Katholieke Universiteit Leuven
Faculteit Psychologie en Pedagogische Wetenschappen
Onderzoekseenheid Psychologie

Extensions and applications of the diffusion model for two-choice response times

Promotor: Prof. Dr. F. Tuerlinckx

Proefschrift aangeboden tot het behalen van de graad van Doctor in de Psychologie

door

Joachim Vandekerckhove

April 2009

Cover art *"A Thousand Words"* created with `http://www.wordle.net/`.

Joachim Vandekerckhove, **Extensions and applications of the diffusion model for two-choice response times.** Dissertation submitted to obtain the degree of Doctor of Philosophy in Psychology, April 2009. Promoter: Prof. Dr. F. Tuerlinckx.

Two-choice response time data (2CRT) is one of the most common formats of empirical data in experimental psychology. Unfortunately, such data do not adhere to the requirements of standard statistical models (such as the general linear model). The main goal of this thesis is to develop, extend, and apply methods for the analysis of 2CRTs on the basis of diffusion process models.

The diffusion process is a generalization of a standard random walk to continuous time and with a continuous state space. In our applications, we will always consider one-dimensional diffusions—a mathematical formalism to describe continuous changes in a scalar value over time. The central dogma of the diffusion model framework is that this fluctuating number represents an abstract 'evidence counter'. It is further assumed that the decider executes a response as soon as one of two boundaries is crossed, that the diffusion process may drift towards one of these boundaries at a lesser or greater rate, and that the process may be biased to start at a value near or far from a certain boundary. The challenge in diffusion modeling then lies in recovering the numerical values of these boundaries, the drift rate, and the bias, given only the times at which each boundary was hit. This challenge, and variations on the theme, are the focus of this thesis.

In the ***Introduction***, we give a brief introduction to the general problem of analyzing two-choice response times, and a bird's-eye overview of the five chapters of the dissertation.

In ***Chapter 1***, we describe a general method for fitting a diffusion model to empirical data. This method extends existing methods with a flexible way to constrain parameters across experimental conditions. Using design matrices as a constraining framework, this Chapter also discusses issues of statistical inference as applied to the design matrix method for diffusion models. Additionally, strategies are presented for handling outliers and contaminants—observed data points that are not generated by the decision process of interest. We demonstrate this collection of methods with several real examples.

In ***Chapter 2***, we introduce and describe the *Diffusion Model Analysis Toolbox*, a MATLAB toolbox that accompanies the design matrix method outlined in Chapter 1.

Until this point, we had only considered classical, 'frequentist' methods for statistical inference. In ***Chapter 3***, we move to the more general Bayesian statistical framework and demonstrate that this framework allows for more flexibility in modeling. We also experienced fewer numerical problems using Bayesian estimation methods.

The novel Bayesian methods proved most useful in extending the diffusion model into a hierarchical framework, which we describe in ***Chapter 4***. The hierarchical Bayesian diffusion model allows for the inclusion of random effects—something which would be technically possible, but highly impractical in a frequentist framework. The inclusion of random effects permits us to pool data across stimuli or participants that otherwise share nothing beyond being random draws from a common superpopulation. The random effects concept allows for more robust estimation, and it has the added virtue of being an accurate representation of the sampling scheme used in many empirical studies. The possibility of accounting for individual differences inside a population while retaining a conceptually interesting process model as the measurement level makes the hierarchical diffusion model an instance of *cognitive psychometrics*.

Finally, in ***Chapter 5***, we apply this novel method to a large data set, relating two-choice reaction times to semantic properties of the stimulus items. While the classical analysis, involving mainly general linear modeling, painted a heterogeneous and confusing picture, the hierarchical diffusion model approach succeeded in disentangling different sources of variability between items.

Joachim Vandekerckhove, **Uitbreidingen en toepassingen van het diffusiemodel voor binaire keuzereactietijden.** Proefschrift aangeboden tot het verkrijgen van de graad van Doctor in de Psychologie, April 2009. Promotor: Prof. Dr. F. Tuerlinckx.

Binaire keuzereactietijden komen erg vaak voor in de experimentele psychologie. Helaas voldoet dit soort gegevens niet aan de vereisten en assumpties van statistische standaardmodellen zoals het *general linear model*. Het hoofddoel van deze thesis is het ontwikkelen, uitbreiden, en toepassen van methoden voor de analyse van binaire keuzereactietijden op basis van diffusieprocesmodellen.

Het diffusieproces is een toevalsbeweging in continue tijd en in een continue ruimte. In onze toepassingen gaat het telkens om ééndimensionale diffusies—een wiskundig formalisme om continue veranderingen in een scalaire waarde over de tijd uit te drukken. Het centrale dogma van het diffusiemodelkader is dat dit variërende getal gekoppeld kan worden aan een abstracte 'evidentieteller'. Verder wordt aangenomen dat de beslisser een respons uitvoert zodra deze teller één van twee grenzen overschrijdt, dat het diffusieproces preferentieel naar één van deze grenzen kan wegdrijven en dat het startpunt van het proces dichter bij de ene grens dan de andere kan liggen. De uitdaging van diffusiemodelleren ligt erin de waarden van deze grenzen, van de drijfkracht en van de a priori voorkeur terug te vinden op basis van enkel de latentietijden waarop de grens werd geraakt. Deze uitdaging, en varianten ervan, zijn het onderwerp van deze thesis.

In de ***Introduction*** geven we een korte inleiding tot het algemene probleem met het analyseren van binaire keuzereactietijden, alsook een overzicht van de vijf hoofdstukken van de dissertatie.

In ***Chapter 1*** beschrijven we een algemene methode voor het passen van een diffusiemodel aan empirische gegevens. Deze methode breidt bestaande methoden uit met een flexibele manier om parameters constant te houden tussen experimentele condities. Met ontwerpmatrices als kader, bespreken we in dit Chapter ook statistische inferentie voor diffusiemodellen. Daarnaast stellen we enkele strategieën voor waarmee uitbijters en contaminanten—data die geobserveerd zijn maar die niet voortkomen uit het bestudeerde beslissingsproces—kunnen worden behandeld. We demonstreren deze methoden met enkele voorbeelden.

