1 and α_2 are each allowed to take on any value independently of each other. For example, let's say the value for α_1 is 10 and that for α_2 is 20. This can be accomplished by setting θ_{sum} to 30 and θ_{diff} to -10. In Palamedes, the transformation will happen behind the scenes: Palamedes will estimate θ_{sum} and θ_{diff} but report the values of α_1 and α_2 .

In order to constrain thresholds to have identical values, we omit the second row from the above contrast matrix. Essentially, only the sum of the thresholds is estimated and nothing else, effectively fixing the value of θ_{diff} to 0. In other words, the thresholds can take on any value just as long as $\alpha_1 = \alpha_2$. Designing a contrast matrix that can be used to test a specific research hypothesis can be a bit of an art. Here we give some general principles and guidelines to follow that will help you get started. In general, each row of a contrast matrix specifies the weights that should be applied to each of the thresholds to define a new parameter that is some linear combination of the threshold values. We should have one weight for each condition and thus as many columns in our matrix as there are conditions. (Note that any of these weights may be equal to 0.)

1. Any contrast matrix that defines a model which allows the sum (and thus also the average) of thresholds to differ from 0 should contain a row of 1s. Some may be offended if we refer to such a row as a contrast (because it violates rule (2) it is not a contrast) but nevertheless, sometimes we will.
2. The weights (or coefficients) in each of the other rows should add to 0.
3. Rows should be uncorrelated (or "orthogonal").

BOX 9.4 (*cont'd*)

4. Any contrast matrix that adheres to (1) through (3) and has as many rows as there are conditions will allow all thresholds to take on any value independently of the other thresholds.
5. The sign of the coefficients is irrelevant, as long as rule (2) is adhered to (e.g., $[1 \ 1; \ 1 \ -1]$ would lead to an identical model fit as $[1 \ 1; \ -1 \ 1]$).
6. The exact value of any of coefficients is irrelevant as long as rule (2) is adhered to (e.g., $[2 \ 2; \ -3 \ 3]$ would lead to identical results as $[1 \ 1; \ -1 \ 1]$).

Note that even though rule (2) is implied by (1) and (3), we think it is worth spelling out explicitly.

Note also that sensible contrast matrices can be formed that do not adhere to the above rules. For example, contrast matrix $[1 \ 0; \ 0 \ 1]$ (i.e., the identity matrix) is a sensible matrix that will result in an identical fit as $[1 \ 1; \ -1 \ 1]$ does. Once again, apologies to those offended by us referring to $[1 \ 0; \ 0 \ 1]$ as a “contrast” matrix.

When there are more than two groups, it gets a bit trickier to find appropriate contrasts, but you can also make more interesting contrasts. Let's look at an experiment with three conditions. Let's say you suspect on theoretical grounds that the threshold in condition 1 differs from the (averaged) thresholds in the other two conditions but that the thresholds in conditions 2 and 3 do not differ from each other. We have three conditions, so our contrast matrix needs three columns. Both of our models should allow the average threshold to differ from 0, so our first row in the contrast matrix is easy: $[1 \ 1 \ 1]$. Testing whether the threshold in condition 1 differs from the averaged thresholds in conditions 2 and 3 comes down to testing whether the value of $\alpha_1 - (\alpha_2 + \alpha_3)/2$ equals 0. Rewriting $\alpha_1 - (\alpha_2 + \alpha_3)/2$ as $(1)(\alpha_1) + (-1/2)(\alpha_2) + (-1/2)(\alpha_3)$ makes apparent that the contrast that can be used to test this is $[1 \ -0.5 \ -0.5]$ or, by rule (6), $[2 \ -1 \ -1]$. Note that rule (2) is followed and the contrast coefficients add to 0. This will generally be true for sensible contrasts. Testing whether the threshold in condition 2 differs from the threshold in condition 3 comes down to testing whether $\alpha_1 = \alpha_2$ or, stated differently, whether $(0)(\alpha_1) + (1)(\alpha_2) + (-1)(\alpha_3) = 0$, which gives us our third contrast: $[0 \ 1 \ -1]$. Our third contrast also follows rule (2). We finally note that the three contrasts are orthogonal. Having defined three contrasts that are orthogonal across an equal number of conditions means that the full contrast matrix can fully specify any pattern of thresholds. Another way to think of this is that any set of three numbers can be described as some linear combination of the three contrasts. For example, $[10 \ 12 \ 14] = (12)([1 \ 1 \ 1]) + (-1)([2 \ -1 \ -1]) + (-1)([0 \ 1 \ -1])$. Note also that any linear combination of only the first two contrasts can produce only those sets of three thresholds for which $\alpha_2 = \alpha_3$.

In order to test whether α_1 differs from the average of α_2 and α_3 we would compare a model in which thresholds are reparameterized using the full contrast matrix to a model in which we omit the second row from the contrast matrix. In order to test whether α_2 differs from α_3 , we compare a model in which we use the full matrix to a model in which we omit the third row from the matrix.

9.3.4.1.1 EXAMPLE: TREND ANALYSIS

The use of contrast matrices to specify models provides for much flexibility with respect to the specific research questions that can be addressed. For example, let us revisit the four-condition experiment in [Section 9.2.5](#), the results of which are shown in [Figure 9.6](#). Earlier, we performed a model comparison and concluded that the assumption that the true thresholds were identical between the four conditions was untenable. This was, of course, not particularly surprising. The eyeball method leaves little doubt as to the statistical reliability of the difference between the threshold in the first condition and any of the other three thresholds. However, we may wish to answer more specific questions. For example, whereas it is quite clear that thresholds generally increase with adaptation duration, there also appears to be a decelerating trend. That is, the increase of threshold values appears to level off as the adaptation duration increases. The question arises whether this effect is “real” or might be attributed entirely to sampling error.

The model comparison to be performed would involve a lesser model that does not allow thresholds to deviate from a straight line (i.e., constrains the relationship between the threshold value and the adaptation duration to be linear). This lesser model would be compared against a fuller model that does allow thresholds to deviate from a straight line. Readers that are well-versed in ANOVA or the general linear model will have realized by now that polynomial contrasts will allow us to formulate our models. Let us explain the logic by way of the current example. The coefficients that correspond to polynomial contrasts are listed in many introductory level statistics texts. Palamedes has a routine `PAL_Contrasts` that generates polynomial contrast matrices.

The reparameterization matrix that uses polynomial contrasts for our four-condition example is

$$M = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -3 & -1 & 1 & 3 \\ 1 & -1 & -1 & 1 \\ -1 & 3 & -3 & 1 \end{bmatrix}$$

Hence, the θ and α parameters are related thus

$$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -3 & -1 & 1 & 3 \\ 1 & -1 & -1 & 1 \\ -1 & 3 & -3 & 1 \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix}$$

Consider [Figure 9.9](#). In panel A, all four θ parameters are fixed at 0. This will, in turn, fix all thresholds at 0. If we allow θ_1 to vary (but keep the other θ s fixed at 0), the thresholds are allowed to differ from 0 but are not allowed to differ from each other. In other words, the modeled thresholds will fall along a zero degree polynomial. Graphically, this would mean that all thresholds are constrained to fall on a single horizontal line (panel B). If we also allow θ_2 to vary, the thresholds are still constrained to fall on a straight line, but this line is now allowed to have a slope unequal to 0 (i.e., it is a first-degree polynomial). In panel D, θ_3 is free to vary, and threshold estimates are now allowed to follow any second-degree

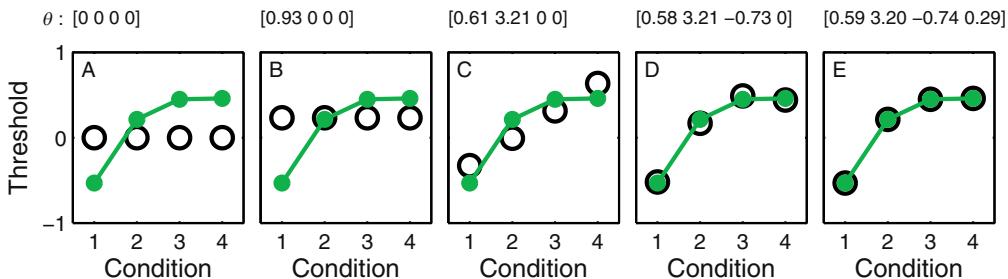


FIGURE 9.9 The results of the four-condition experiment of Section 9.2.5 (green) along with best-fitting models of varying restrictiveness (open circles).

polynomial. Finally, in panel E all four θ parameters are free to vary, allowing each threshold to take on any value independent of the others.

The likelihood ratio test may be used to compare any of the models in Figure 9.9 to any other model in the figure. For any model, the parameter space is a subset of the parameter space of any model that is to its right. For example, the model comparison we performed in Section 9.2.5 of this chapter compared model B to model E. Based on that comparison we concluded that model B was untenable. However, the question we are currently attempting to answer requires a different comparison. The question is whether the decelerating trend in the thresholds apparent in the data is real or may have arisen due to sampling error alone. In order to answer this question we should compare model C, which assumes thresholds follow a first-degree polynomial, to model D, which allows thresholds to follow a second-degree polynomial that accommodates the deceleration seen in the data. We again constrain the values of the slope to be equal across conditions, and we do the same for the lapse rates. In Box 9.5, we demonstrate how to perform this model comparison in Palamedes. In model C,

BOX 9.5

LINEAR REPARAMETERIZATIONS USING CONTRAST MATRICES IN PALAMEDES

This box will demonstrate how to perform the fits and model comparison discussed in Section 9.3.4.1.1. This is the model comparison between models C and D in Figure 9.9. First we create a contrast matrix containing polynomial contrasts using the Palamedes function `PAL_Contrasts`. In order to do so we type

```
>>Contrasts=PAL_Contrasts(4, 'polynomial');
```

The first argument (4) specifies the number of conditions in the experiment and the second argument ('polynomial') specifies that we wish to generate polynomial contrasts.

```
>>Contrasts
Contrasts =
  1   1   1   1
 -3  -1   1   3
  1  -1  -1   1
 -1   3  -3   1
```

BOX 9.5 (*cont'd*)

In the function `PAL_PFLR_ModelComparison` we pass the appropriate contrast matrices instead of options 'unconstrained', 'fixed', etc. that we used in [Box 9.1](#). Even though it is not necessary in order to perform the model comparison, let us first fit the two models using `PAL_PFML_FitMultiple`. First we set up the necessary arrays.

```
>>StimLevels = [-2:1:2; -2:1:2; -2:1:2; -2:1:2];
>>OutOfNum = [150 150 150 150 150; 150 150 150 150 150; ...
    150 150 150 150 150; 150 150 150 150 150];
>>NumPos = [84 92 128 137 143; 67 85 103 131 139; 73 85 ...
    92 125 143; 82 86 97 122 141];
>>PF = @PAL_Logistic;
>>params = [-.6 1.8 .5 .02; .1 1.8 .5 .02; .6 1.8 .5 .02; ...
    .9 1.8 .5 .02]; %guesses
```

For each of the two models, we need to create a contrast matrix that defines the relationships among thresholds in the model. Let us start with model C. The matrix `Contrasts` created above corresponds to model E and would be equivalent to allowing thresholds to take on any value independently of the other thresholds. In order to constrain the parameters as in model C, we need to limit the contrast matrix to the first two rows, which allow the mean threshold to differ from 0, and the thresholds to increase in a linear fashion with condition, respectively.

```
>>ContrastsModelC = Contrasts(1:2,:);
```

Note that:

$$\begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ -3 & -1 & 1 & 3 \end{bmatrix} * \begin{bmatrix} \alpha_2 \\ \alpha_2 \\ \alpha_2 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4 \\ -3\alpha_1 - 1\alpha_2 + 1\alpha_3 + 3\alpha_4 \end{bmatrix}$$

We are now ready to call `PAL_PFML_FitMultiple`:

```
>>paramsC=PAL_PFML_FitMultiple(StimLevels, NumPos, ...
    OutOfNum, params, PF, 'Thresholds', ContrastsModelC, ...
    'Slopes', 'constrained', 'GuessRates', 'fixed', 'LapseRates',...
    'constrained');
```

Note that for the slopes, guess rates, and lapse rates we simply use the verbal labels 'constrained' and 'fixed' as in [Box 9.1](#). When we inspect `paramsC`, we note that the thresholds indeed increase linearly with condition:

```
>>paramsC
paramsC =
-0.3283 1.7138 0.5000 0.0397
-0.0072 1.7138 0.5000 0.0397
0.3140 1.7138 0.5000 0.0397
0.6351 1.7138 0.5000 0.0397
```

BOX 9.5 (cont'd)

Note that we can inspect the values of the θ parameters using the transformation given in Section 9.3.4.1.1. We will use the full matrix `Contrasts` in order to show explicitly that θ_3 and θ_4 are equal to 0:

```
>>thetas = Contrasts*paramsC(:,1)
thetas =
    0.6136
    3.2114
    0
   -0.0000
```

In order to define model D, we set up `ContrastsModelD` to contain the first three rows of the full contrast matrix:

```
>>ContrastsModelD=Contrasts(1:3,:);
```

And call `PAL_PFML_FitMultiple` using `ContrastsModelD` instead of `ContrastsModelC`.

```
>>paramsD = PAL_PFML_FitMultiple(StimLevels, NumPos, ...
    OutOfNum, params, PF, 'Thresholds', ContrastsModelD, ...
    'Slopes', 'constrained', 'GuessRates', 'fixed', ...
    'LapseRates', 'constrained');
```

A quick check confirms that θ_3 is now indeed no longer equal to 0, while θ_4 is

```
>>thetas = Contrasts*paramsD(:,1)
thetas =
    0.5762
    3.2064
   -0.7335
    0.0000
```

Let us now perform the hypothesis test. Remember that we only need to specify the deviations from the default settings, which are listed in Box 9.1.

```
>>B = 4000;
>>[TLR pTLR paramsC paramsD TLRSim converged] = ...
    PAL_PFLR_ModelComparison(StimLevels, NumPos, OutOfNum, ...
    paramsD, B, PF, 'lesserthreshold', ContrastsModelC, ...
    'fullerthreshold', ContrastsModelD, 'lesserlapse', ...
    'constrained', 'fullerlapse', 'constrained', ...
    'fullerslope', 'constrained');
```

(We may again need to use the optional arguments `lapseLimits`, `maxTries`, and `rangeTries` in order to avoid failed model fits. See `PAL_PFLR_FourGroup_Demo.m` in `PalamedesDemos` folder). On completion of the procedure, we may inspect `pTLR`:

```
>>pTLR
pTLR =
    0.0065
```

θ_1 and θ_2 are estimated to equal 0.61 and 3.21, respectively, while θ_3 and θ_4 are fixed at a value of 0. Model D allows θ_3 to take on any value, and the estimates for θ_1 , θ_2 , and θ_3 are 0.58, 3.21, and -0.73 respectively, while θ_4 remains fixed at 0. The TLR for the model comparison equals 7.3014. Based on 4000 simulations, the p -value associated with the model comparison equals 0.0065 (i.e., 26 of the 4000 simulations resulted in greater TLR values). The “asymptotic” p -value (i.e., based on the χ^2 -distribution with one degree of freedom) equals 0.0069 and corresponds closely to that obtained using simulations.

Thus, the results obtained from our human observer are not likely to arise from an observer who acts strictly according to model C, and we conclude that the observed decelerating trend in thresholds is real. We leave it up to the reader to confirm that adding the fourth row in the contrast matrix (allowing all thresholds to take on any value independent of each other) does not lead to a fit that is significantly better than that of model D.

It appears that model D suffices to model the data. However, remember that the comparison between model C and model D does not in any way validate the assumptions that both models make. These are the assumptions of stability, independence of observations, the assumption that the form of the PF is a logistic function, that the slopes are equal between conditions, that the guessing rate equals 0.5, etc. We can check all but the assumptions of stability and independence by performing a goodness-of-fit test, i.e., by comparing model D against the saturated model, which makes only the assumptions of stability and independence. Using 4000 simulations we found the goodness-of-fit p -value for Model D to equal 0.7670 (i.e., 3068 of the 4000 simulations resulted in greater Deviance values, indicating a worse fit than the Deviance of the model fitted to the human observer’s data). The p -value derived from the asymptotic χ^2 distribution with 15 degrees of freedom equals 0.7691 and is very close to the p -value based on simulations. We conclude that model D provides a good fit to the data.

9.3.4.1.2 EXAMPLE: PAIRWISE COMPARISONS

Imagine that, for whatever reason, it is of interest to determine whether the difference between the thresholds in conditions 3 and 4 is real or could be explained by sampling error alone. None of the comparisons performed above answers this question. Using the eyeball method with reference to Figure 9.7, few would suspect that this difference is real. However, for the sake of demonstration let us perform the comparison formally. Our fuller model should allow all thresholds (including those in conditions 3 and 4) to take on any value independent of each other; the lesser should be identical except that thresholds 3 and 4 should be constrained to equal each other. So-called Helmert contrasts allow us to define these models. A reparameterization matrix that uses Helmert contrasts to recode the four threshold parameters is given by

$$M = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 3 & -1 & -1 & -1 \\ 0 & 2 & -1 & -1 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

as before

$$\boldsymbol{\theta} = \mathbf{M} * \boldsymbol{\alpha}$$

The first row of \mathbf{M} allows the average threshold to deviate from 0. The second row allows threshold 1 to differ from the average of thresholds 2, 3, and 4. The third row allows threshold 2 to differ from the average of thresholds 3 and 4. Finally, the fourth row allows threshold 3 to differ from threshold 4. In conjunction, these contrasts allow all thresholds to take on any value (there are four orthogonal contrasts reparameterizing an equal number of parameters), and if we use the full matrix to define the model, we will get an identical fit as compared to using, for example, a full polynomial set of contrasts. Let us now define the lesser model. It is θ_4 , defined by the fourth row of the contrast matrix, which allows thresholds 3 and 4 to differ, and this is the parameter we need to fix at a value of 0 to define our lesser model, while in the fuller model we allow it to take on any value. A fit of the lesser model results in the following threshold estimates for the four conditions: -0.5317, 0.2142, 0.4531, and 0.4531. Note that thresholds in conditions 3 and 4 are indeed equal, as specified by the lesser model.

When the fuller and lesser models are compared using the likelihood ratio test, we find that the p -value based on the χ^2 distribution with 1 df (the models differ only with respect to whether θ_4 is fixed or free to vary) equals 0.9570. When we used 4000 simulations to derive a p -value we found it to equal 0.9530.

Different research questions require different model comparisons. Contrasts may be used to define a variety of models. Trend analysis and pairwise comparisons are just two examples. For example, contrasts may also be used to test for the marginal effects of two or more variables as well as their interaction in a factorial design (e.g., [Prins, 2008](#)). The use of contrasts to define models is routine in the context of the General Linear Model and an excellent introduction is given in [Judd et al. \(2008\)](#).

9.3.4.2 Nonlinear Reparameterizations

Contrasts can be used to define linear transformations of parameters only. Here we discuss, by example, how to address a research question that requires a nonlinear reparameterization. In order to investigate the mechanism by which perceptual learning of texture segregation occurs, [Prins and Streeter \(2009\)](#) tested whether perceptual learning of texture segregation transfers across retinal location and first-order characteristics of the textures. We present data from one observer (MH) who was first trained in one stimulus condition for 13 sessions of 500 trials each. After this initial training MH performed two sessions of 500 trials in which the stimuli were presented at a different retinal location from that trained, then two sessions of 500 trials each at the trained retinal location but using stimuli that had first-order characteristics different from those trained. Finally, MH performed her last two sessions under the same condition as her 13 initial training sessions.

Threshold estimates obtained by fitting the 19 sessions individually are shown in [Figure 9.10](#). These thresholds were obtained by fitting a Gumbel function to the log-transformed stimulus intensities, with the threshold and slope parameters free to vary but the guess and lapse rate fixed at 0.5 and 0.0495, respectively (we will shortly motivate our choice for the specific lapse rate value). Also shown are standard errors of the threshold estimates that were derived by parametric bootstrap.

There is clearly a learning effect for the “Trained” stimuli. By eyeballing thresholds and their standard errors ([Section 9.2.1](#)) it seems safe to conclude that learning does not transfer fully to a different retinal location (square symbols). The results regarding a change in first-order characteristics of the stimuli (triangular symbols) are not so clear. We will here devise

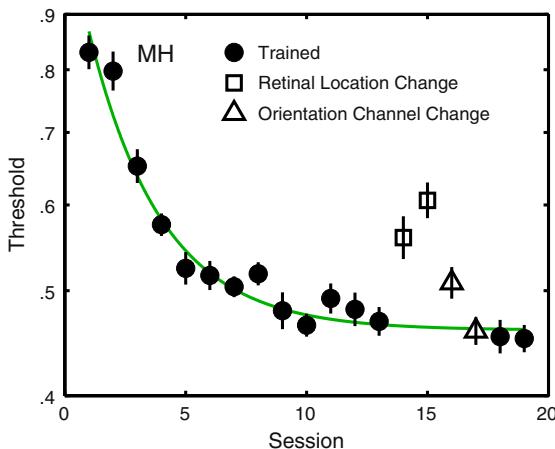


FIGURE 9.10 Thresholds obtained in a perceptual-learning study.

a test that can be used whether learning has transferred to the stimuli with different first-order characteristics. We do so by defining two models, one in which the triangular symbols are constrained to follow the same learning curve as the “Trained” sessions and another in which they are free to deviate from this curve. Let us first define a model F (for “Fuller”) in which sessions one through 13 as well as 18 and 19 (i.e., those sessions that use the trained stimuli) follow an exponential decay function (a function often used to model learning curves) but in which all four sessions containing stimuli different from those trained are allowed to deviate from the curve. We reparameterize thresholds as

$$\begin{aligned}
 \alpha(\text{session}) &= \theta_1 + \theta_2 \times e^{-\theta_3 \times (\text{session}-1)} && \text{for session } \in \{1, \dots, 13, 18, 19\} \\
 \alpha(\text{session}) &= \theta_1 + \theta_2 \times e^{-\theta_3 \times (\text{session}-1)} + \theta_4 && \text{for session } = 14 \\
 \alpha(\text{session}) &= \theta_1 + \theta_2 \times e^{-\theta_3 \times (\text{session}-1)} + \theta_5 && \text{for session } = 15 \\
 \alpha(\text{session}) &= \theta_1 + \theta_2 \times e^{-\theta_3 \times (\text{session}-1)} + \theta_6 && \text{for session } = 16 \\
 \alpha(\text{session}) &= \theta_1 + \theta_2 \times e^{-\theta_3 \times (\text{session}-1)} + \theta_7 && \text{for session } = 17
 \end{aligned} \tag{9.3}$$

The three parameters determining the characteristics of the learning curve are θ_1 , θ_2 , and θ_3 . Parameter θ_1 determines the lower asymptote of the curve, θ_2 determines the difference between the (modeled) threshold in session one and the asymptote, and θ_3 determines the decay rate (i.e., rate of learning). Our parameterization of the thresholds for sessions 14 through 17 may appear awkward at first glance in that we could have simply parameterized these thresholds by assigning each their own θ parameter without involving the learning curve in their definition. However, note that in the above parameterization, θ_4 through θ_7 effectively correspond to deviations from the learning curve. This will allow us to define our “lesser” model (in which thresholds for sessions 16 and 17 are not allowed to deviate from the learning curve) by fixing θ_6 and θ_7 at 0. Moreover, the above strategy (specifically, defining a “lesser” model by fixing the value of a parameter (or parameters) that is free in the “fuller” model) will ensure that the lesser model is nested under the fuller model.

Both the fuller and the lesser model assumed that the slope parameter was equal across all sessions, that the guess rate was 0.5 in all sessions, and that the lapse rate was equal across all

sessions. Thus, the fuller model had nine free parameters: the seven θ s that define thresholds, one shared slope, and one shared lapse rate (we used the estimate of the lapse rate under this model [0.0495] as the fixed value in the individual fits shown in the figure). The lesser model is identical to the fuller except that it fixes θ_6 and θ_7 at 0 and thus has only seven free parameters. The green curve in Figure 9.9 is the learning curve defined by the estimates of θ_1 , θ_2 , and θ_3 derived under the fuller model (note that under the lesser model the thresholds in sessions 16 and 17 are constrained to follow the learning curve and this constraint will affect the estimates of θ_1 , θ_2 , and θ_3).

The model comparison between the fuller and lesser model resulted in a p -value equal to 0.0292 when based on Monte Carlo simulations ($B = 10,000$ simulations) and a p -value equal to 0.0267 when based on the theoretical χ^2 distribution with two parameters (the difference between the number of free parameters in the fuller and lesser models). Thus, the fuller model provides a significantly better fit as compared to the lesser model. A goodness-of-fit test for the fuller model indicated that it fit the data well (the p -value based on 10,000 Monte Carlo simulations equaled 0.1161). We conclude that for observer MH, learning did not fully transfer to stimuli in which the information was contained in an orientation channel different than that used during training. Box 9.6 discusses how to use nonlinear reparameterizations in Palamedes in order to fit the above models and perform the model comparison.

BOX 9.6

NONLINEAR REPARAMETERIZATIONS IN PALAMEDES

Palamedes allows users to specify their own function that reparameterizes any of the four parameters of the PF. We will demonstrate how to use this functionality by performing the analysis of observer MH's data discussed in Section 9.3.4.2. MH's data are needed to run the code in this example. They are stored in `MH_data.mat` in the `PalamedesDemos` folder. In the fuller model the thresholds in session 1 through 13 and sessions 18 and 19 are constrained to follow an exponential decay function, which is specified by three parameters, and the thresholds in sessions 14 through 17 are allowed to take on any value (Eqn (9.3)).

In order to fit our model using Palamedes, we must write a MATLAB function that implements the above parameterization. The function should accept as input a vector containing values for the 7 θ s and return a vector with the 19 corresponding α s:

```
function alphas = ParameterizeThresholds(thetas)

session = 1:19;
alphas(1:19) = thetas(1)+thetas(2)*exp(-thetas(3)*(session-1));
alphas(14:17) = alphas(14:17)+thetas(4:7); %will overwrite alphas(14:17)
```

In order to instruct `PAL_PFML_FitMultiple` to utilize the above parameterization, we pass it a structure that contains: a function handle for the parameterization function, our guesses for the initial values of the thetas and a vector that indicates which of the thetas is free to vary. The function `PAL_PFML_setupParameterizationStruct` sets up the structure:

```
>>funcParamsF = PAL_PFML_setupParameterizationStruct
```

BOX 9.6 (*cont'd*)

```
>>funcParamsF =
    funcA: []
    paramsValuesA: []
    paramsFreeA: []
    funcB: []
    paramsValuesB: []
    paramsFreeB: []
    funcG: []
    paramsValuesG: []
    paramsFreeG: []
    funcL: []
    paramsValuesL: []
    paramsFreeL: []
```

For each of the four parameters of the PF (the letters A, B, G, and L in the above stand for alpha, beta, gamma, and lambda, respectively), the structure has three fields. Let us explain by implementing the above parameterization for alphas. The field `.funcA` must contain a handle to the function, that parameterizes α s

```
>>funcParamsF.funcA = @ParameterizeThresholds;
```

We indicate our initial guesses for the seven θ s in the field `.paramsValuesA`:

```
>>funcParamsF.paramsValuesA = [-.5 .35 .2 .1 .1 .1 .1];
```

The field `.paramsFreeA` allows us to specify which of the θ s are free to vary (1) and which are fixed (0). In our fuller model, we will make all θ s free to vary:

```
>>funcParamsF.paramsFreeA = [1 1 1 1 1 1 1];
```

We will constrain the β s to be equal across all sessions, as we did in our previous example. Let us here also constrain the β s to have positive values by using the following reparameterization, also implemented in a MATLAB function:

```
function betas = ParameterizeSlopes(rho)
    betas (1:19) = exp(rho(1));
```

Note that this reparameterization assigns the single parameter ρ (ρ) to all 19 values for β , in effect constraining the slopes to be equal across all 19 sessions.

We change our `funcParamsF` structure accordingly:

```
>>funcParamsF.funcB = @ParameterizeSlopes;
>>funcParamsF.paramsValuesB = [log(3)];
>>funcParamsF.paramsFreeB = [1];
```

BOX 9.6 (*cont'd*)

We will use the Gumbel function, fix the guess rate at a value of 0.5, and we will estimate a single lapse rate across all sessions and constrain it to have a value in the interval [0 1]. Thus, our call to `PAL_PFML_FitMultiple` is:

```
>>PF=@PAL_Gumbel;
>>[paramsFFitted LLF exitflagF outputF funcParamsFFitted numParamsF] = ...
    PAL_PFML_FitMultiple(StimLevels, NumPos, OutOfNum, ...
    [0 0 .5 .03], PF, 'thresholds', funcParamsF, ...
    'slopes', funcParamsF, 'guessrates', 'fixed', 'lapses', ...
    'constrained', 'lapselimits', [0 1]);
```

Note that it is possible to mix the manners in which we define our parameterizations: 'thresholds' and 'slopes' are defined by the structure defined above, while 'guessrates' and 'lapses' are defined by verbal labels (as discussed in [Box 9.1](#)). Because we define the guess rates and lapse rates using verbal labels we may leave the entries for these parameters in the `funcParamsF` structure empty. Note also that the vector containing parameter values ([0 0 .5 .03]) contains the fixed value we wish to use for gamma and our initial guess for lambda to be used but that the entries for alpha and beta will be ignored as these are reparameterized. The initial values for the reparameterized thetas are passed to the function in the reparameterization structure `funcParamsF`.