In ***Chapter 2*** wordt de *Diffusion Model Analysis Toolbox* voorgesteld: een MATLAB toolbox die toelaat de ontwerpmatrixmethode uit Chapter 1 toe te passen.

Tot hiertoe werden enkel de klassieke, 'frequentistische' methoden voor statistische inferentie gebruikt. In ***Chapter 3*** verplaatsen we ons naar het meer algemene Bayesiaanse kader en tonen hoe dit kader een grotere flexibiliteit toelaat in het modelleren. De Bayesiaanse schattingsmethoden leidden ook tot minder numerieke problemen.

De Bayesiaanse methoden maakten het ook eenvoudiger om het diffusiemodel in een hiërarchisch kader uit te breiden. Dit beschrijven we in ***Chapter 4***. Het hiërarchische Bayesiaanse diffusiemodel maakt het mogelijk om toevalseffecten toe te voegen. In een frequentistisch kader zou dit weliswaar mogelijk, maar zeer onpraktisch zijn. Met behulp van deze formulering met toevalseffecten kunnen we gegevens van verschillende condities of deelnemers samenbrengen—zelfs als die niets méér gemeen hebben dan hun lidmaatschap van een grotere groep of categorie. Het werken met toevalseffecten zorgt voor meer robuste schattingen en heeft als belangrijke meerwaarde dat het een getrouwe weergave is van de manier waarop deelnemers en stimuli in de praktijk geselecteerd worden. De combinatie van een interessant, interpreteerbaar procesmodel als meetniveau met het ondervangen van individuele verschillen maakt van het hiërarchische diffusiemodel een voorbeeld van *cognitieve psychometrie*.

In ***Chapter 5*** passen we deze nieuwe methode toe op een grote gegevensset waarin binaire keuzereactietijden worden gekoppeld aan semantische eigenschappen van de stimuli. Terwijl de klassieke analyse—op basis van lineaire modellen—een heterogeen en oninterpreteerbaar beeld schetste, konden we dankzij de toepassing van een hiërarchisch diffusiemodel verschillende bronnen van variabiliteit tussen items ontwarren.

Finalement, finalement
Il nous fallut bien du talent
Pour être vieux sans être adultes

Acknowledgements

The present dissertation is the fruit of collaborations and discussions with a host of admirable people, many of whom the author counts among his friends. Most prominent among these scientific collaborators are, in randomised order: Scott Brown, Eric-Jan Wagenmakers, Steven Verheyen, Andrew Heathcote, Michael Lee, Roger Ratcliff, Zita Oravecz, Francis Tuerlinckx, Gilles Dutilh, Ruud Wetzels, and Jeff Rouder.

A special mention is warranted of the entire Research Group for Quantitative Psychology and Individual Differences, where for three most enjoyable years I was able to feel free to engage in the projects that captured my interest. The unforgettable measure of joviality, warmth, and collegiality with which members of the group are received, together with a unique combination of professional and traditional values, have contributed to the research group's well-earned reputation.

Much of the work presented here would have been impossible without the logistic assistance provided by Emmy Bergen and Véronique Limbourg, technical assistance from Kristof Meers and the kind people at LUDIT (i.a., Wim Obbels, Samuel Xavier-de-Souza, Jo Vandeginste, and Herman Moons) and DICT (Johan Hendrickx, Jochen Janssens, Johan Van Rensbergen, and Noël Bovens), and material contributions from FWO (for travel grants to Irvine, CA, and Amsterdam), LUDIT (for the high-performance computing facilities), and Dell Inc. and Microsoft Corp. (for additional computing resources and software).

On the personal level, it is obvious that none of this would have been possible without the constant and unconditional support that my family is always ready to provide. With gratitude, I dedicate this thesis to these excellent people who have made me who I am.

Finally, particular thanks also go out to all those fantastic individuals

whose friendship I have been most fortunate to enjoy in recent years and, no doubt, in years to come. This is a shout-out to Yannick G., Hilde M., Nicolas A., Guy M., Zita O., Johannes K., Ellen G., Caroline S., and Joris G. Then there are those friends whom I think of as my posse, with whom I graduated and who remain close friends. Cheers to Wim M., Johan B., Tom W., Inneke K., Kenny B., Robbe G., Hannelore G., Ben S., and Tom T. Last but not least, where would I be without the companionship of (still in randomised order), Amber W., Laura O., Emma M., Julie A., Femke N., Marie I., Britt S., Charlotte E., Lisa R., Lotte I., Sarah N., Jolien D., Hanne E., Jana E., and Eline D. You all mean more to me than I ever seem to find the opportunity to tell you.

Joachim Vandekerckhove
Leuven, 1 April 2009

Contents

Introduction

The central topic in this dissertation is the analysis of reaction time (RT) data. RT data are omnipresent in psychological research, occurring in many different subdisciplines (e.g., reasoning, perception, learning, memory...). It is safe to say that RTs are among the most popular experimental measures in psychology. It is unfortunate in that regard that RTs are, from a statistical point of view, such a strange beast.

The goal of this brief introduction is to provide a layman's introduction to the background of our work. In the first section, we will explain why RTs are exceptional in a statistical sense. In the second section, we will introduce diffusion models, which are at the heart of our data-analytical strategy for dealing with two-choice response time (2CRT) data. After that, we will touch upon the subject of cognitive psychometrics (CP)—a fairly new[1] subdiscipline in cognitive science—and hope to infect the reader with some of our enthusiasm regarding future directions of CP. In a final section, we give a bird's-eye overview of the work presented in this

[1] And therefore, exciting.

dissertation.

In order to avoid cluttering the main text of this introduction with many references to the scientific corpus, we have relegated all references to the *Final Considerations* section.