After the function completes, `paramsFFitted` will contain estimated PF parameters, where all values adhere to the constraints specified by our model:

```
>>paramsFFitted
ans =

```

-0.0616	5.5895	0.5000	0.0495
-0.1393	5.5895	0.5000	0.0495
-0.1951	5.5895	0.5000	0.0495
-0.2352	5.5895	0.5000	0.0495
-0.2640	5.5895	0.5000	0.0495
-0.2847	5.5895	0.5000	0.0495
-0.2996	5.5895	0.5000	0.0495
-0.3102	5.5895	0.5000	0.0495
-0.3179	5.5895	0.5000	0.0495
-0.3234	5.5895	0.5000	0.0495
-0.3274	5.5895	0.5000	0.0495
-0.3302	5.5895	0.5000	0.0495
-0.3323	5.5895	0.5000	0.0495
-0.2551	5.5895	0.5000	0.0495
-0.2193	5.5895	0.5000	0.0495
-0.2912	5.5895	0.5000	0.0495
-0.3399	5.5895	0.5000	0.0495
-0.3365	5.5895	0.5000	0.0495
-0.3368	5.5895	0.5000	0.0495

BOX 9.6 (cont'd)

Note that the value for the lapse rate estimate obtained here is the value we used (for this reason) in the individual fits that gave us the thresholds shown in [Figure 9.10](#). The estimates for our theta parameters are given in `funcParamsFFitted`, which is a structure with fields identical to the structure `funcParamsF`:

```
>> funcParamsFFitted.paramsValuesA  
ans =  
-0.3375 0.2759 0.3308 0.0787 0.1155 0.0443 -0.0038  
>> funcParamsFFitted.paramsValuesB  
ans =  
1.7209
```

Note that this value is the estimate of ρ and that $\beta = \exp(\rho)$, such that our estimate for $\beta = \exp(1.7209) = 5.5896$. The green curve shown in [Figure 9.10](#) is the learning curve defined by the estimates of θ_1 , θ_2 , and θ_3 under this model. Our “lesser” model L is identical to Model F except that θ_6 and θ_7 are fixed at the value of 0.

```
>>funcParamsL = funcParamsF;  
>>funcParamsL.paramsValuesA(6:7) = 0; %specify values at which to fix  
%theta(6) and theta(7)  
>>funcParamsL.paramsFreeA(6:7) = 0; %specify theta(6) and theta(7) to be  
%fixed parameters.  
>>[paramsLFitted LLL exitflagL outputL funcParamsLFitted numParamsL] = ...  
PAL_PFML_FitMultiple(StimLevels, NumPos, OutOfNum,...  
[0 0 .5 .03], PF,'thresholds',funcParamsL,'slopes',...  
funcParamsL, 'guessrates','fixed','lapsrates',...  
'constrained','lapselimits',[0 1]);  
  
>>paramsLFitted  
ans =  
  
-0.0627 5.8441 0.5000 0.0537  
-0.1420 5.8441 0.5000 0.0537  
-0.1983 5.8441 0.5000 0.0537  
-0.2382 5.8441 0.5000 0.0537  
-0.2665 5.8441 0.5000 0.0537  
-0.2866 5.8441 0.5000 0.0537  
-0.3009 5.8441 0.5000 0.0537  
-0.3110 5.8441 0.5000 0.0537  
-0.3182 5.8441 0.5000 0.0537  
-0.3233 5.8441 0.5000 0.0537  
-0.3269 5.8441 0.5000 0.0537  
-0.3295 5.8441 0.5000 0.0537  
-0.3313 5.8441 0.5000 0.0537  
-0.2600 5.8441 0.5000 0.0537  
-0.2203 5.8441 0.5000 0.0537  
-0.3342 5.8441 0.5000 0.0537  
-0.3346 5.8441 0.5000 0.0537  
-0.3350 5.8441 0.5000 0.0537  
-0.3352 5.8441 0.5000 0.0537
```

BOX 9.6 (*cont'd*)

```
>> funcParamsLFitted.paramsValuesA
ans =
-0.3358 0.2730 0.3430 0.0726 0.1132 0 0

>> funcParamsLFitted.paramsValuesB
ans =
1.7654
```

In order to perform the model comparison we can call the function `PAL_PFLR_ModelComparison` in a manner similar to that used in [Boxes 9.1 and 9.5](#):

```
>>[TLR pTLR paramsL TLRSim converged funcParamsL funcParamsF] = ...
PAL_PFLR_ModelComparison(StimLevels, NumPos, OutOfNum, paramsLFitted, ...
400,PF,'lesserthresholds',funcParamsL,'lesserslopes',funcParamsL, ...
'lesserguessrates','fixed','lesserlapserates','constrained', ...
'fullerthresholds',funcParamsF,'fullerslopes',funcParamsF, ...
'fullerguessrates','fixed','fullerlapserates','constrained', ...
'lapseslimits',[0 1]);
```

The script `PAL_PFLR_LearningCurve_Demo` in the `PalamedesDemos` folder performs the analysis described here.

9.3.5 A Note on Failed Fits

On occasion, not all fits to simulated experiments converge on a solution. We discussed the basic issue in some detail in Box 4.7 in Chapter 4. Generally speaking, the more free parameters a model has, the more likely it is that a fit will fail to converge; in particular, models that are defined across multiple conditions tend to have large numbers of parameters. For example, Model E in [Figure 9.9](#) has six free parameters (four thresholds, one shared slope and one shared lapse rate). All Palamedes functions that perform simulations will issue a warning when a fit fails. Most of the suggestions we gave in Box 4.7 will apply here. However, the use of a brute-force search through a user-defined parameter grid that precedes the iterative Simplex search is not implemented in Palamedes functions that fit more than 1 PF simultaneously. It should be kept in mind that some datasets may never be fit successfully, simply because the likelihood function may not have a global maximum. We have a few options when the fit to the data is successful but not all the simulated datasets that are used to determine the standard error of parameter values or *p*-values for model comparisons converged.

We encountered the problem in Section 4.3.3.1.3 and will briefly reiterate what we discussed there. The tempting solutions (ignoring the failed fits and calculating the standard error or p -value across the successful fits only, replacing the simulations that could not be fit with new simulations, or retrying the entire set of B simulations until we have a set of B simulations that were all successfully fit) are all inappropriate, since our standard errors or p -values would then be based on a nonrandom sample of possible simulations. Instead, we should try to make all fits successful. Generally, the convergence of fits will improve with a decrease in the number of free parameters in the model(s) and with an increase in the number of observations, as discussed in Box 4.7.

If all but a very small percentage of simulations converged successfully, we might ignore the failed fits in the calculation of standard errors or p -values *as long as we report to our audience that our numbers are based on an incomplete set of simulations*. Our audience should then make up their own minds as to the value they wish to place on our results. When we are running simulations in order to estimate a p -value, we could count any unsuccessful fits as evidence contradicting the argument we wish to make. For example, assume you wish to show that adaptation affects a detection threshold. You specify a lesser model that constrains the thresholds to be equal in the adaptation and no-adaptation conditions and a fuller model that allows thresholds to differ between the conditions. You then perform a model comparison by way of the likelihood ratio test to derive a p -value using $B = 4000$. Imagine that 103 of the simulated TLRs were larger than the TLR obtained from the experimental data, 3882 were smaller, and the remaining 15 simulations failed to result in a successful fit. Since you are trying to show that the lesser model is inappropriate, you wish to obtain a small p -value. You could make the argument that even if these 15 failed fits would all lead to TLRs greater than that obtained from the experimental data your p -value would still be small enough ($[103 + 15]/4000 = 0.0295$) to reject the lesser model. Once again, you would have to report that not all fits to the simulated data succeeded, and you would have to report how you derived your p -value. Finally, if our purpose is to derive a p -value from a TLR, we might compare our TLR against the theoretical χ^2 distribution with the appropriate number of degrees of freedom. This does not require any simulations.

9.3.6 Some Cautionary Words Regarding the Interpretation of p -Values

The likelihood ratio test is an example of what are sometimes termed “frequentist,” “Fisherian,” or “Null Hypothesis” (NH) tests. Other common examples of such tests that the reader might be more familiar with are the t -test and ANOVA. All NH tests have in common that the decision criterion in the model comparison is the conditional probability commonly referred to as the test’s p -value. We have discussed the p -value in great detail above in the context of the likelihood ratio test. The general meaning of a p -value is identical for all NH tests, though. Loosely speaking, a p -value is the probability that an experiment would produce the result that our experiment produced if the NH were true. The NH states the assumption in the model that is tested. It has the name “null hypothesis” because often the assumption can be stated as the value of some parameter having a value of 0. For example, the NH “adaptation does not affect the detectability of a stimulus” can be stated as “the parameter that corresponds to the difference between the threshold values in the

adaptation and the no-adaptation conditions has a value of 0". In introductory statistics courses we are taught that if the p -value is smaller than some criterion value (in the social sciences typically 0.05 or 5%), we should reject the null hypothesis.

The logic behind the NH test seems very straightforward and we noted before (Section 4.3.3.2.1 in Chapter 4) that it has a tremendous intuitive appeal. Unfortunately, there are many issues that complicate the use and interpretation of p -values. Here, we list some of these issues with a brief explanation. In [Box 9.7](#) we illustrate, by way of examples, some of these issues in the context of goodness-of-fit tests. In our experience, it is in the context of goodness-of-fit tests that researchers are often most confused by regarding the proper interpretation of a p -value.

So, what may be and what may not be concluded when you obtain a p -value that is low? The easy part of this question is what may be concluded. Ronald Fisher, who popularized the use of the p -value, explained that if the p -value that is obtained has a low value, either the NH is false or an unlikely event has occurred. And that is what may be concluded. Tempting as it may seem, what may not be concluded based on a p -value being low is that the NH is likely false. People that do make this conclusion commit what is sometimes referred to as the conversion error. This involves taking a statement such as "if P, then probably not Q" to mean also "if Q, then probably not P." In other words, the conversion error leads people to believe that a p -value is a statement regarding the probability that the NH is true given that the experiment produced the result it did. We discussed this error in Section 4.3.3.2.1 by way of an example. A researcher who believes, after rejecting the NH, that his or her p -value equals the probability that he or she has committed a type-I error (i.e., rejected the NH while it is true) has also committed the conversion error.

Another common error committed in NH testing is to decide which model comparison to perform only after inspecting the results of the experiment. A simple example would be deciding to test whether the threshold in condition C of an experiment with, say, 10 conditions differs significantly from the threshold in condition F because you note that the difference between thresholds C and F is greater than the difference between any other pair of thresholds. The error here is essentially the same as that committed by the twin researcher in [Section 9.3.2](#), which is to decide to calculate the probability of a seemingly unlikely event only after it has already happened. This error is sometimes referred to as the Texas Sharpshooter Fallacy. Imagine a person pointing a gun at a barn that stands a few hundred yards in the distance and firing a shot at it. He then walks over to the barn, locates the bullet hole, paints a bull's eye around it, and proudly exclaims that his shot was right in the center the bull's eye. Just as one would only be impressed with hitting the center of a bull's eye if it was there before the shot was taken, one should only be impressed with a low p -value if it was decided beforehand where to look for it.

A similar error is to plan many NH tests beforehand but apply to each the same p -value criterion that one would apply to a test if it were the only test one performed. In the ten-condition experiment of the previous paragraph one can form 45 pairs of conditions and for each test whether the thresholds in the pair differ significantly. Even if all NHs were true, one can expect to reject a few of them if one uses the 0.05 criterion for each. In terms of the Texan sharpshooter, one would not be impressed if the shooter paints many bull's eyes on the side of the barn, takes a shot in the general direction of the barn, finds the bull's eye that he happened to have hit, and exclaims, "did it again!"

BOX 9.7

INTERPRETING A GOODNESS-OF-FIT p -VALUE

A lot of confusion exists about the interpretation of the goodness-of-fit of a model. Often, researchers are confused because even though a model appears to fit the data very well based on a visual inspection of the model relative to the data, a goodness-of-fit test may indicate an unacceptably poor fit. We will compare two experiments in order to illustrate some of these issues. Consider first the results of Experiment "A," which are shown in [Figure B9.7.1\(a\)](#). The number of trials used at each of the five stimulus values was 100. The model that was fitted to the data is shown in green. The model assumed that the upper asymptote equals 0.98 and the lower asymptote equals 0.02 and that the shape of the PF was the Logistic. The threshold and slope value were estimated from the data using a maximum likelihood criterion. From a visual inspection of the model it appears to describe the data very well. For the sake of comparison we also show the outcome of experiment "B," which was similar to Experiment A but used only 10 trials for each of the five stimulus values. The data for this experiment and the model that was fitted to the data are shown in [Figure B9.7.1\(b\)](#). These data look much messier and do not appear to correspond very well to the fitted model.

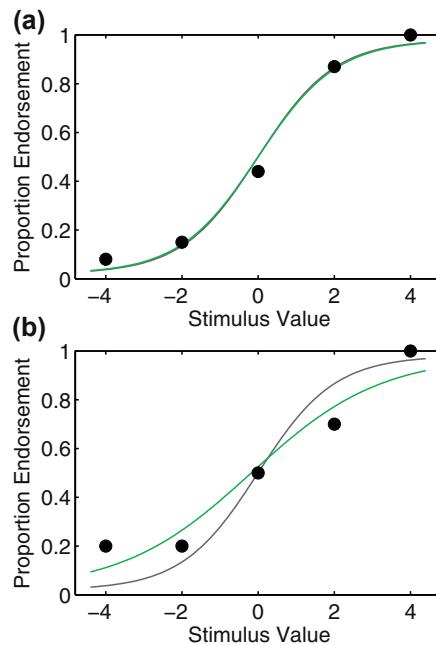


FIGURE B9.7.1 Hypothetical experimental results. In (a), each data point is based on 100 trials, and in (b) each data point is based on 10 trials. The functions that correspond to the best-fitting Logistic are shown in green and the generating functions are shown in gray.

BOX 9.7 (*cont'd*)

The gray curve shown in both plots is the function that actually generated the data. Clearly, the nice-looking fit based on 100 trials per condition does a much better job of estimating the threshold and slope of the generating function as compared to the fit based on 10 trials per condition. This is not surprising, of course. What many people would find surprising, however, is that a goodness-of-fit test of the fit for Experiment A indicates that it is a very poor fit and that the fitted model should actually be rejected. The *p*-value based on a large number of Monte Carlo simulations was 0.0050, well below the commonly utilized cut-off value of 0.05. At the same time, and perhaps equally surprising, is that a goodness-of-fit test of the fit for Experiment B indicates that this is a good fit (*p* = 0.4540).

Based on the examples above, it is clear that the *p*-value based on a goodness-of-fit test does not measure how close the modeled probabilities of positive responses are to the observed proportions of positive responses (although this certainly plays a role). It is also clear from these examples that goodness-of-fit does not measure how closely a model matches the mechanism that generated the data. What then does it measure? A goodness-of-fit test is simply a statistical model comparison using the Fisherian approach of the NH test. The logic behind the Fisherian approach is presented in detail elsewhere in this chapter. Here we provide only a brief description. One of the two models to be compared in a goodness-of-fit test is the so-called saturated model (this is what makes it a goodness-of-fit test). The saturated model does not make any assumptions about how the probabilities of a positive response relate to stimulus intensity and simply estimates a probability of a correct response for each of the five stimulus intensities that were used. The other model in the comparison makes more restrictive assumptions. The models fitted above assume that the function that relates the probability of a correct response to the stimulus intensity is a Logistic function. The models further assume the specific values for the upper and lower asymptotes listed above. The models require the estimation of only two parameters (threshold and slope of the PF). In general, a model comparison tests whether there is enough evidence to suggest that the additional assumptions that the lesser model makes are false. Simplifying a bit, the classical Fisherian approach to this issue is to determine the probability that the differences between the data and the model would occur if the assumptions that the model makes are correct. That is, it determines whether these differences would be likely to occur by sampling error alone. This probability is the *p*-value associated with a model comparison, including goodness-of-fit tests. If this probability is low, we conclude that the differences are not due to a sampling error alone but rather should be taken to suggest that the model is incorrect.

Loosely speaking then, the poor goodness-of-fit for the fit to the data of Experiment A is to be interpreted as indicating that a mechanism that acts according to the model would have been unlikely to generate the observed data. In Experiment B the discrepancies between the modeled probabilities of positive responses and the actually observed proportions of positive responses were much greater than they were in Experiment A. However, the goodness-of-fit test takes into account that the observed proportions of correct responses were based on a

BOX 9.7 (*cont'd*)

much smaller number of trials and as such tend to be subject to much more sampling error. In about 45% (i.e., the goodness-of-fit p -value) of experiments such as Experiment B, a mechanism that acts according to the model would produce results just as messy and different from the model as the results that were actually observed. Therefore, we cannot rule out that the mechanism that produced the data did act in accordance with the model, and thus we say that the goodness-of-fit of the model is "good."

The reader may have gotten the impression that a researcher should not collect too much data because apparently one's data need to be a lot better behaved with a large number of observations as compared to a smaller number of observations. Well, "yes" and "no." "No" because data will automatically become better behaved as one collects more trials. That is, as one collects more trials, the observed proportions of positive responses will tend to approach the true probabilities of positive responses. As such, as long as the model that you fit does in fact accurately describe these true probabilities, increasing the number of trials will not negatively affect your goodness-of-fit. More specifically, as long as the model that you fit is correct, the probability density function of the goodness-of-fit p -value will be a rectangular distribution between the values 0 and 1. In other words, as long as your model is correct, there is a 0.05 probability that your p -value will be less than 0.05 (the common criterion by which a model is deemed to provide a poor fit), regardless of how many trials were collected. "Yes" because, in practice, any model you fit to your data will be wrong. For example, the fits to both experiments assumed that the true values for the lower and upper asymptotes equal 0.02 and 0.98, respectively. They are not equal to these values. They may have values close to these quantities, but they will not be exactly equal to these values. Also, the shape of the generating function will not exactly follow the Logistic function. Whenever the model fitted to data does not match the generating mechanism exactly, a goodness-of-fit test will tend to result in increasingly poorer goodness-of-fits with an increasing number of data points.

Another reason not to collect a small amount of data is that generally one does not conduct an experiment only to fit a single model to the data collected and demonstrate that this model provides a good fit to these data. If that was one's goal, the above shows that all one needs to do is collect very few trials. Even if the model is very wrong, a goodness-of-fit test will likely indicate that the fit is good if you collect few enough data! As we argued above, all this would mean is that there is not enough evidence to demonstrate that the model is wrong. However, one generally performs an experiment to test assumptions regarding the effect some variable has on a sensory or perceptual process. For example, one might be interested in whether detection of some stimulus is affected by adaptation to another stimulus. This can be tested by performing a model comparison between a model that states that the generating function in the adaptation condition and the generating function in the no-adaptation condition are equal to a model that states that they are not equal. [Section 9.2](#) discusses how such a model comparison can be carried out. In order to demonstrate that adaptation does affect detection one would need to collect enough data to

BOX 9.7 (*cont'd*)

reject the model that incorporated the NH, which says that adaptation does not affect detection. Thus, while collecting very few trials will likely lead to well-fitted data (as measured by the model's goodness-of-fit), it will also mean that one will likely not be able to argue that some experimental manipulation affects performance. All of the above makes the issue of how many trials to use in one's experiment complex. On the one hand, one wants to use a number of trials that is sufficient to demonstrate an effect of the experimental manipulation. On the other hand, as we argued above, one wants to use few enough trials to obtain a well-fitting model by the criterion of the goodness-of-fit.

Another common misunderstanding regarding goodness-of-fit tests states that more complex models will tend to have better goodness-of-fit merely because they are more flexible and can thus display a closer correspondence between the observed and modeled proportions of correct responses. Let us consider an example. Imagine that trials are collected in two experimental conditions but that the generating functions in the two conditions are actually identical. Let's say that two models are under consideration: model L (for "lesser") states that the generating functions were equal between the two conditions and model F (for "fuller") states that the generating functions were not equal. Let's say both models assume a Logistic function and further assume that the guess rate equaled 0.5 and the lapse rate equaled 0.02. Model L requires the estimation of a single, shared threshold and a single, shared slope. Model F requires the estimation of two thresholds (one for each condition) and two slopes.

Since Model L is nested under Model F, Model F will always have a greater likelihood value as compared to that of Model L. However, if Model L is correct, Model F has no advantage in terms of the goodness-of-fit *p*-value. Which, by the way, is not to say that the goodness-of-fit *p*-values would then be equal between the two models. Sometimes Model F would have better goodness-of-fit, sometimes Model L. More specifically, if the data happen to be similar between the two conditions in the experiment, model L will tend to have better goodness-of-fit, but if the data happen to be dissimilar, model F will tend to have better goodness-of-fit. Insofar as the assumptions that are made by both models (Logistic, guess rate = 0.5, lapse rate = 0.02) are correct, the probability density function of goodness-of-fit *p*-values for both models will be a rectangular distribution bound between 0 and 1. It is important to remember that the above is the case only in case neither model makes any false assumptions. In case an assumption that Model L makes (but Model F does not) is false, goodness-of-fit will tend to be better for Model F as compared to Model L.

So far, all of the issues discussed result from an improper interpretation of the *p*-value and can be remedied, in theory, by getting researchers not to make conclusions that are not warranted by the *p*-value. Strictly speaking, based on a *p*-value alone, we will then not be able to conclude that the NH is false or even that is likely to be false.

More damaging to the NH test is the argument that the NH is almost always false. It may even be argued that we should drop “almost” from the previous sentence. Let us use an example that we used before and that should not be contentious on this point. In Section 9.3.2 we presented a hypothetical experiment that tested whether a coin was fair. That is, does the coin have an equal chance of coming up heads as it does tails? In reality, no experiment is needed here to reject the fairness of this or any other coin. The probability that any given coin will produce heads is not going to equal 0.5 exactly. It may be infinitesimally near 0.5, but it will not be exactly equal to 0.5. As such, whether the NH gets rejected is not so much a function of whether it is true or not (it is not) but rather a function of the experiment we perform. If we do not want to reject the NH, we flip the coin a very modest number of times. If, on the other hand, we wish to reject the NH we flip the coin a very large number of times. If we take a stab at how unfair the coin might be, we can even figure out how many times we will need to flip the coin in order to have a probability of at least, say, 0.99 of successfully rejecting the NH.

So, how should one deal with all these issues and others we have not mentioned here? Does the p -value give us any valuable information? Should we abandon the p -value? If so, how should we go about evaluating the effect of our experimental manipulation? These issues have been discussed for a long time. We estimate that the number of opinions that have been expressed on these issues is roughly equal to the number of statisticians that have expressed an opinion on them. Shortly before this second edition went to press, the journal *Basic and Applied Social Psychology* (BASP) added some fuel to the fire by announcing in their first issue of 2015 that “From now on, BASP is banning the NHSTP [Null Hypothesis Significance Testing Procedure]” ([Trafimow and Marks, 2015](#), p. 1). In a “comment” ([Wasserstein, 2015](#)), the executive director of the American Statistical Association, Ronald Wasserstein, acknowledges the concern that this decision has created in the statistics community and notes further that a group of distinguished professionals is preparing a statement on the issues surrounding p -values and NH testing that is to be released later in 2015 (unfortunately after this second edition went to press).

9.4 SOME ALTERNATIVE MODEL COMPARISON METHODS

As discussed, the likelihood ratio test is an NH test and as such is subject to all the issues discussed in the previous section. Moreover, the likelihood ratio test described above can only be used to compare two models when one of the models is nested under the other. Some research questions require us to make a decision between two or more models that are not nested. In such cases, the likelihood ratio test cannot be used. Here we discuss some other methods that can be used to select the best model from a range of possible models of some data. It should be noted that these methods also suffer from many of the issues that the NH suffers from and are discussed above.

9.4.1 Information Criteria: AIC and BIC

This section will briefly discuss some methods that can be used to select between any two models for which a likelihood can be calculated, whether they are nested or not. Remember

that using the likelihood as a metric in which to compare the goodness-of-fit of models directly is inappropriate. The reason is that this would unjustly favor models that have many parameters. For example, adding parameters to a model would always be preferred if we judge the fit of the model by likelihood only, because adding parameters can only increase the likelihood. However, by the scientific principle of parsimony, simpler models should be preferred over more complex models. In model selection, then, the question is whether any increase in likelihood that results from the inclusion of additional parameters is sufficiently large to warrant the inclusion of the extra parameters. Akaike's (1974) Information Criterion (AIC) is a measure of the relative goodness-of-fit of a model. AIC rewards increases in the likelihood but simultaneously penalizes models for complexity (as measured by the number of free parameters included in the model). The AIC_i of any model M_i is given as

$$AIC_i = -2LL(\hat{\theta}|\mathbf{y}, M_i) + 2K_i \quad (9.4)$$

where $LL(\hat{\theta}|\mathbf{y}, M_i)$ is the log-likelihood for model M_i using maximum likelihood estimates for its parameters θ , based on the observed data y , and K_i is the number of free parameters in model M_i . The reader should not get the impression that the particular formulation of the AIC, particularly the factor 2 with which K is multiplied, is arbitrary. The derivation of the equation for AIC is firmly grounded in information theory but is well beyond the scope of this text.

Note that increases in log-likelihood and decreases in the complexity of the model (both of which are to be favored) lead to smaller values of AIC_i . Thus, models with smaller associated AIC values are preferred over models with higher AIC values. Note that only the relative value of the AIC is informative, as AIC is greatly dependent on the particulars of the experiment, most notably the number of trials. For this reason we should not compare AIC values between models unless the AIC values are based on the same set of observations, y .

Let us revisit our trend analysis of Section 9.3.4.1.1. There we wished to determine whether the bend in the line describing thresholds as a function of adaptation duration was real or not. In order to do so, we compared model D to model C (model fits are shown in Figure 9.9). Model D allows thresholds to follow a second-order polynomial, while model C constrains them to vary according to a first-order polynomial. We performed a likelihood ratio test and concluded that the observed bend in the line would probably not have occurred in case the true trend followed a first-order polynomial. Thus, model D was preferred over model C. Let us now compare models C and D using AIC. The log-likelihood associated with model C is -1.5542×10^3 and that of model D is -1.5506×10^3 . Model C has four free parameters (a common slope, a common lapse rate, and two additional parameters to code the first-order polynomial), while Model D has five (a common slope, a common lapse rate, and three additional parameters to code the second-order polynomial).

Thus

$$AIC_C = -2 \times -1.5542 \times 10^3 + 2 \times 4 = 3.1164 \times 10^3$$

and

$$AIC_D = -2 \times -1.5506 \times 10^3 + 2 \times 5 = 3.1112 \times 10^3$$

Model D has a lower AIC and is thus preferred by this criterion. Since the AIC is a relative measure of fit and its absolute value is of no consequence for model selection, it is common practice to report the differences in AIC between models, rather than their absolute values. **Table 9.7** lists, in order of fit (best to worst), the differences between the AIC values of all five models shown in [Figure 9.9](#) and that of the best-fitting model (model D).

Also shown in [Table 9.7](#) are differences between Bayesian Information Criterion (BIC) values of the models, given as

$$\text{BIC}_i = -2\text{LL}(\hat{\theta}|\mathbf{y}, M_i) + \log_e(n)K_i \quad (9.5)$$

where the shared terms are as in AIC, and n is the number of observations on which the likelihood is based. In other words, the penalization for the inclusion of additional parameters increases with the sample size. We note that the penalization of additional parameters is greater in BIC as compared to AIC (except for the most modest of sample sizes: $\log_e(n) < 2$). Note that the ranking of models under the BIC criterion differs from that under the AIC criterion. The best model under the BIC criterion is model C, although model D (the best model under the AIC criterion) is a close second.

9.4.2 Bayes Factor and Posterior Odds

Model comparisons may also be performed using the Bayes Factor (BF). The BF gives researchers the opportunity to incorporate their prior beliefs regarding parameter values in the form of prior distributions across the parameter space. The BF is given as

$$\text{BF} = \frac{\int L(\theta_1|\mathbf{y}; M_1)p(\theta_1)d\theta_1}{\int L(\theta_2|\mathbf{y}; M_2)p(\theta_2)d\theta_2} \quad (9.6)$$

where $L(\theta_i|\mathbf{y}; M_i)$ is the likelihood function of parameter (or parameter set) θ_i of model M_i , having observed responses \mathbf{y} , and $p(\theta_i)$ is the prior distribution on parameter space θ_i . The quantity $\int L(\theta_1|\mathbf{y}; M_1)p(\theta_1)d\theta_1$ is termed the marginal likelihood for model M_i . In other words, the BF is somewhat similar to the likelihood ratio, except that it uses the mean of the likelihood function (rather than its mode as the likelihood ratio does), and it weighs the likelihood function by a prior distribution before determining its mean. A $\text{BF} > 1$ favors

TABLE 9.7 ΔAIC and ΔBIC values for the five models shown in [Figure 9.9](#)

Model	ΔAIC	ΔBIC
D	0	0.7050
E	3.4455	10.1568
C	5.3014	0
B	26.9578	15.6500
A	29.0635	11.7494

M_1 and a $\text{BF} < 1$ favors M_2 . The computation of the marginal likelihood is generally nontrivial and must, in most practical applications, occur by numerical integration.

Researchers also have the opportunity to incorporate prior beliefs regarding the relative likelihoods of the two models by applying Bayes Theorem (Section 4.3.3.2.1) to obtain the “posterior odds.” Prior beliefs regarding the relative likelihoods of M_1 and M_2 are expressed as “prior odds,” which are given as

$$\text{prior odds} = \frac{p(M_1)}{p(M_2)} \quad (9.7)$$

Note that $p(M_1)$ and $p(M_2)$ need not sum to 1. The posterior odds may be obtained by applying Bayes Theorem:

$$\text{posterior odds} = \frac{p(M_1|\mathbf{y})}{p(M_2|\mathbf{y})} = \text{BF} \frac{p(M_1)}{p(M_2)} \quad (9.8)$$

Note that in case the prior probability of model M_1 equals that of M_2 , the posterior odds simply equal the BF. Usage of BF or posterior odds does not require that models M_1 and M_2 are nested.

FURTHER READING

Hays (1994) is a classic text on frequentist statistical methods. The model comparison approach emphasized in this chapter is developed much more thoroughly (but in the context of least squares error criterion methods) by Judd et al. (2008). Cohen (1994) discusses in a very readable manner some of the issues of NH testing. An introduction to Bayesian statistics may be found in Jaynes (2003). Burnham and Anderson (2002) provide a thorough introduction to the information-theoretic approach to model selection that underlies AIC and BIC.

EXERCISES

1. A researcher conducts an experiment with two conditions. He then performs a model comparison using the likelihood ratio test in order to determine whether the PFs differ between the two conditions. He assumes that a Logistic function describes the underlying mechanism well and also assumes that the lapse rate equals 0. The p -value for the model comparison equals 0.5604.
 - a. Does this mean that he can conclude that the Logistic function describes the data well?
 - b. Does this mean that the lapse rate does not differ significantly from 0?
 - c. What may the researcher conclude?
2. This question refers to the example data given in Table 9.1.
 - a. In the text (Section 9.2.3.1) it was tested whether adaptation affected the detection threshold. Repeat this test but now assume that the slopes are identical between conditions. Compare the results to that given in the text.

- b.** In the text (Section 9.2.3.2) it was tested whether adaptation affected the slope parameter of the PF describing detection performance. Repeat this test but now assume that the thresholds are identical between conditions. Compare the results to that given in the text.
- c.** How would you go about determining whether the assumption that the thresholds were equal between conditions is a valid assumption?
- 3.** This question refers to the example data given in Section 9.2.5.
- Use contrasts to test whether the threshold at the adaptation duration of 0 s differs significantly from that at 4 s.
 - Use contrasts to test whether it is reasonable to believe that the differences that exist among the threshold estimates at adaptation durations 4, 8, and 12 s occurred by sampling error alone. Do this using a single model comparison only.
- 4.** Below is a table which lists the assumptions four models make regarding the two PFs and their parameters in the two different conditions in an experiment.