The strangeness of reaction times

Formally strange

In the previous section, we remarked that RTs are exceptional in a statistical sense. To explain why this is so, let us look back at the most common class of statistical models: the linear model (LM). Imagine that two paired variables, X and Y, have been measured on n occasions, so that $(x_{(i)}, y_{(i)})$ is the i^{th} occasion. The well-known LM can then be used to predict values of Y (the criterion) on the basis of values of X (the predictor). At the core of the LM lie three assumptions:

Linearity: This is the most straightforward assumption: there is a roughly linear relationship between X and Y, so that the predicted values of Y adhere to the regularity $y^{\text{est}}_{(i)} = \mu_0 + x_{(i)}\mu_1,\ \mu_0, \mu_1 \in \mathbb{R}$. Of course, typically this prediction will not be perfect, and the prediction error $\varepsilon_{(i)} = y_{(i)} - y^{\text{est}}_{(i)}$ will not be exactly zero for all i.

Normality: The second assumption pertains to the distribution of the prediction errors $\varepsilon_{(i)}$, which is assumed to be normal with mean zero and some variance σ^2_ε: $\varepsilon_{(i)} \sim \mathrm{N}(0, \sigma^2_\varepsilon)$.

Homoscedasticity: The third assumption also relates to the distribution of prediction errors, but simply states that σ^2_ε is constant for all i. That is, it is the same for all conditions, or for all levels of X.

In the case of RTs, and of speeded RTs in particular, these assumptions are typically not met. RT distributions tend to be right-skewed, with a far greater range of high (supramedian) values than of low values, and 'extreme' RTs are almost always much larger than the average. 'Negative' outliers, on the other hand, are never much smaller than the mean and tend to have a 'hard threshold' (i.e., there is a non-zero lower bound in the RT distribution). Extreme positive prediction errors are therefore more likely than extreme negative ones, thereby violating the normality assumption. As a result, applying a LM like this one to RT data is inappropriate from a statistical point of view.

A common technique for dealing with the typical right skew of RT distributions is to take the logarithm of each RT before applying a LM. While this does alleviate the influence of positive outliers to an extent, the issue with the non-zero lower bound persists. Even more complex extensions into the *generalized lineal model* framework provide no solution for this lower bound issue. Gamma or exponential regression models, for example, are restricted to lower bounds that are exactly zero.

The issue becomes even more difficult when the RTs are paired with choice alternatives or accuracy information. Imagine an experiment with trials that vary in difficulty. Typically, difficult items will elicit more errors than easy items, and they will also require more time from the participant. Since both measures (RT and accuracy) simultaneously depend on this item difficulty, they cannot be considered independent from one another. This dependence between measures further complicates the application of classical statistical models such as multivariate linear models.

In addition to these statistical concerns, there are profound conceptual and substantive issues with applying LMs to (choice) RT data. We describe these issues in the next subsection.

Conceptually strange

In any modeling exercise, the ultimate goal is to extract from empirical data a handful of numbers (model parameters) that, together with the functional form of a model, summarize the data in a parsimonious but comprehensive way. A normal LM focuses chiefly on changes in the *mean* of the criterion—but the mean of a distribution of RTs may be influenced by many (all interesting) factors such as a participant's ability to make fast decisions, their level of caution (the so-called speed-accuracy trade-off), their state of readiness, and the amount of time it takes for them to process a stimulus and execute a motor response. As a result, observing a change in mean RT is highly ambiguous to the researcher, as it could indicate a change in any of these possible determinants (assuming that we are unaware of participants' abilities, caution, etc.). Hence, by focusing almost exclusively on the mean, the normal LM ignores many interesting aspects of the distribution.[2] Gamma or exponential regression models allow the variance to differ between conditions, but there it is directly linked to the mean. Ideally, our processing of the data would allow us to make statements about the psychologically interesting features of the *process* by which the data have come about. To accomplish this, we apply so-called process models.

In the next section, we will introduce one specific type of process model—the diffusion model—which will be at the center of the rest of the dissertation.

[2]Indeed, the author would go so far as to say that it fails to capture *any* of the interesting aspects of the data.

Diffusion models

Meet the family: Sequential sampling models

Constructing and applying a process model begins with enumerating the basic assumptions that we are willing to make about the process that has generated the data. For example, in the context of speeded binary choices, we might be willing to assume that the participant (or the observer or 'decider') accumulates small quanta of information, sequentially over time, from whatever stimulus he or she was exposed to. Suppose that a participant in an experiment is shown a visual stimulus and is asked to determine whether the stimulus belongs to category A or category B. Suppose further that most stimuli are at least somewhat ambiguous concerning category membership, so that the information they contain may favor A or B in mixed proportions. The decider then draws samples of information (or 'evidence') from this stimulus, one at a time, and aggregates this information with the information already accumulated. After each accumulation step, the decider evaluates whether the total amount of information in favor of A dominates the amount of information in favor of B, or vice versa. If so, then the process ends and a response is executed. This is the fundamental assumption—the "central dogma"—of sequential sampling models for choice response times.

The one-dimensional cousins

Within this family of models, there are many subdivisions according to further, more detailed assumptions one is willing to make. If the 'evidence counters' for A and B are independent, the model is called a *race model* (because the two counters race one another to a certain value). In the case of the diffusion model, evidence for A counts as evidence against

B, so that there is only one evidence counter. A response is executed if this counter reaches a certain high value (because that indicates a large amount of evidence for A) or a certain low value (because this indicates much evidence *against* A and hence *for* B).

Figure 1 shows an example illustration of the information accumulation process during a single trial in an experiment. The observer begins with an evidence counter at 0.4 (in arbitrary units). The evidence then accumulated is -0.08, $+0.33$, -0.05, -0.10, and -0.05, leading to the following values of the (total) evidence: [0.4 0.32 0.65 0.6 0.5 0.45]. The double-headed arrow on the left shows the changes in a single counter over time (with the time points labeled). Subfigure 'a' shows the same values in a more conventional way: with time shown on the horizontal axis. Subfigure 'b' shows a longer process with similar step sizes—this is a typical graphical presentation of a one-dimensional *random walk*: a process in which a single value changes over time with random steps.