	Model A	Model B	Model C	Model D	Model E
PF	Logistic	Logistic	Logistic	Gumbel	Gumbel
Thresholds	Unequal	Unequal	Equal	Equal	Unequal
Slopes	Unequal	Equal	Equal	Equal	Equal
Guess rate	0.5	0.5	0.5	0.5	0.5
Lapse rate	0	Equal	0	Equal	Unequal

- Which pairs of models may be compared using the likelihood ratio test?
 - How many free parameters does each of the models have?
 - For each of these comparisons, what may be concluded if a significant difference (i.e., $p < 0.05$) is obtained?
 - Which models may be compared against the saturated model?
- 5.** Verify the ΔBIC values given in Table 9.7.

References

- Akaike, H., 1974. A new look at the statistical model identification. *IEEE Trans. Automat. Contr.* 19, 716–723.
- Berkson, J., 1980. Minimum chi-square, not maximum likelihood! *Ann. Stat.* 8 (3), 457–487.
- Burnham, K.P., Anderson, D.R., 2002. Model Selection and Multimodal Inference. A Practical Information-theoretical Approach, second ed. Springer, New York, NY.
- Cohen, J., 1994. The earth is round ($p < 0.05$). *Am. Psychol.* 49, 997–1003.
- Hays, W.L., 1994. Statistics. Wadsworth Group/Thomson Learning, Belmont, CA.
- Jaynes, E.T., 2003. Probability Theory. The Logic of Science. Cambridge University Press, New York, NY.
- Judd, C.M., McClelland, G.H., Ryan, C.S., 2008. Data Analysis. A Model Comparison Approach. Routledge, New York, NY.
- Prins, N., 2008. Correspondence matching in long-range apparent motion precedes featural analysis. *Perception* 37, 1022–1036.
- Prins, N., Kingdom, F.A.A., 2009. Palamedes: Matlab routines for analyzing psychophysical data. <http://www.palamedestoolbox.org>.

- Prins, N., Streeter, K.D., 2009. Perceptual learning of texture segmentation is specific for retinal location but not first-order orientation channel. *J. Vis.* 9 (8), 868 [abstract].
- Trafimow, D., Marks, M., 2015. Editorial. *Basic Appl. Soc. Psych.* 37 (1), 1–2.
- Wasserstein, R., April 14, 2015. ASA comment on a Journal's ban on Null Hypothesis Statistical Testing. <http://community.amstat.org/blogs/ronald-wasserstein/2015/02/26/asa-comment-on-a-journals-ban-on-null-hypothesis-statistical-testing>.
- Wichmann, F.A., Hill, N.J., 2001. The psychometric function: I. Fitting, sampling, and goodness of fit. *Percept. Psychophys.* 63, 1293–1313.

Quick Reference Guide

Absolute threshold Traditional term for the magnitude of a stimulus that is just discriminable from its null, as exemplified by a contrast detection threshold.

ABX Term used in Signal Detection Theory for a match-to-sample task with two match alternatives, i.e., the observer selects a previously viewed sample stimulus (X) from two alternative match stimuli (A,B). In this book the task is termed 2AFC match-to-sample.

Accuracy Denotes how close a sensory measurement is to its corresponding physical measurement. A typical measure of accuracy is the reciprocal of the difference between the perceived and physical measurements. Example: the perceived midpoint between two dots is accurate if it is close to the physical midpoint.

Acuity *See* Visual acuity.

Adaptive procedure Also termed a staircase procedure. An efficient method for estimating the parameters of a psychometric function (PF) in which the stimulus magnitude on each trial is based on the observer's responses on previous trials, such that the amount of information gained from the trial is optimized.

Additive noise Internal noise that is constant with stimulus magnitude.

Additive summation When the signals from two or more stimuli are summed within the same sensory channel or mechanism.

Adjustment *See* Method of adjustment.

Akaike's Information Criterion (AIC) Measure of model fit that may be used to compare the fits of two or more models to a single data set.

$$AIC_i = -2LL(\hat{\theta}|\mathbf{y}, M_i) + 2K_i$$

where $LL(\hat{\theta}|\mathbf{y}, M_i)$ is the log-likelihood for model M_i using maximum likelihood estimates for its parameters $\hat{\theta}$, based on the observed data \mathbf{y} , and K_i is the number of free parameters in model M_i . Smaller AIC values indicate better fit.

Arcdeg Abbreviation for arc degrees. Measure of visual angle. An arc degree is 1/360 of a full circle, or $\pi/180$ radians.

Arcmin Abbreviation for arc minutes. Measure of visual angle. An arc minute is 1/60 of an arc degree, 1/21,600 of a full circle, or $\pi/10,800$ radians.

Arcsec Abbreviation for arc seconds. Measure of visual angle. An arc second is 1/3600 of an arc degree, 1/60 of an arc minute, 1/1,296,000 of a full circle, or $\pi/648,000$ radians.

Asymmetric brightness matching Procedure in which the observer matches the brightnesses of two stimuli set in different contexts in order to obtain their point-of-subjective-equality, or PSE.

Asymptotic performance level Stimulus intensity that is so high that it is safe to assume that an incorrect response made is due to an attentional lapse or finger error.

Bayes Factor (BF) Expresses the relative evidence for two models provided by some data.

$$BF = \frac{\int L(\theta_1|\mathbf{y}; M_1)p(\theta_1)d\theta_1}{\int L(\theta_2|\mathbf{y}; M_2)p(\theta_2)d\theta_2}$$

where $L(\theta_i|\mathbf{y}; M_i)$ is the likelihood function of parameter (or parameter set) θ_i of model M_i , having observed responses \mathbf{y} , and $p(\theta_i)$ is the prior distribution on parameter space θ_i . $BF > 1$ favors model 1, $BF < 1$ favors model 2.

Bayes' Theorem A general statement of Bayes' Theorem is

$$P(H|D) = \frac{p(H)p(D|H)}{p(H)p(D|H) + p(\bar{H})p(D|\bar{H})} = \frac{p(H)p(D|H)}{p(D)}$$

where $p(H)$ is the prior probability of hypothesis H , $p(D|H)$ is the probability of obtaining data D , assuming hypothesis H (i.e., the likelihood), and $p(H|D)$ is the posterior probability of hypothesis H . Bayes' Theorem allows us to adjust our prior beliefs regarding H based on our empirical results D .

Bayesian information criterion (BIC) Measure of model fit that may be used to compare the fits of two or more models to a single data set.

$$\text{BIC}_i = -2LL(\hat{\boldsymbol{\theta}}|\mathbf{y}, M_i) + \log_e(n)K_i$$

where $LL(\hat{\boldsymbol{\theta}}|\mathbf{y}, M_i)$ is the log-likelihood for model M_i using maximum likelihood estimates for its parameters $\boldsymbol{\theta}$, based on the observed data \mathbf{y} , n is the number of observations on which the likelihood is based, and K_i is the number of free parameters in model M_i . Smaller BIC values indicate better fit.

Best PEST Adaptive method for estimating a threshold. On each trial a maximum likelihood estimate is made of the threshold using the responses from previous trials and assuming a particular shape of psychometric function. The stimulus magnitude on the subsequent trial is then set to the threshold estimate.

Bias (of estimate) The difference between a parameter's true value and the expected value of its estimate.

Bias (of observer) Observer bias has two related meanings. The first is the tendency to make more of one type of response than another in performance tasks. For example, in a two-interval forced-choice (2IFC) task the observer might be biased toward responding "first interval" even if the target was equally likely to occur in both intervals. Second, in appearance tasks observer bias can refer to the difference between the point of subjective equality and the point of physical equality. For example, in a Vernier alignment task the point of subjective alignment might be biased away from the point of physical alignment.

Binocular rivalry The phenomenon in which stimuli presented to the two eyes alternate in perceptual dominance.

Binocular summation The internal summation of signals from the two eyes.

Binomial coefficient Formula for calculating the total number T of unique combinations of N different events, with k different events per combination.

$$T = \frac{N!}{k!(N-k)!}$$

Note that $N! = N \times (N-1) \times (N-2) \times \dots \times 1$ and that $0!$ equals 1 by definition. For example, if you have $N = 5$ stimulus magnitudes, with $k = 3$ different stimulus magnitudes presented per trial, there are a total of $T = 10$ unique combinations of stimulus magnitudes.

Bisection scaling See Partition scaling.

Bisection task Task to measure the perceptual midpoint between two stimuli that lie at different points along a stimulus dimension. Examples: to measure the perceived midpoint of a line or the midpoint in perceived contrast between two different contrasts.

Bootstrap method Method used to estimate a parameter's sampling distribution through repeatedly simulating an experiment using known or assumed parameter values. The empirical sampling distribution is then used to determine the standard error of estimate of the parameter.

Brightness The perceptual correlate of luminance or light intensity.

Cancellation procedure See Nulling procedure.

Channel A sensory mechanism that is selective for a particular stimulus attribute.

Chromaticity Specification of the color of an object regardless of its luminance, referring to both the colorfulness (or saturation) and hue.

Class A observation Term coined by Brindley (1970) for the psychophysical observation in which two physically different stimuli are perceptually indiscriminable.

Class B observation Term coined by Brindley (1970) for the psychophysical observation in which two physically different stimuli remain perceptually discriminable even when matched along one or more stimulus dimensions.

Compound stimulus A stimulus in which two or more stimuli are present. The stimuli can differ along one or more dimensions or be identical stimuli in different locations.

Constant noise See Additive noise.

Contrast A measure of the relative luminance between two stimuli. Measures of contrast include Weber contrast, Michelson contrast, and root mean square (RMS) contrast. The contrast of an object with its surround is invariant to changes in the intensity of illumination.

Contrast sensitivity The reciprocal of contrast threshold.

Contrast threshold The amount of luminance contrast required to reach a criterion level of detection performance.

Criterion Usually denotes the bias of an observer toward making one type of response over another in a psychophysical task.

Criterion C Measure of bias in a forced-choice experiment derived by signal detection analysis. For one-alternative forced-choice (1AFC) tasks C is defined as:

$$C = -\frac{[z(pH) + z(pF)]}{2}$$

where $z(pH)$ and $z(pF)$ are the z-values for the proportion of hits and false alarms, respectively.

Criterion C' ("C-prime") As above but normalized to d' :

$$C' = -\frac{[z(pH) + z(pF)]}{2[z(pH) - z(pF)]}$$

where $z(pH)$ and $z(pF)$ are z-values for the proportion of hits and false alarms, respectively.

Criterion-dependent A psychophysical task or procedure in which observers are likely to be biased toward making one type of response over another, or a psychophysical measurement provided by a biased observer.

Criterion-free A psychophysical task or procedure in which observers are unlikely to be biased toward making one type of response over another, or a psychophysical measurement that is provided by an unbiased observer or computed in such a way as to take into account bias.

Criterion $\ln\beta$ Measure of bias in a forced-choice experiment derived by signal detection analysis, defined as:

$$\ln\beta = \ln \frac{\phi[z(pH)]}{\phi[z(pF)]}$$

where $\phi[z(pH)]$ and $\phi[z(pF)]$ are the ordinate values of a standardized normal distribution corresponding to the z-values for the proportions of hits and false alarms, respectively.

Cross-modal matching Method for measuring the apparent magnitude of a stimulus by matching it to a stimulus in another sensory modality. For example, the perceived slant of a visual texture might be measured by hand-adjusting the slant of an object with a planar surface.

Cumulative normal function

$$F_N(x; \alpha, \beta) = \frac{\beta}{\sqrt{2\pi}} \int_{-\infty}^x \exp\left(-\frac{\beta^2(x - \alpha)^2}{2}\right)$$

The cumulative normal function is the integral of the normal distribution, where α determines the location (threshold) and β determines the slope of the function.

d' (d-prime) Measurement of observer sensitivity or stimulus discriminability derived from Signal Detection Theory. On the assumption that the stimuli are represented internally as random variables drawn from normal distributions with given means and variances, d' is a measure of the distance between the means of the distributions normalized to their common standard deviation.

Decibel The decibel (dB) is a logarithmic unit used to express the ratio between two values of a physical quantity. One of the values is typically a reference or baseline value. For example, when applied to contrast a decibel is usually defined as:

$$20 \log_{10}\left(\frac{C_{\text{ref}}}{C}\right)$$

where C_{ref} is the reference contrast and C is a contrast value. If C_{ref} is the maximum contrast, a decibel is a measure of physical contrast attenuation and can be used as a behavioral measure of contrast sensitivity. For example, if $C_{\text{ref}} = 1$, a C of 0.05 is approximately 26 dBs. *See also* Summation ratio.

Detection Usually refers to tasks/procedures/experiments that measure the magnitude of a stimulus that can be discriminated from its null. Examples: measurement the detection of contrast when the null stimulus is a blank field, or measurement of the detection of curvature when the null stimulus is a straight line. The term is also used to denote the measurement itself, e.g., a detection threshold.

Deviance Term that is used for the transformed likelihood ratio (TLR) when the fuller model is the saturated model. Deviance is used to test the goodness-of-fit of the model.

Difference threshold Traditional term for the magnitude of a stimulus difference that is just detectable when both stimuli are above detection threshold.

Differencing strategy Strategy of basing forced-choice decisions on the perceived difference between stimuli. Differencing strategies have been suggested to mediate perceptual decisions in the standard 2AFC, same-different, oddity, and match-to-sample tasks.

Dipper function The part of a threshold-versus-pedestal or threshold-versus-mask function where thresholds are lower in the presence compared to the absence of the pedestal/mask.

Discriminand One of the stimuli in a discrimination experiment.

Discrimination Most commonly refers to tasks/procedures/experiments that determine the just noticeable difference (JND) in stimulus magnitude between two stimuli with nonzero magnitude. Also used to denote the type of measurement itself, e.g., a discrimination threshold.

Discrimination scale A perceptual scale derived by integrating JNDs. Also termed a Fechnerian scale.

Equisection scaling *See* Partition scaling.

Exponent of transducer function The exponent of the power function relating the physical intensity or amplitude of a stimulus to its perceptual or internal response. Thus:

$$\psi = gx^\tau$$

where ψ is the internal response, x is stimulus intensity, g a stimulus scaling factor, and τ the transducer exponent.

Exposure duration (of stimulus) See Stimulus exposure duration.

False alarm Responding that a stimulus is present when it is not.

Fechnerian integration The method of deriving a perceptual scale by summing or integrating discrimination thresholds (or JNDs).

Fechnerian scaling See Discrimination scale.

Fixed Attention Window Scenario in which the observer monitors all the channels or mechanisms that might potentially be activated by a stimulus.

Fixed parameter A model parameter that is not allowed to vary during model fitting.

Forced-choice Term used here to refer to any task/procedure/experiment in which the observer is required on each trial to make a response from a predefined set of choices. In the signal detection literature the term tends to be used more restrictively to refer to tasks in which the observer selects a target from two or more stimulus alternatives.

Free parameter A model parameter that is varied during model fitting in order to optimize the fit of the model.

Fuller model In a statistical model comparison, the fuller model is the less parsimonious of the two models that are compared.

Geometric mean The antilog of the mean of the logarithm of a set of numbers. If the numbers are X_1, X_2, \dots, X_n , the geometric mean computed using logarithms to the base 10 is given by:

$$10^{\left(\frac{\sum_{i=1}^N \log X_i}{N}\right)}$$

Geometric series A series of numbers in which adjacent pairs have identical ratios.