Parameters of a one-dimensional random walk

Figure 2 shows three separate random walks, together with the upper and lower limits that indicate whether sufficient evidence has gathered in order to elicit a response. As soon as the process reaches one of these boundaries, the decision process terminates. The RT of the decider is determined by the number of steps the process took, and the response depends on which boundary was hit. If the distance to these boundaries is large, then it will take more steps to reach a boundary and the system will generally be slow to generate an output.

The three random walks shown in Figure 2 differ in an important way that is not clearly visible from the graphical presentation. Their *average* step sizes—the amount of information gathered at each time point—are

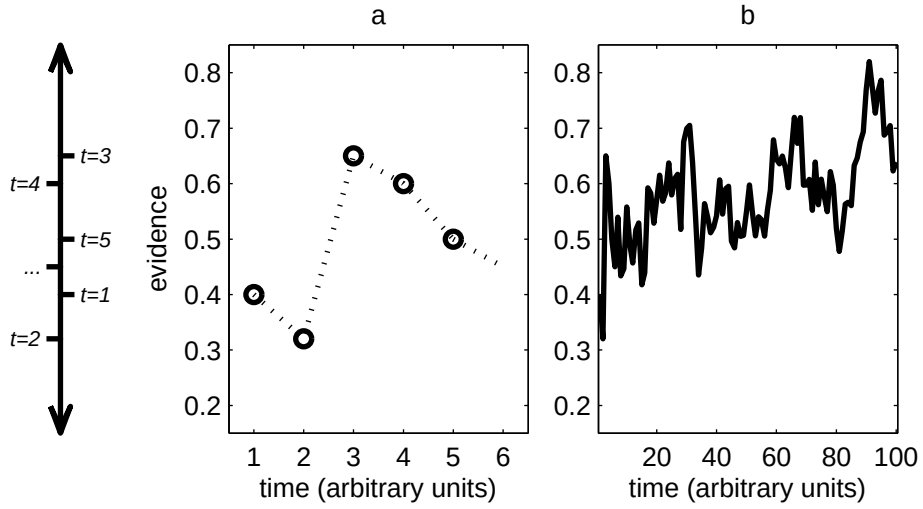


Figure 1: *An illustration of a random walk in one dimension. A counter changes value with discrete time steps. The double-headed arrow on the left emphasizes the unidimensionality: A counter takes a new value at each time point $t = 1$, $t = 2$, and so on. A random walk is more typically depicted as a graph over time, with time as the horizontal dimension. Subfigure 'a' shows the first steps of a random walk. Longer walks, as in Subfigure 'b', become irregular, jagged lines.*

positive (0.01), negative (-0.01), or zero. Because of this, one of the processes has a tendency to drift towards the upper boundary, another drifts to the lower boundary, and a third has no drift. The first process will hit the upper boundary with a higher probability, and will also do so after an expected number of steps that is smaller than the process with no drift. Similarly, the second process has a higher probability of hitting the lower boundary, and its average run time will also be shorter than that of the process without drift. Finally, the third random walk has equal chances of hitting either boundary, but will on average take more steps to arrive at any boundary. The average step size of a random walk process is called its *drift rate*.

The distance to the boundaries and the drift rate are the main para-

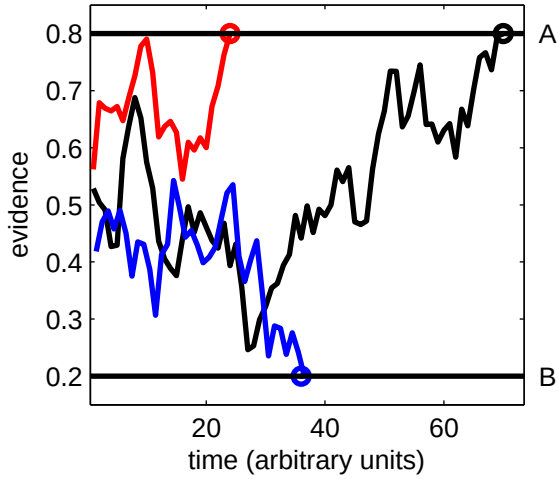


Figure 2: *Three random walks with absorbing boundaries. As soon as a process hits one of the boundaries, the process ends. In psychological terms, a decision has been made.*

Table 1: *The qualitative effects of the boundary separation and drift rate parameters of a one-dimensional random walk.*

Boundary separation	Drift rate	RT	Probability of 'A'
High	Positive	Medium	Very high ($\gg$50%)
High	Zero	Slow	50%
High	Negative	Medium	Very low ($\ll$50%)
Low	Positive	Fast	High (>50%)
Low	Zero	Medium	50%
Low	Negative	Fast	Low (<50%)

meters of the one-dimensional random walk. Importantly, the effects of changes in these parameters are distinct. Table 1 gives a quick qualitative summary of the combined effects of boundary separation and drift rate.

When we apply such a model to data, it makes sense to assume that the boundary separation is a property of the decider—that is, the system decides a priori how much information it needs in order to make a response. Similarly, the rate of information accumulation depends on the quality and

(lack of) ambiguity of the task or the stimulus. In psychological terms, the boundary separation is typically considered *person-specific*, while the drift rate is *stimulus-dependent*.

The Wiener diffusion model

The *Wiener diffusion model* is conceptually very similar to the one-dimensional random walk described above. In fact, the only conceptual difference is that for the Wiener process evidence is sampled in continuous time—so not in discrete steps. This makes no difference for the qualitative interpretation of the parameters—a Wiener process has a boundary separation and a drift rate just the same. Importantly, the continuous-time assumption means that we can express the expected distribution of RTs and choices with a closed-form formula (the likelihood function). We can use this likelihood function as a template to recover the approximate parameters of a Wiener process from only the RTs and associated choices.