Goodness-of-fit test A statistical model comparison between two models in which the fuller model is the saturated model. The saturated model makes the assumptions of stability and independence only. As such, a goodness-of-fit test tests all the assumptions of the lesser model, except for the assumptions of stability and independence, simultaneously.

Grating induction The illusory brightness modulation observed in a uniform stripe running at right angles to the bars of a real luminance grating.

Guess rate Corresponds to chance level performance: the expected proportion correct for a hypothetical observer who guesses on each trial. Note that the term is based on assumptions of the discredited High-Threshold Theory (under the framework of Signal Detection Theory an observer never truly guesses). The guess rate is the parameter of a psychometric function that corresponds to the lower asymptote of a psychometric function (γ).

Gumbel function

$$F_G(x; \alpha, \beta) = 1 - \exp(-10^{(\beta(x-\alpha))})$$

where α determines the location (threshold) and β determines the slope of the function. The Gumbel function is the analog of the Weibull function when a log transform on x is used and, for that reason, is sometimes referred to as the log-Weibull function or simply, but confusingly, as the Weibull function.

High-Threshold Theory A theory of detection that states that detection occurs only when the sensory evidence exceeds an internal criterion or threshold. The threshold is set such that it will not be exceeded in the absence of a stimulus (i.e., by noise alone). While the central tenets of High-Threshold Theory have been discredited, many of the terms used in psychophysics (e.g., "threshold," "guess rate") are remnants of the theory.

Hit Responding that a stimulus is present when it is present.
Hyperbolic Secant function

$$F_{HS}(x; \alpha, \beta) = \frac{2}{\pi} \tan^{-1} \exp\left(\frac{\pi}{2} \beta(x - \alpha)\right)$$

where α determines the location (threshold) and β determines the slope of the function.

Identification Sometimes used as an alternative term to “discrimination,” especially when the observer has to not only detect a stimulus but also identify some additional stimulus property, such as whether the stimulus is red or green, moving leftward or rightward, behind or in front. Sometimes also used instead of the term “recognition.”

Independence, assumption of In the context of model fitting, this assumption states that the probability of observing a particular response (“yes,” “first interval,” etc.) on any given trial is independent of observations made on other trials.

Independent observation strategy Observer strategy of basing a forced-choice decision on the independent assessment of the likelihood that each observation is from a particular stimulus. Independent observation strategies have been suggested to underlie certain types of same-different, oddity, and match-to-sample tasks.

Internal noise The random fluctuation in the observer’s internal representation of a stimulus magnitude.

Inter-stimulus-interval (ISI) The temporal interval between the offset of one stimulus and the onset of another.

Inter-trial-interval (ITI) The temporal interval between the end of one trial and the beginning of the next trial.

Interval scale A perceptual scale in which the differences in scale values are proportional to perceived differences in stimulus magnitude. An interval scale can be rescaled by $aX + b$ without loss of information where a and b are arbitrary constants.

Joint probability Probability of two or more independent events occurring together. If each event has a probability of p then the probability of n events occurring together is p^n .

Just noticeable difference (JND) The smallest difference in stimulus magnitude that is just discriminable.

Lapse rate The probability of an incorrect response that is independent of the stimulus. Lapses are most evidenced by incorrect responses to stimulus magnitudes that are considerably above threshold. Lapse rate is the parameter of a PF (λ) that determines the upper asymptote $(1 - \lambda)$.

Legge-and-Foley transducer function A transducer function of the form

$$\psi = \frac{x^p}{z + x^q}$$

where ψ is the internal response, x is stimulus intensity, and p, q , and z are constants. Typically both p and q are positive and greater than unity, resulting in a function of x that first accelerates and then decelerates.

Lesser model In a statistical model comparison, the lesser model is the more parsimonious of the two models that are compared.

Lightness The perceptual correlate of the reflectance or perceived “shade-of-gray” of an object.

Likelihood The probability with which a hypothetical observer characterized by assumed model parameters would reproduce exactly the responses of a human observer. The likelihood is a function of parameter values, not responses. The likelihood serves as the metric in which “best-fitting” is defined in maximum likelihood estimation.

Likelihood ratio The likelihood ratio (LR) is the ratio of likelihoods of two events. If expressed in terms of probabilities, the likelihood ratio is the ratio of probabilities that two events will occur.

Linear summation A special case of additive summation in which signals from multiple stimuli are linearly combined.

Log-likelihood Logarithmic transform (base e) of the likelihood. See also Likelihood.

Logarithmic spacing Spacing of numbers according to a geometric series, i.e., in which the ratios of adjacent pairs of numbers are the same. The i th value of a set of n logarithmically spaced values starting with a and ending with b is:

$$x(i) = 10^{\left[\log a + \frac{(i-1)\log(b/a)}{(n-1)}\right]}$$

Logarithmically spaced values can be computed in MATLAB® using: `>>x=logspace(log10(a),log10(b),n)`

Logistic function

$$F_L(x; \alpha, \beta) = \frac{1}{1 + \exp(-\beta(x - \alpha))}$$

where α determines the location (threshold) and β determines the slope of the function.

log-Quick function

$$F_{IQ}(x; \alpha, \beta) = 1 - 2^{(-10^{(\beta(x-\alpha))})}$$

where α determines the location (threshold) and β determines the slope of the function.

Luminance Measure of light intensity. Common measures are candelas per square meter (cd/m^2) or foot-lamberts (fl or ft-L).

Magnitude estimation Method for deriving a perceptual scale in which observers provide a numerical estimate of the perceived magnitudes of the stimulus.

Match-to-sample Forced-choice procedure in which the observer views a “sample” stimulus and then selects the sample from a number of alternative “match” stimuli. The minimum number of stimuli is 3: one sample, two match.

Matched Attention Window Scenario in which the observer monitors only the channels or mechanisms that are activated by a stimulus.

Matrix A two-dimensional array of numbers.

MAX rule The decision rule in a forced-choice detection task in which the observer chooses the alternative/interval that contains the biggest internal signal.

Maximum Likelihood Difference Scaling (MLDS) Method for deriving an interval perceptual scale from judgements about perceived stimulus differences, in which the perceptual values corresponding to each stimulus magnitude are estimated using a maximum likelihood criterion.

Maximum likelihood estimation Estimation procedure in which the best-fitting model is defined to be the model that maximizes the likelihood function.

Metamers Stimuli that are physically different yet perceptually indiscriminable.

Method of adjustment Method in which observers freely adjust the magnitude of a stimulus in order to reach a criterion, for example a threshold or point of subjective equality (PSE).

Method of constants Method in which the magnitude of the stimulus presented on each trial is selected from a predefined set.

Method of limits Method in which observers are presented with a series of stimuli of either increasing (ascending method of limits) or decreasing (descending method of limits) magnitude and report when the stimulus appears to change state, e.g., from visible to invisible or vice versa. A threshold is considered to be the stimulus magnitude at which the change of state occurs. Typically, the ascending and descending methods are used alternately and the thresholds from each are averaged, minimizing errors due to habituation and expectation.

Method of paired comparisons Method for deriving a perceptual scale involving stimulus pairs. On each trial two stimuli are selected from a range of stimuli and the observer decides which has the higher perceived magnitude. The set of pair responses are used to derive estimates of the perceptual values corresponding to each stimulus magnitude.

Method of quadruples Method for deriving an interval perceptual scale involving four stimuli per trial. The stimuli are presented in two pairs and the observer decides which pair is more similar (or more different). The set of quadruple responses are used to derive estimates of the perceptual values corresponding to each stimulus magnitude.

Method of triads Method for deriving an interval perceptual scale involving three stimuli per trial. One of the stimuli is allocated as the target and the observer decides which of the two remaining stimuli is most similar (or different) to the target. The set of triad responses are used to derive estimates of the perceptual values corresponding to each stimulus magnitude.

Michelson contrast Defined as $(L_{\max} - L_{\min}) / (L_{\max} + L_{\min})$ where L_{\max} and L_{\min} are the maximum and minimum luminances. Michelson contrast is the favored metric of contrast for periodic stimuli such as sinusoidal gratings but is also applicable to any stimulus defined by two luminance levels.

Minkowski summation Minkowski summation is a way of expressing the amount of summation with multiple stimuli. Typically expressed in terms of behavioral sensitivity, Minkowski summation is defined as:

$$S_{\text{cmb}} = \left[\sum_{i=1}^n S_i^m \right]^{1/m}$$

where S_{cmb} is sensitivity, typically the reciprocal of detection threshold, of the stimulus combination, S_i is sensitivity to the i th stimulus presented alone, n is the number of stimuli, and m is the parameter that expresses the inverse of the degree of summation. m is sometimes termed the Minkowski exponent.

Monochromatic Light composed of a single or very narrow band of wavelengths.

Müller–Lyer illusion The illusory difference in length between a line with acute-angle fins at both ends and a line with obtuse-angle fins at both ends.

Multipartition scaling Also termed the “simultaneous solution,” in partition scaling the observer adjusts the magnitudes of a range of stimuli until they appear at equal perceptual intervals. The first and last stimulus magnitudes in the range are usually nonadjustable anchor points.

Multiplicative noise Internal noise that is proportional to stimulus magnitude.

Multiplicative summation Term for when signals from two or more stimuli are multiplied for detection. Multiplicative summation is a form of AND-gating, in which detection only occurs if more than one signal is present.

Nanometer Unit of light wavelength λ , usually abbreviated to nm (10^{-9} m).

Noise distribution Distribution of the relative probabilities of noise samples of different magnitude.

Normal distribution

$$\phi(x) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right]$$

where μ is the mean of the distribution and σ the standard deviation. Also known as the Gaussian distribution.

Nulling procedure The procedure whereby a stimulus whose percept has been altered along some stimulus dimension is returned to its original perceptual state via a change to some other stimulus dimension.

Odd-man-out task See Oddity task.

Oddity task Forced-choice task in which the observer is presented with a number of stimuli, all but one of which are the same, and chooses the stimulus that is different. The minimum number of stimuli is 3.

One up/three down Adaptive (or staircase) method that targets 79.4% correct responses. Stimulus magnitude is increased after each incorrect response and decreased after three consecutive correct responses.

One up/two down Adaptive (or staircase) method that targets 70.71% correct responses. Stimulus magnitude is increased after each incorrect response and decreased after two consecutive correct responses.

Ordinal scale Perceptual scale in which stimuli are rank-ordered according to perceived magnitude.

Paired comparisons See Method of paired comparisons.

Partition scaling Method for deriving a perceptual scale that involves observers adjusting a stimulus to be perceptually midway between two fixed, termed anchor, stimuli.

Pedestal The baseline stimulus to which an increment or a decrement in stimulus magnitude is added.

Perceptual scale The function describing the relationship between the perceived and physical magnitudes of a stimulus dimension. Examples: perceived contrast as a function of contrast, perceived velocity as a function of velocity, and perceived depth as a function of retinal disparity.

Point of subjective alignment (PSA) The relative positions of two lines at which they appear aligned.

Point of subjective equality (PSE) The physical magnitude of a stimulus at which it appears perceptually equal in magnitude to that of another stimulus. An example is a stimulus with, say, a contrast of 0.5 that appears to have the same contrast as a larger stimulus with, say, a contrast of 0.4.

Posterior odds Reflects a researcher's beliefs regarding the relative probabilities of two alternative models of some data taking into account prior beliefs as well as empirical data.

$$\text{posterior odds} = \frac{p(M_1|y)}{p(M_2|y)} = \text{BF} \frac{p(M_1)}{p(M_2)}$$

where BF is the Bayes Factor and $\frac{p(M_1)}{p(M_2)}$ is the prior odds.

Posterior probability Reflects a researcher's beliefs regarding the truth of a hypothesis taking into account prior beliefs as well as empirical data. See also Bayes' Theorem.

Power function $F(x; a, n) = ax^n$.

Precision The inverse of the variability of a psychophysical measurement. The measure of variability may be the spread of the psychometric function or the standard deviation of a set of measurements.

Prior odds Reflects a researcher's beliefs regarding the relative probabilities of two alternative models of some data.

$$\text{prior odds} = \frac{p(M_1)}{p(M_2)}$$

where $p(M_i)$ reflects the researcher's prior belief in model M_i in terms of a probability.

Prior probability Reflects a researcher's beliefs regarding the truth of a hypothesis prior to the collection of empirical data. See also Bayes' Theorem.

Probability density function Function describing the relative probabilities of events. The function must be integrated to derive actual probabilities.

Probability summation Model of summation in which multiple stimuli in a detection task are detected by separate channels or mechanisms. The improvement in detection with multiple stimuli from probability summation is attributable to the increased chance that one of the stimuli will either exceed the threshold or produce the biggest signal.

Progressive solution (in partition scaling) Partition scaling method in which the observer first divides the perceptual distance between two anchor points into equal parts, then divides the two subjectively equal parts into four, then into eight, etc., until the required number of partitions has been reached.

Proportion correct The proportion of trials in which the observer makes a correct response.

Proportion false alarms The proportion of target-absent trials in which the observer responds that the target is present.

Proportion hits The proportion of target-present trials in which the observer responds that the target is present.

Psi method Adaptive method that optimizes the efficiency of estimation of the threshold as well as the slope parameter of a psychometric function. On each trial the stimulus magnitude is chosen that will lead to the lowest expected entropy across the posterior distribution defined across threshold and slope parameters.

Psychometric function (PF) A function that describes the relationship between probabilities of observer responses and stimulus magnitude. The general form of the psychometric function is:

$$\psi(x; \alpha, \beta, \gamma, \lambda) = \gamma + (1 - \gamma - \lambda)F(x; \alpha, \beta)$$

where $F(x; \alpha, \beta)$ is the function with parameter α determining the x value at which the function reaches some criterion value (e.g., 0.5) and β determines the slope of the function. Parameters γ and λ are the guess and lapse rates, respectively. Commonly used functions are the Logistic, Cumulative Normal, Weibull, Gumbel, and Hyperbolic Secant.

Pulsed-pedestal Procedure in which a pedestal stimulus and its increment (or decrement) are presented in synchrony.

Quadruples See Method of quadruples.

QUEST Adaptive method that can be considered to be a Bayesian version of the best PEST (*see also* best PEST). After each response, the posterior distribution across possible threshold parameter values is determined from the prior distribution, which reflects the experimenter's assumptions about the threshold, and the likelihood function based on all preceding trials. The threshold estimate with the highest posterior probability serves as the stimulus magnitude for the subsequent trial.

Quick function

$$F_Q(x; \alpha, \beta) = 1 - 2^{\left(-\left(\frac{x}{\alpha}\right)^{\beta}\right)},$$

where α determines the location (threshold) and β determines the slope of the function. The Quick function differs from the Weibull function only in the base of the exponent.

Quick pooling model Formula for predicting the improvement in sensitivity to multiple stimuli assuming probability summation under High-Threshold Theory:

$$S_{cmb} = \left[\sum_{i=1}^n S_i^\beta \right]^{1/\beta}$$

where S_{cmb} is sensitivity (typically the reciprocal of detection threshold) to the stimulus combination, S_i is sensitivity to the i th stimulus when presented alone, and n is the number of stimuli. β is the slope of the Weibull function fitted to the psychometric function of proportion correct as a function of stimulus intensity. The formula is a special case of Minkowski summation in which the exponent is Weibull β .

Ratio scale A perceptual scale in which the ratio of scale values corresponds to the ratios of perceived magnitudes of the corresponding stimuli. A ratio scale can be rescaled by aX without loss of information where a is an arbitrary constant.