The diffusion model for two-choice reaction times

While the Wiener diffusion process seems like a plausible model for choice RTs, we typically extend it with two more parameters in order to make it more suitable for real data. Firstly, we have so far assumed that the distance to the two boundaries is identical, making the choice process initially unbiased. However, we could forgo this assumption and instead allow one boundary to be closer to the starting point than the other, making the response associated with that boundary more relatively more likely. The parameter used to encode this bias is usually called the *starting point* or the *initial bias*—depending on which formal conventions we adopt to express it.

Secondly, the Wiener process (including the unbiased variety) models

Table 2: *The four main parameters of the Wiener diffusion model, with their substantive interpretations.*

Symbol	Parameter	Interpretation
α	Boundary separation	Speed-accuracy trade-off (high α means high accuracy)
β	Initial bias	Bias for either response ($\beta > 0.5$ means bias towards response 'A')
δ	Drift rate	Quality of the stimulus (close to 0 means ambiguous stimulus)
τ	Nondecision time	Motor response time, encoding time (high means slow encoding, execution)

a decision process. However, in experimental psychology, the RTs observed are a sum of more components than just the decision time. Before a decision is made, stimuli have to be encoded, and after a decision is made, a response has to be executed. To account for this extra time, we add a fourth parameter that expresses the *encoding and responding time*. With the four parameters defined, the description of the basic model is now complete. Table 2 lists the four parameters and their psychological interpretations.

For use in the following section, let us assign symbols to the parameters. The boundary separation will be α, the initial bias will be β, the drift rate will be δ, and the encoding and responding time will be τ.

Some nuance

Throughout this dissertation, it will be assumed (unless specifically stated otherwise) that *the process described here is the real process that brings about each individual response by a participant to a stimulus.* While the substantive conclusions drawn may be robust to some misspecification of the true process, it is generally advisable to keep in mind the old adage: "Rubbish in, rubbish out." This holds true for the application of unsuited (e.g., linear) models to response times, but for the diffusion model as

well. If, for example, the experimental paradigm allows for self-correcting processes (i.e., a participant second-guessing a response), then a most basic assumption of the diffusion model—what we have called the "central dogma"—is violated and a diffusion model should not be applied.

On the other hand, if all the assumptions of the model seem acceptable, then the rewards of applying a diffusion model can be great: Rather than merely observing that some RTs are longer or shorter than others, a well-applied process model can paint a more detailed, substantively interesting picture.

Cognitive psychometrics

Psychometrics is the field of psychology that is concerned with measuring psychological traits such as ability and attitude. Clearly, measuring a person's ability to respond accurately in a speeded RT task falls under that definition. If we adopt the convention that "response A" is the *correct* response and "response B" is an *error*, then we can use the diffusion model as a psychometric tool.

In psychometrics, however, one typically wants to account for individual differences. Suppose that we want to apply a diffusion model to data collected on a large population. We are interested in the different items' i drift rate parameters $\delta_{(i)}$ (differences on the *item side*). In order to achieve a parsimonious model for these data, our model should not contain superfluous parameters. The different $\delta_{(i)}$ are necessary, but the other parameters (such as the person-specific boundary separations; $\alpha_{(p)}$ for person p) are not of major interest. However, it is important to allow for different boundary separations; we cannot merely assume that all participants apply the same speed-accuracy trade-off (i.e., that all $\alpha_{(p)}$ are equal, thereby constraining many parameters).

An elegant approach that is often applied in psychometrics is to consider the person-specific differences as *random effects*. By assuming that, for example, $\alpha_{(p)} \sim \mathrm{N}(\mu_\alpha, \sigma_\alpha^2)$, the number of parameters to be considered is reduced.Making this extra distributional assumption (i.e., that the values $\alpha_{(p)}$ follow a normal distribution in the population) implies that we will make statements about the larger population from which the parameter is a member. By including the information that parameters are members of the same, somewhat homogeneous population, we allow 'cross-talk' between data that would otherwise be considered completely independent. One case in which this has an obvious effect is when some parameters (that are members of the same population) are assigned a value that is relatively extreme. For these measurements, it is slightly more likely that they are (to some extent) the effect of measurement error. These parameters will be adjusted towards the mean parameter value to account for this—but the mean parameter value is of course influenced by all the other parameter values. In this way, the values of parameters are no longer independent of one another. The obvious conceptual advantage of this strategy is that each parameter is informed by a greater amount of data, and is hence less sensitive to measurement noise.

In the example with the ability parameter, the parameters $\theta_{(i)}$ are draws from the population that is defined by the normal distribution with mean μ_α and standard deviation σ_α. Interestingly, we can quite easily incorporate covariate information into this model. Suppose for example that the experiment was performed with two groups of participants (e.g., young and elderly). We might not want to assume that these participants' αs are draws from the same population, but rather of two distinct populations, so that

$$\alpha_{(p)} \sim \begin{cases} \mathrm{N}(\mu_{\alpha\ \mathrm{young}}, \sigma^2_{\alpha\ \ \mathrm{young}}) & \text{if } p \text{ young} \\ \mathrm{N}(\mu_{\alpha\ \mathrm{elderly}}, \sigma^2_{\alpha\ \ \mathrm{elderly}}) & \text{if } p \text{ elderly} \end{cases}$$

In the above example, we have applied what is called a *hierarchical model*, with the diffusion model as its *measurement level*. This combination is particularly interesting, because it would allow us to address a very clear question in a direct manner: Is there a difference in speed-accuracy trade-off between young and elderly participants? This contrast would show in a difference in the population parameters ($\mu_{\alpha\ \text{young}}$ vs. $\mu_{\alpha\ \text{elderly}}$ and $\sigma^2_{\alpha\ \text{young}}$ vs. $\sigma^2_{\alpha\ \text{elderly}}$).

The possibility to draw substantive, population-level conclusions is a direct result of the combination of an interesting cognitive model (like the diffusion model) with psychometric techniques (like hierarchical models). The application of this approach to the diffusion model is considered particularly apposite because this model provides a flexible and well-supported account of performance in a wide variety of simple decision tasks. Combining concepts from cognitive psychology with psychometrics is a relatively recent idea, and has been dubbed *cognitive psychometrics*.