Rayleigh match A traditional tool for studying color vision and diagnosing color deficiency. Defined as the relative intensities of a mixture of red (say 679 nm) and green (say 545 nm) light required to match a monochromatic yellow (590 nm) light.

Receiver operating characteristic The function that describes how the relative proportions of hits and false alarms changes with the observer's criterion.

Recognition Refers to experiments/tasks in which the observer names a stimulus from memory or selects from a set of stimuli a stimulus previously shown. The term is often used to characterize experiments involving relatively complex stimuli such as faces, animals, household objects, etc.

Reflectance The proportion of incident light that is reflected by an object.

Reliability The reproducibility of a psychophysical measurement.

Response bias See Bias (of observer).

Retinal disparity The horizontal or vertical difference between the angle subtended by an object to each eye with respect to fixation.

Root mean square (RMS) contrast Defined as $\text{RMS} = \sqrt{\frac{1}{n} \sum_{i=1}^n \frac{(x_i - \bar{x})^2}{\bar{x}}} = \frac{\text{SD}_x}{\bar{x}}$, where n is the number of pixels in the image, x_i is the luminance value of pixel, i is the mean luminance, and SD_x is the standard deviation of luminance values x . Contrast measure of choice for complex images.

Same-different task Task in which the observer decides whether a pair of stimuli are the same or different. In the 1AFC version, one of the pairs (Same or Different) is presented on a trial and the observer responds "same" or "different." In the 2AFC version, both pairs (Same and Different) are presented on a trial, and the observer responds "first" or "second," depending on the alternative/interval perceived to contain the Same (or the Different) pair.

Sampling distribution Probability density function of a statistic (e.g., parameter estimate). May be approximated by repeated estimation based on samples from an assumed population.

Sampling error The difference between a parameter estimate and the parameter's true value.

Saturated model A model that makes no assumptions other than the assumptions of stability and independence. As such, a saturated model contains a parameter for each unique stimulus condition. A model comparison that compares a more restricted model to the saturated model is known as a goodness-of-fit test.

Scalar A single number, e.g., 8, 1.5, 1.4e-5.

Sensory scale See Perceptual scale.

Signal Detection Theory A theory of how observers make perceptual decisions based on the premise that the internal representation of a stimulus is a sampling distribution, typically a Gaussian distribution, with a mean and a variance that determines the discriminability of the stimulus from other stimuli.

Signal distribution Distribution of the relative probabilities of signal samples of various magnitude.

Simultaneous brightness contrast The phenomenon in which the brightness of a stimulus depends reciprocally on the luminance of its surround.

Simultaneous solution (in partition scaling) See Multipartition scaling.

Sine-wave pattern A pattern in which the stimulus dimension is modulated in space or time according to a sinusoidal function:

$$F(x; m, a, f, \rho) = m + a \sin(2\pi f x + \rho)$$

where m is the mean stimulus magnitude, a the amplitude of modulation, f the frequency of modulation (in cycles per unit space or time), and ρ the phase of modulation (in radians; one full cycle equals 2π radians). The inclusion of 2π in the equation means that a full cycle of modulation will be completed in the interval $0 \leq x \leq f^{-1}$.

Slope (of psychometric function) Rate of change of response as a function of stimulus magnitude.

One of the four parameters that characterize a PF (β). Note, however, that whereas β is often referred to as the slope of the PF, it generally will not correspond in value to the slope of the function as defined in calculus (i.e., the first derivative of the function).

Spatial frequency (SF) The number of cycles of modulation of a stimulus dimension per unit visual angle. Typically measured as cycles per degree (cpd).

Spread (of psychometric function) Also known as support of psychometric function. Stimulus range within which a PF goes from $\gamma + \delta$ to $1 - \lambda - \delta$, where γ is the lower and $1 - \lambda$ the upper asymptote of the PF. δ is an arbitrary constant ($0 < \delta < [1 - \lambda - \gamma]/2$). Thus, if we let σ symbolize spread:

$$\sigma = \psi^{-1}(1 - \lambda - \gamma; \alpha, \beta, \gamma, \lambda) - \psi^{-1}(\gamma + \delta; \alpha, \beta, \gamma, \lambda)$$

where $\psi^{-1}(y; \alpha, \beta, \gamma, \lambda)$ is the inverse of the psychometric function $\psi(x; \alpha, \beta, \gamma, \lambda)$.

Stability, assumption of The assumption that the performance of an observer (for example, the probability of a correct response as a function of stimulus intensity x) does not change during the course of the experiment.

Staircase methods See Adaptive methods.

Standard deviation Square root of the variance (see also Variance). A measure of the variability among scores.

Standard error The standard deviation of a parameter's sampling distribution. Used to quantify the reliability of a parameter estimate.

Standard(ized) normal distribution The normal distribution with mean equal to 0 and standard deviation equal to 1.

$$\phi(z) = \frac{1}{\sqrt{2\pi}} \exp\left(\frac{-z^2}{2}\right)$$

Steady-pedestal Procedure in which the pedestal stimulus is presented alone before the addition of the increment or decrement.

Stereopsis The means by which the relative depth of an object is determined by virtue of the fact that the two eyes view the object from a slightly different angle.

Stimulus exposure duration The length of time a stimulus is exposed during a trial.

Stimulus-onset-asynchrony (SOA) The temporal interval between the onset of two stimuli.

Stimulus scaling factor The factor g by which stimulus amplitude or intensity is scaled in order to be converted to the Signal Detection Theory measure d' .

Summation ratio The summation ratio (SR) is a measure of the improvement in sensitivity for detecting combined (i.e., multiple) compared to single stimuli. If T_{cmb} and T are the thresholds for detecting the combined and single stimuli, $SR = T/T_{cmb}$. SR can also be expressed in decibels (dBs), i.e.,

$$20 \log_{10}\left(\frac{T}{T_{cmb}}\right)$$

Summation square The pattern of thresholds obtained for the detection of two stimuli presented in various intensity or amplitude ratios. The summation square plots thresholds as points in a two-dimensional graph with the X and Y axes corresponding to the intensities of the two stimuli.

Support (of psychometric function) See Spread of psychometric function.

Symmetric (form of 1AFC) Type of single alternative forced-choice task/procedure in which the two discriminands can be considered to be mirror opposites, for example grating patches that are left- and right-oblique.

Temporal frequency (TF) The number of cycles of modulation of a stimulus dimension per unit time. Typically measured as cycles per second (cps).

Termination criterion In adaptive methods, the rule that is used to terminate a staircase. For example, a staircase may be terminated after a set number of trials or a set number of reversals.

Threshold In general refers to the difference in magnitude between two stimuli or stimulus states that enables them to be just discriminable. Examples: a contrast detection threshold, a contrast discrimination threshold, or the threshold for binocular rivalry.

Threshold-versus-contrast (TvC) The function relating the threshold for detecting an increment (or decrement) in contrast as a function of the pedestal (or baseline) contrast.

Threshold-versus-intensity (Tvi) The function relating the threshold for detecting an increment (or decrement) in intensity (or luminance) as a function of the pedestal (or baseline) intensity.

Thurstonian scaling Method for deriving an interval perceptual scale using the method of paired comparisons, in which the scale is derived from the proportions of times that each stimulus magnitude is perceived to be greater than each of the other stimulus magnitudes.

Transducer function Function relating the physical intensity or amplitude of a stimulus to its internal or perceptual response. *See also* Perceptual scale.

Transformed likelihood ratio (TLR) Statistic used to determine whether two models differ significantly.

When one of the two models is nested under the other, TLR is asymptotically distributed as χ^2 with degrees of freedom equal to the difference in the number of free parameters between the two models.

When the fuller model is the saturated model, the transformed likelihood ratio is known as deviance.

Triads *See* Method of triads.

Triangular method Alternative name for a 3AFC oddity task.

Two-alternative forced-choice (2AFC) Here defined as any procedure in which the observer selects a stimulus from two alternatives. Examples: selecting the left oblique grating from a left- and a right-oblique grating pair or choosing from two alternatives a stimulus previously shown.

Two-interval forced-choice (2IFC) Procedure in which the observer selects a stimulus from two stimuli presented in a temporal sequence.

Type 1 experiment A psychophysical experiment/procedure/task in which there is a correct and an incorrect response on each trial.

Type 2 experiment A psychophysical experiment/procedure/task in which there is no correct and incorrect response on each trial.

Variance For any set of numbers x_i , the variance is given as:

$$\sigma^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n}$$

where \bar{x} is the mean of x , and n is the number of scores. If the numbers x_i are a random sample drawn from a population, the following expression is that of an unbiased estimate of the variance of the population from which the x s were drawn.

$$\hat{\sigma}^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}$$

Vector An $m \times 1$ or $1 \times n$ array of numbers.

Vernier acuity The smallest misalignment of two stimuli that can be reliably detected.

Vernier alignment Experiment/task aimed at measuring the threshold (or precision) for detecting that two stimuli are misaligned and/or measuring the physical separation at which the two stimuli are perceived to be aligned, i.e., the bias.

Visual acuity Measure of the acuteness or clearness of vision. Traditionally measured using an eye chart.

Visual angle The angle subtended by a stimulus to the eye. Usually measured in arc degrees, arc minutes, or arc seconds.

Weber contrast Defined as $\Delta L/L_b$ where ΔL is the difference between the luminance of the stimulus and its background, and L_b the luminance of the background. Weber contrast is normally employed to measure the contrast of a uniform patch on a background and not normally used for periodic stimuli or noise patterns.

Weber's Law Law that states that the just discriminable difference in stimulus magnitude is proportional to stimulus magnitude.

Weibull function

$$F_W(x; \alpha, \beta) = 1 - \exp\left(-\left(\frac{x}{\alpha}\right)^\beta\right)$$

where α determines the location (threshold) and β determines the slope of the function.

Winner-take-all summation A term sometimes used to describe the situation in which there is no improvement in detection with multiple as compared to single stimuli.

Yes/No Experiment/task in which a single stimulus is presented on each trial, and the observer is required to indicate whether or not it contains the target.

z-score A score that corresponds to the number of standard deviations a score is above (if positive) or below (if negative) the mean. The z-scores corresponding to any distribution of scores will have a mean equal to 0 and a standard deviation equal to 1. The z-scores corresponding to any normally distributed variable will be distributed as the standard normal distribution.

List of Acronyms

1AFC	One-alternative-forced-choice
1IFC	One-interval-forced-choice
2AFC	Two-alternative-forced-choice
2IFC	Two-interval-forced-choice
3AFC	Three-alternative-forced-choice
4AFC	Four-alternative-forced-choice
AFC	Alternative-forced-choice
AIC	Akaike's information criterion
APE	Adaptive probit estimation
APL	Asymptotic Performance Level
AS	Additive summation
BIC	Bayesian information criterion
CPD	Cycles per degree
CPS	Cycles per second
CRT	Cathode ray tube
dB	Decibels
HTT	High Threshold Theory
IFC	Interval-forced-choice
ISI	Inter-stimulus-interval
ITI	Inter-trial-interval
JND	Just-noticeable-difference
LL	Log likelihood
LR	Likelihood ratio
M-AFC	<i>M</i> -alternative-forced-choice
MDS	Multi-dimensional scaling
ML	Maximum likelihood
MLDS	Maximum likelihood difference scaling
PEST	Parameter Estimation by Sequential Testing
PF	Psychometric function
PS	Probability summation
PSA	Point-of-subjective-alignment
PSE	Point-of-subjective-equality
ROC	Receiver operating characteristic
RT	Reaction time
SD	Standard deviation
SDT	Signal Detection Theory
SE	Standard error
SF	Spatial frequency

SOA	Stimulus-onset-asynchrony
SR	Summation ratio
TF	Temporal frequency
TLR	Transformed likelihood ratio
T.v.C	Threshold versus contrast
T.v.I	Threshold versus intensity
T.v.n	Threshold versus number of stimuli

Index

Note: Page numbers followed by "f" and "b" indicate figures and boxes, respectively.

A

- Accuracies, non-threshold tasks, 45
- Acronyms, list, 323–324
- Adaptive procedures
 - overview, 53, 119–120
 - psi method
 - overview, 137–147
 - Palamedes, 142b
 - practical tips, 144–145
 - psi-marginal method, 137–141
 - termination criteria, 141
 - threshold estimate, 141
 - variations, 145–147
 - running fit methods
 - best PEST, 131–132
 - Palamedes, 134b
 - practical tips, 133–137
 - Quest, 132–133
 - termination criteria, 133
 - threshold estimate, 133
 - up/down method
 - Palamedes, 125b, 133
 - practical tips, 129–131
 - principles, 120–122
 - termination criteria, 124
 - threshold estimate, 124
 - transformed and weighted up/down method, 123–124
 - transformed up/down method, 122
 - weighted up/down method, 122–123
- Additive summation, 221–222
- High-Threshold Theory (HTT)
 - multiple stimuli compared to one, 222
- Signal Detection Theory (SDT)
 - equations for additive summation, 196–197
 - expressing summation using the Minkowski formula, 201–203
 - multiple stimuli compared to one, 198–201
- Adjustment, nonforced-choice matching with two stimuli per trial, 46–48

- AFC. *See* Alternative forced-choice
- AIC. *See* Akaike's information criterion
- Akaike's information criterion (AIC), model comparisons, 302–304
- Alternative forced-choice (AFC)
 - calculations
 - criterion C
 - 1AFC, 176–177
 - biased 2AFC, 180
 - criterion C' for 1AFC, 177
 - criterion $\ln\beta$
 - 1AFC, 177
 - biased 2AFC, 180
 - d'
 - 1AFC, 175–176
 - 1AFC same-different, 182–184
 - 2AFC match-to-sample, 185
 - 2AFC same-different, 180–182
 - biased 2AFC, 178–180
 - M-AFC, 172–174
 - M-AFC oddity task, 185–187
 - unbiased 2AFC, 178
 - $P_{c_{\max}}$
 - 1AFC, 177–178
 - biased 2AFC, 180
 - Z-score relationship with probability, 171–172
- d' measurement
 - 1AFC tasks
 - d' from pH and pF , 154–157
 - demonstration programs, 157b, 159b
 - rating scale experiment, 158–160
 - same–different tasks, 183–184
 - 2AFC tasks with observer bias, 160–161
 - comparing with P_c s across difference tasks, 166
 - conversion from P_c for unbiased M-AFC tasks, 153–154
- match-to-sample tasks
 - 2AFC, 164
 - M-AFC, 164

- Alternative forced-choice (AFC) (*Continued*)
 oddity tasks, 165
 rationale, 151–153
 same-different tasks
 1AFC, 162–163
 2AFC, 162–163
 overview, 162–163
 definition, 26
 M-AFC tasks, 44–45
 Palamedes exercises, 153–170
 $P_{c_{\max}}$ estimation with observer bias, 165–166
 Appearance, versus performance, 20–24
 Appearance-based tasks
 matching
 forced-choice matching with two stimuli per trial, 46
 nonforced-choice matching with two stimuli per trial
 adjustment, 46
 nulling, 46–48
 overview, 45–48
 scaling
 forced-choice scaling procedures
 four stimuli per trial, 50
 greater than four stimuli per trial, 51
 multidimensional scaling, 51
 three stimuli per trial, 50
 two stimuli per trial, 50
 nonforced-choice scaling
 magnitude estimation, 51
 multi-partition scaling, 52
 partition scaling, 51–52
 perceptual scale types, 48
- B**
 Bayes Factor, model comparisons, 304–305
 Bayesian criterion, psychometric function fitting
 Bayes' theorem, 106–108
 error estimation, 111–112
 prior distribution, 108–111, 109f
 Bayesian information criterion (BIC), model comparisons, 304
 Best PEST
 Palamedes, 134b
 practical tips, 133–137
 termination criteria, 133
 threshold estimate, 133
 BIC. *See* Bayesian information criterion
- C**
 Class A observations, 14–19
 Class B observations, 14–19
 Coin tossing exercise, 218–219
 Contrast detection threshold, measurement, 12–13
- Criterion C, calculations
 1AFC, 176–177
 biased 2AFC, 180
 Criterion C', calculation for 1AFC, 177
 Criterion-free, versus criterion-dependent, 27–28
 Criterion $\ln\beta$, calculations
 1AFC, 177
 biased 2AFC, 180
 Cumulative Normal distribution, psychometric function, 79
- D**
 d' . *See* Signal Detection Theory
 Detection, versus discrimination, 29–31
 Dipper function, 229–230
 Discrimination, versus detection, 29–31
 Discrimination scale
 dipper function, 229–230
 Fechner's integration of Weber's Law, 228–229
 limitations, 231–232
 overview, 227–232
- F**
 Fechner, Gustav Theodor, 1
 Fechner's integration of Weber's Law, 228–229
 Forced-choice tasks
 appearance-based task procedures. *See* Matching:
 Scaling
 denotations
 response number, 26
 stimuli number, 26
 matching with two stimuli per trial, 46
 proportion correct in, 219–220
 threshold procedures
 four stimuli per trial, 44–45
 M-AFC tasks, 44–45
 one stimulus per trial
 method of limits, 40
 symmetric discriminands, 42
 yes/no procedure, 40–42
 overview, 39–45
 three stimuli per trial
 oddity task, 43–44
 two-alternative forced-choice
 match-to-sample, 44
 two stimuli per trial
 one-alternative forced-choice same-different task, 43
 two-alternative forced-choice task, 42–43
 two-interval forced-choice task, 42–43
 versus nonforced-choice tasks, 24–27
 Fourth root summation, 208
 Function. *See* Psychometric function

G

- Goodness-of-fit
 likelihood ratio test *vs.* Pearson's chi-square test, 269b
 model comparisons, 264–268
- Guessing rate
 definition, 61
 error estimation, 65–69
 psychometric function fitting, 61–64
- Gumbel function, 80

H

- High-Threshold Theory
 psychometric function, 73–76
 summation model, 218–222
 additive summation, 221–222
 multiple stimuli compared to one, 222
 probability summation, 218–222
 coin tossing exercise, 218–219
 multiple stimuli compared to one, 221
 proportion correct in forced-choice tasks, 219–220
 Quick pooling formula, 221–222
 summation psychometric functions, 220–221
- Hyperbolic Secant function, 84

I

- Identification, definition, 31
- IFC. *See* Interval forced-choice
- Inference. *See* Statistical inference
- Interval forced-choice (IFC), definition, 26
- Inverse function, 65

J

- Just-noticeable difference (JND)
 measurement, 30
 perceptual scales and internal noise, 153–170

L

- Lapse rate
 error estimation, 65–69
 issues of, 91b
 psychometric function fitting, 63–64
- Likelihood ratio test, 275
- Logistic function, 80
- Log-Quick function, 84

M

- M-AFC. *See* Alternative forced-choice
- Magnitude estimation, Class B observation, 18

Matching

- appearance-based tasks
 forced-choice matching with two stimuli
 per trial, 46
 overview, 45–48
- nonforced-choice matching with two
 stimuli per trial
 adjustment, 46
 nulling, 46–48

Match-to-sample task

- d'* calculation
 2AFC match-to-sample, 185
d' measurement
 2AFC, 164
 M-AFC, 164
 two-alternative forced-choice, 44

Maximum likelihood criterion, psychometric function fitting

- error estimation, 90–106
 example, 85–87
 likelihood function, 87–90
 overview, 64–65
 in Palamedes, 66b
 procedure, 94b

Maximum Likelihood Difference Scaling (MLDS)

- and internal noise, 243–244
 method of quadruples, 232–233
 overview, 232–236

Palamedes

- data fitting, 237b
 demonstration program, 237b
 observer response simulation, 237b
 plotting, 237b
 stimulus set generation, 237b
 and paired comparisons, 243
 partition scaling, 244–245

MDS. *See* Multi-dimensional scaling

- Method of constant stimuli, 52–53
- MLDS. *See* Maximum Likelihood Difference Scaling

Model comparisons. *See* Statistical inference

- Muller–Lyer illusion
 Class B observation, 17f, 18
 objective versus subjective judgment, 28
- Multi-dimensional scaling (MDS), forced-choice scaling, 51

N

- Nonforced-choice tasks
 appearance-based task procedures. *See* Matching; Scaling
 versus forced-choice tasks, 24–27
- Nulling, 46–48

O

- Objective, versus subjective, 28–29
 Observations, Class A versus Class B, 14–19
Oddity task
 d' calculation for M – AFC oddity task, 185–187
 d' measurement, 165
 three stimuli per trial, 43–44
One-alternative forced-choice task. *See* Alternative forced-choice task

P

- Palamedes
 acronyms, 8b
 adaptive procedures
 ψ method, 142b
 running fit methods, 134b
 up/down method, 125b, 133
 basic summation computation, 197b
 demonstration programs, 5b
 error estimation, 65–69, 111–112
 error messages, 5b
 fitting psychometric functions, 60b, 63–64, 66b, 84–112, 112b
 functions, 5b
 goodness-of-fit estimation, 69–71
Maximum Likelihood Difference Scaling
 data fitting, 237b
 demonstration program, 237b
 observer response simulation, 237b
 plotting, 237b
 stimulus set generation, 237b
 model comparisons
 failed fits, 295–296
 more than two conditions, 268–275
 pairwise comparisons, 288–289
 slope effects, 263–264
 threshold effects, 262–263
 trend analysis, 284–288
 multiple-fit summation, 213b
 organization, 5b
 overview, 3b
 psychometric function types
 Cumulative Normal distribution, 79
 Gumbel function, 80
 Hyperbolic Secant function, 84
 Logistic function, 80
 log-Quick function, 84
 Quick function, 83
 Weibull function, 80
 Signal Detection Theory exercises, 153–170, 154b
 spread of psychometric functions, 84

- Partition scaling. *See* Scaling
 P_c . *See* Signal Detection Theory
 Pearson's chi-square test, 269b
 Perceptual scales. *See* Scaling
 Performance, versus appearance, 20–24
 PF . *See* Psychometric function
 Point of selective alignment (PSA), 22–23
 Point of subjective equality (PSE), 16–20, 22, 45
 Probability summation, 218–222
High-Threshold Theory (HTT)
 coin tossing exercise, 218–219
 multiple stimuli compared to one, 221
 proportion correct in forced-choice tasks, 219–220
 Quick pooling formula, 221–222
 summation psychometric functions, 220–221
Signal Detection Theory
 applying the PS_{SDT} functions, 207–208
 equations for probability summation, 203–207
 equal stimulus intensities, 204b
 unequal stimulus intensities, 204b
 multiple stimuli compared to one, 208
PSA. *See* Point of selective alignment
PSE. *See* Point of subjective equality
Psi method
 overview, 137–147
 Palamedes, 142b
 practical tips, 144–145
 psi-marginal method, 137–141
 termination criteria, 141
 threshold estimate, 141
 variations, 145–147
Psychometric function (PF)
 choice of function, 64
 error estimation, 65–69
 fitting
 Bayesian criterion
 Bayes' theorem, 106–108
 error estimation, 111–112
 prior distribution, 108–111, 109f
 goodness-of-fit estimation, 69–71
 maximum likelihood criterion
 error estimation, 90–106
 example, 85–87
 likelihood function, 87–90
 overview, 64–65
 procedure, 94b
 methods for, 64–65
 software, 3b
 inverse functions, 84
 model comparisons. *See* Statistical inference
 modeling with Signal Detection Theory, 166–170

- number of trials, 57
 overview, 56–57
 in Palamedes
 evaluation, 60b
 of standard errors, using bootstrap, 70b
 fitting, 266b
 goodness-of-fit determination, 72b
 maximum likelihood criterion, 66b
 spread, 84
 stimulus levels
 linear versus logarithmic spacing, 58–59
 range, 57–58
 summation functions, 220–221
 theory, 71–79
 High-Threshold Theory, 73–76
 Signal Detection Theory, 76–79
 types
 Cumulative Normal distribution, 79
 Gumbel function, 80
 Hyperbolic Secant function, 84
 Logistic function, 80
 log-Quick function, 84
 overview, 56–57
 Quick function, 83
 Weibull function, 80, 81b
- P**
Psychophysics
 definition, 1
 experiment classification schemes, 32–33, 38f
- Q**
Quest
 Palamedes, 134b
 practical tips, 133–137
 principles, 132–133
 termination criteria, 133
 threshold estimate, 133
Quick function, 83, 221–222
- R**
Rayleigh match, 15–16, 15f
Reaction time, performance-based non-threshold tasks, 45
Receiver operating characteristic (ROC), relationship between *pH* and *pF*, 156–157
Recognition, definition, 30–31
ROC. *See* Receiver operating characteristic
- S**
Same-different task
 d'
 calculation
 1AFC same-different, 182–184
 2AFC same-different, 180–182
- measurement
 1AFC, 162–163
 2AFC, 162–163
 overview, 162–163
two-alternative forced-choice, 44
two-interval forced-choice task, 44
- Scaling**
 appearance-based tasks
 forced-choice scaling procedures
 four stimuli per trial, 50
 greater than four stimuli per trial, 51
 multidimensional scaling, 51
 three stimuli per trial, 50
 two stimuli per trial, 50
 nonforced-choice scaling
 magnitude estimation, 51
 multi-partition scaling, 52
 partition scaling, 51–52
 perceptual scale types, 48
 discrimination scale
 dipper function, 229–230
 Fechner's integration of Weber's Law, 228–229
 limitations, 231–232
 overview, 227–232
Maximum Likelihood Difference Scaling
 method of quadruples, 232–233
 overview, 232–236
- Palamedes**
 data fitting, 237b
 demonstration program, 237b
 observer response simulation, 237b
 plotting, 237b
 stimulus set generation, 237b
 partition scaling, 244–245
 perceptual scales and internal noise, 243–244
 perceptual scale principles, 225–227
- SDT**. *See* Signal Detection Theory
Shannon entropy, psi method, 138–139
Signal Detection Theory (SDT)
 calculations
 criterion *C*
 1AFC, 176–177
 biased 2AFC, 180
 criterion *C'* for 1AFC, 177
 criterion *In* β
 1AFC, 177
 biased 2AFC, 180
- d'
 1AFC, 175–176
 1AFC same-different, 182–184
 2AFC match-to-sample, 185
 2AFC same-different, 180–182

- Signal Detection Theory (SDT) (*Continued*)
 biased 2AFC, 178–180
M-AFC, 172–174
M-AFC oddity task, 185–187
 unbiased 2AFC, 178
- $P_{c_{\max}}$
 1AFC, 177–178
 2AFC, 180
- Z-score relationship with probability, 171–172
- d' measurement
 1AFC tasks
 d' from pH and pF , 154–157
 demonstration programs, 157b, 159b
 rating scale experiment, 158–160
 same-different task, 183–184
- 2AFC, 164
- 2AFC tasks with observer bias, 160–161
- comparing with P_c s across difference tasks, 166
- conversion from P_c for unbiased *M*-AFC tasks,
 153–154
- M*-AFC, 165
- match-to-sample tasks oddity tasks, 165
- rationale, 151–153
- same-different tasks
 1AFC, 162–163
 2AFC, 162–163
 overview, 162–163
- modeling with, 166–170
- overview, 76–79, 150
- Palamedes exercises, 153–170, 154b
- $P_{c_{\max}}$ estimation with observer bias, 165–166
- summation model, 194–217
 additive summation, 195–203
 equations for additive summation, 196–197
 expressing summation using the Minkowski
 formula, 201–203
 multiple stimuli compared to one, 198–201
- modeling summation with simulated
 psychometric functions, 209–211
- preliminaries, 194–195
- probability summation, 203–208
 applying the PS_{SDT} functions, 207–208
 equations for probability summation, 203–207
 multiple stimuli compared to one, 208
- summation squares simulation, 211–212
- working with psychometric function data,
 212–217
- terminology, 150–151
- Spread, psychometric functions, 84
- Standard error, eyeballing, 249–252
- Standard error of estimation (SE)
 Bayesian criterion calculation, 111–112
 maximum likelihood calculation, 103–105
- Statistical inference
 failed fits, 295–296
 model comparisons
 Akaike's information criterion, 302–304
 Bayes Factor, 304–305
 Bayesian information criterion, 304
 goodness-of-fit, 264–268
 likelihood ratio test, 275
 more than two conditions, 268–275
 slope effects, 263–264
 threshold effects, 262–263
 underlying logic, 252–262
- overview, 247–249
- pairwise comparisons, 288–289
- standard error eyeballing, 249–252
- transformed likelihood ratio (TLR), 269b
- trend analysis, 284–288
- Stimulus presentation, timing, 53–54
- Subjective, versus objective, 28–29
- Summation measures
 frameworks, 190–194
- High-Threshold Theory, 212–217
 additive summation, 221–222
 probability summation, 218–222
- overview, 189–194
- scenarios, 190–194
- Signal Detection Theory, 194–217
 additive summation, 195–203
 modeling summation with simulated
 psychometric functions, 209–211
- preliminaries, 194–195
- probability summation, 203–208
- summation squares simulation, 211–212
- working with psychometric function data,
 212–217
- summation ratios, 202b
- types, 190–194
- Suprathreshold, versus threshold, 31
- T**
- Threshold
 adaptive procedures and estimates
 up/down method, 124
 psi method, 141
 running methods, 133
 error estimation, 65–69
 forced-choice threshold procedures
 four stimuli per trial, 44
 M-AFC tasks, 44–45
 one stimulus per trial
 method of limits, 40
 symmetric discriminands, 42
 yes/no procedure, 40–42

- overview, 39–45
three stimuli per trial
 oddity task, 43–44
 two-alternative forced-choice
 match-to-sample, 44
two stimuli per trial
 one-alternative forced-choice same–different
 task, 43
 two-alternative forced-choice task, 42–43
 two-interval forced-choice task, 42–43
model comparisons, 262–263
nonforced-choice threshold procedures and method
 of adjustment, 45
psychometric function fitting, 56–57,
 62–64
versus suprathreshold, 31
TLR. *See* Transformed likelihood ratio
Transformed and weighted up/down method,
 123–124
Transformed likelihood ratio (TLR), model
 comparisons, 269b
Transformed up/down method, 122
Trend analysis, statistical inference,
 284–288
Two-alternative forced-choice
 match-to-sample, 44
 overview, 24–25
 same–different task, 44
two stimuli per trial, 42–43. *See also* Alternative
 forced-choice
Two-interval forced-choice task
 measurement, 12–14
 same–different task, 44
 two stimuli per trial, 42–43
Type 1 observation, 19–20
Type 2 observation, 19–20
- U**
- Up/down method
 Palamedes, 125b, 133
 practical tips, 40–42
 termination criteria, 124
 threshold estimate, 124
 transformed and weighted up/down method,
 123–124
 transformed up/down method, 122
 weighted up/down method, 122–123
- W**
- Weber’s Law, Fechner’s integration of, 228–229
Weibull function, 80
Weighted up/down method, 122–123
- Z**
- Z-score, relationship with probability,
 171–172