Final considerations

Publication list

The thesis is a collection of five manuscripts that are either submitted or already accepted for publication. As a consequence of this formula, there is some overlap between the chapters. Save for some cosmetic changes, each chapter is true to the published work and can be read as a self-contained chapter.

The chapters correspond to the following original publications:

> **Chapter 1:** Vandekerckhove, J. & Tuerlinckx, F. (2007). Fitting the Ratcliff diffusion model to experimental data. *Psychonomic Bulletin & Review, 14*, 1011–1025.

Chapter 2: Vandekerckhove, J. & Tuerlinckx, F. (2008). Diffusion model analysis with MATLAB: A DMAT primer. *Behavior Research Methods, 40*, 61–72.

Chapter 3: Vandekerckhove, J., Tuerlinckx, F., & Lee, M. D. (2008). A Bayesian approach to diffusion process models of decision-making. In B. C. Love, K. McRae, & V. M. Sloutsky (Eds.), *Proceedings of the 30th Annual Conference of the Cognitive Science Society* (pp. 1429–1434). Austin, TX: Cognitive Science Society.

Chapter 4: Vandekerckhove, J., Tuerlinckx, F., & Lee, M. D. (2009). *Hierarchical diffusion models for two-choice response times.* Submitted.

Chapter 5: Vandekerckhove, J., Verheyen, S., & Tuerlinckx, F. (2009). *A crossed random effects diffusion model for speeded semantic categorization data.* Submitted.

The author of this dissertation is also author or coauthor of the following manuscripts:

Oravecz, Z., Tuerlinckx, F., & Vandekerckhove, J. (in press). A hierarchical Ornstein-Uhlenbeck model for continuous repeated measurement data. *Psychometrika*.

Panis, S., De Winter, J., Vandekerckhove, J., & Wagemans, J. (2008). Identification of everyday objects on the basis of fragmented versions of outlines. *Perception, 37*, 271-89.

Spruyt, A., Hermans, D., De Houwer, J., Vandekerckhove, J., & Eelen, P. (2007). On the predictive validity of indirect attitude measures: Prediction of consumer choice behavior on the basis of affective

priming in the picture-picture naming task. *Journal of Experimental Social Psychology, 43*, 599-610.

Vandekerckhove, J., Panis, S., & Wagemans, J. (2007). The concavity effect is a compound of local and global effects. *Perception & Psychophysics, 69*, 1253-1260.

Vandekerckhove, J., & Tuerlinckx, F. (in press). MATLAB for behavioral scientists: A novice's guide to MATLAB. *Experimental Psychology.*

Wetzels, R., Vandekerckhove, J., Tuerlinckx, F., & Wagenmakers, E.-J. (in press). Bayesian parameter estimation in the Expectancy Valence model in the Iowa gambling task. *Journal of Mathematical Psychology.*

Helpful references

On the topic of *linear modeling* and the background assumptions of the linear model, see Ramsey and Shafer (2002). For RT analysis in general, Luce (1986) is the standard reference work. More in-depth discussion of *sequential sampling models* in particular can be found in Bogacz, Brown, Moehlis, Holmes, and Cohen (2006) and Ratcliff and Smith (2004). On *model fitting* in general, Myung (2000) provides an excellent low-threshold introduction. The application to RT distributions is covered more thoroughly by Van Zandt (2000) and Heathcote, Brown, and Mewhort (2000).

The *diffusion model* for two-choice response times was introduced by Ratcliff (1978) and specialized fitting strategies for it are discussed in Ratcliff and Tuerlinckx (2002). An interesting review of recent *applications* of the model can be found in Wagenmakers (in press).

The budding field of *cognitive psychometrics*, finally, boasts very few

articles in the scientific literature. The term was coined by Batchelder (1998) and Batchelder and Riefer (1999), and the principle is applied in many articles by Rouder et al. (e.g., Rouder & Lu, 2005; Rouder et al., 2007). However, at the time of writing, we are not aware of any articles or book chapters providing a review cognitive psychometrics.

References

Batchelder, W. H. (1998). Multinomial processing tree models and psychological assessment. *Psychological Assessment*, *10*, 331–344.

Batchelder, W. H., & Riefer, D. M. (1999). Theoretical and empirical review of multinomial process tree modeling. *Psychonomic Bulletin & Review*, *6*, 57–86.

Bogacz, R., Brown, E., Moehlis, J., Holmes, P., & Cohen, J. D. (2006). The physics of optimal decision making: A formal analysis of models of performance in two-alternative forced choice tasks. *Psychological Review*, *113*, 700–765.

Heathcote, A., Brown, S., & Mewhort, D. J. K. (2000). The power law repealed: The case for an exponential law of practice. *Psychonomic Bulletin & Review*, *7*, 185–207.

Luce, R. D. (1986). *Response times*. New York: Oxford University Press.

Myung, I. J. (2000). The importance of complexity in model selection. *Journal of Mathematical Psychology*, *44*, 190–204.

Ramsey, F. L., & Shafer, D. W. (2002). *The statistical sleuth* (2nd ed.). Belmont, CA: Duxbury.

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, *85*, 59–108.

Ratcliff, R., & Smith, P. L. (2004). A comparison of sequential sampling

models for two-choice reaction time. *Psychological Review*, *111*, 333–367.

Ratcliff, R., & Tuerlinckx, F. (2002). Estimating parameters of the diffusion model: Approaches to dealing with contaminant reaction times and parameter variability. *Psychonomic Bulletin & Review*, *9*, 438–481.

Rouder, J. N., & Lu, J. (2005). An introduction to Bayesian hierarchical models with an application in the theory of signal detection. *Psychonomic Bulletin & Review*, *12*, 573-604.

Rouder, J. N., Lu, J., Sun, D., Speckman, P. L., Morey, R. D., & Naveh-Benjamin, M. (2007). Signal detection models with random participant and item effects. *Psychometrika*, *72*, 621–642.

Van Zandt, T. (2000). How to fit a response time distribution. *Psychonomic Bulletin & Review*, *7*, 424–465.

Wagenmakers, E.-J. (in press). Methodological and empirical developments for the Ratcliff diffusion model of response times and accuracy. *European Journal of Cognitive Psychology*.

Please, enjoy the read.

Design matrices

CHAPTER 1

Fitting the Ratcliff diffusion model to experimental data

Abstract

Many experiments in psychology yield both reaction time and accuracy data. However, no off-the-shelf methods yet exist for the statistical analysis of such data. One particularly successful model has been the diffusion process, but using it is difficult in practice because of numerical, statistical, and software problems. We present a general method for performing diffusion model analyses on experimental data. By implementing design matrices, a wide range of across-condition restrictions can be imposed on model parameters, in a flexible way. It becomes possible to fit models with parameters regressed onto predictors. Moreover, data-analytical tools are discussed that can be used to handle various types of outliers and contaminants. We briefly present an easy-to-use software tool that helps perform diffusion model analyses.

1.1 Introduction

Mental chronometry, the study of psychological processes through observed response times, is one of the most prevalent approaches in cognitive psychology. As early as 1868, Donders (1969) used reaction time measurements in order to investigate differences between mental processes. Since then, reaction time studies have been used in perhaps all fields of cognitive science. Such is the importance of reaction time data to cognitive psychology that methods for analyzing them have become an object of study in their own right (e.g., Luce, 1986).

Continuing this trend, considerable attention has been lent to the combination of reaction time and accuracy data (a ubiquitous combination often referred to as two-choice response time data). For the analysis of this type of data, several nonlinear statistical models have been developed, often with substantive interpretations attached to the parameters and underlying processes (e.g., the discrete random walk model; Laming, 1968; Link & Heath, 1975). A more advanced model—and the one that is at the heart of the present article—is the Ratcliff diffusion model (Ratcliff, 1978; Ratcliff, Van Zandt, & McKoon, 1999). The latter model, which will be described in detail in the next section, has performed remarkably well in the analysis of two-choice response time data. It has successfully been applied to experiments in many different fields, such as memory (Ratcliff, 1978, 1988), letter matching (Ratcliff, 1981), lexical decision (Ratcliff, Gomez, & McKoon, 2004; Wagenmakers, Ratcliff, Gomez, & McKoon, 2008), signal detection (Ratcliff & Rouder, 1998; Ratcliff, Thapar, & McKoon, 2001; Ratcliff et al., 1999), visual search (Strayer & Kramer, 1994), and perceptual judgment (Ratcliff, 2002; Ratcliff & Rouder, 2000; Thapar, Ratcliff, & McKoon, 2003; Voss, Rothermund, & Voss, 2004). In particular, the Ratcliff diffusion model (RDM) succeeds in explaining characteristic

aspects of two-choice response time data such as the occurrence of both fast and slow errors. With the RDM, it is possible to make statements about entire distributions of correct and error latencies, and the parameter estimates allow for inferences that are much more detailed than those provided by classical models such as ANOVA or curve fitting. In particular, the RDM's parameters—which will be described in detail in the next section—can provide insight into the relative contributions of different factors such as quality of the input stimulus, conservativeness of the participant, and time spent on processes other than deciding.

In spite of its advantages, the Ratcliff diffusion model has not yet become a popular or widely used method to analyze two-choice response time data. The reasons for this lack of dispersion have to do with numerical, statistical, and software issues (see also W. Schwarz, 2001). The first set of reasons concerns the fact that the model is prohibitively difficult to implement for applied researchers because of numerical difficulties. One has to deal with an infinite oscillating series in the expression for the cumulative distribution function (CDF) or probability density function (PDF; see Ratcliff & Tuerlinckx, 2002). In addition, some of the parameters are allowed to vary from trial to trial and this leads to (partly) intractable integrals (Ratcliff & Tuerlinckx, 2002; Tuerlinckx, 2004). Recently, Voss and Voss (2008) have proposed a method to circumvent the problem but their solution requires a numerical solution of a partial differential equation. However, once the CDF or PDF have been computed, the task of estimating the parameters still requires some skill regarding function optimization because no analytical estimators exist. In sum, some experience with numerical methods is needed to implement the model.

The second group of reasons that have forestalled widespread use of the RDM is related to statistical issues. The type of data used to apply the diffusion model is rather complex. On each trial, there is a bivariate

response, consisting of a reaction time and a choice. The latter is binary and the former is continuous, but non-normal (positively skewed and with a lower boundary possibly different from zero). The treatment of such data is not a trivial issue and traditional statistical methods suited for linear analysis (e.g., relying on means, computing R^2, etc.) fail in this case. In addition, the reaction time measure is possibly muddied with outliers and contaminants.

The third category of reasons has to do with the fact that at the time of this writing, there is no flexible or general software available for diffusion model analysis. Exceptions are the new program developed by Voss and Voss (2007) and EZ-diffusion by Wagenmakers, van der Maas, and Grasman (2007, see also Appendix 1.B). However, the latter is not able to fit the full RDM. Up until now, in each of the substantive studies cited above that made use of the RDM, fitting software was custom-written. However, researchers often collect data in a design that deviates from designs for which the previously developed software was written, which impedes the application of the custom-written software.

It is the goal of the current paper to make diffusion model analysis more accessible to a general public of researchers by providing numerical and statistical methods that are useful when fitting the Ratcliff diffusion model. Also, we provide some demonstration of a MATLAB tool that implements the methods we present (the *Diffusion Model Analysis Toolbox* or DMAT; Vandekerckhove & Tuerlinckx, 2007). For an introduction to the practical side of working with DMAT, however, we refer the interested reader to the DMAT primer (Vandekerckhove & Tuerlinckx, 2008).

In what follows, we start with a brief explanation of the RDM. Next, we outline a design matrix method that permits one to impose substantive restrictions on the model's parameters. This flexible technique facilitates fitting of the RDM and allows for the construction of models that can

capture a variety of substantive hypotheses. Subsequently, we will discuss techniques related to the estimation of the parameters of the RDM (i.e., the handling of outliers and the construction and minimization of the loss function). In the following section, we will describe the necessary statistical methods for testing substantive hypotheses and comparing different models. We then briefly introduce our diffusion model analysis toolbox for MATLAB. We present results from simulation studies where properties of these statistical methods are investigated. Finally, we demonstrate the use of our methods and software in two example applications.

1.2 The Ratcliff diffusion model

1.2.1 Parameters of the model

The diffusion process (see Figure 1.1) has been used to describe and model the decision component in simple two-choice tasks. In the model, it is assumed that an observer has a one-dimensional internal representation of evidence. When the observer is presented with a stimulus, information regarding it is accumulated sequentially over time until its total amount reaches the upper or lower bound, resulting in a response (absorbing boundaries). The decision time is defined as the time from the start of the process until the moment one of the absorbing boundaries is reached.

The RDM has seven parameters. The first parameter is the boundary separation, denoted by a. If a is small, the process is expected to end sooner but it is more prone to error since random variability inherent to the decision process may cause it to end up at the wrong boundary. When a is large, both accuracy and expected reaction time will increase. The distance between the two absorbing boundaries therefore regulates the relation between speed and accuracy (the so-called speed-accuracy trade-

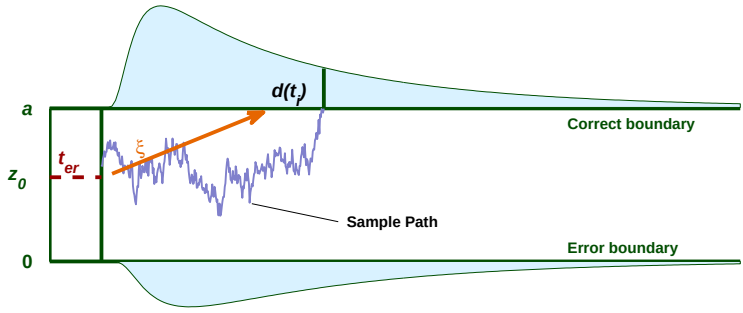


Figure 1.1: *An illustration of the Ratcliff diffusion model.*

off).

A second property of the model is the starting point of the information accumulation process, which is denoted as z_0 ($0 < z_0 < a$). This parameter introduces the possibility of response bias in the decision process because the process is more likely to end at the boundary closer to the starting point. We will assume z0 to vary from trial to trial (Laming, 1968), according to a uniform distribution, with mean z ($0 < z < a$) and range s_z ($0 < s_z < \min(z, a - z)$). These two, z and s_z, are the second and third parameters of the RDM.

Furthermore, the information accumulation process can have a tendency to drift off to one of the two absorbing boundaries, depending on the quality of the stimulus presented. This information accumulation rate, or drift rate, is assumed to vary within a trial, following a Gaussian distribution with mean ξ and standard deviation s, but also across trials (Ratcliff, 1978), such that ξ follows a Gaussian distribution with mean v and standard deviation η. An experimental condition with non-ambiguous stimuli will lead to a large positive mean drift rate v, thus a high probabi-

lity of hitting the upper boundary (indicating a correct response) in a short time. The standard deviation s, which indicates the volatility in drift rate in a single trial, is a non-identified parameter in the model, so we fix it to the arbitrary value 0.1 (which is a consensus value in the literature, e.g., Ratcliff et al., 1999). Thus, we add a fourth and fifth parameter to the model, namely the mean drift rate v and its intertrial standard deviation η.

Finally, another component of the model is the time needed to perform non-decision processes such as encoding of the stimulus, response preparation and execution of the motor response (Luce, 1986). We denote the non-decision part of the observed reaction time as t_{er}. This t_{er} is assumed to vary from trial to trial, according to a uniform distribution with mean T_{er} and range s_t. These two are the sixth and seventh parameters of the RDM.

1.2.2 Some notational conventions

In the preceding section, we have defined the seven key parameters of the diffusion model. We will sometimes capture all of these parameters in a parameter vector $\theta_{(c)} = \big(a_{(c)}, T_{er\,(c)}, \eta_{(c)}, z_{(c)}, s_{z(c)}, s_{t(c)}, v_{(c)}\big)$, where the bracketed subscript (c) refers to the c^{th} condition in an experiment, and $c = 1, \ldots, C$. When working with different conditions in an experiment (and thus different parameter vectors), we will vertically concatenate the parameter vectors into a parameter matrix $\mathbf{P}$. Thus, if we have C conditions, $\mathbf{P} =$

$$\left\{ \begin{array}{ccccccc} a_{(1)} & T_{er(1)} & \eta_{(1)} & z_{(1)} & s_{z(1)} & s_{t(1)} & v_{(1)} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ a_{(C)} & T_{er(C)} & \eta_{(C)} & z_{(C)} & s_{z(C)} & s_{t(C)} & v_{(C)} \end{array} \right\}.$$

A single column in such a parameter matrix then contains estimates of one specific parameter over conditions, and such a column vector will be denoted with a $\mathbf{\Psi}$. For example, the nondecision time in condition c will be denoted as $T_{er(c)}$, which is the c^{th} element of $\mathbf{\Psi}_{T_{er}}$ (the second column of $\mathbf{P}$), and the second element of $\theta_{(c)}$ (the c^{th} row of $\mathbf{P}$).

Finally, we will often use a plain θ to refer to a generic (i.e., any) parameter.

1.3 The design matrix method

There are several reasons why a researcher might not be interested in fitting a model with all parameters free. First, there is the issue of parsimony. Fitting the Ratcliff diffusion model to an experiment with C conditions would leave us with $7 \times C$ distinct parameters to estimate. Even if the number of conditions is moderate, for example $C = 5$, this leads to a large number of parameters to be estimated (i.c., 35 parameters to be estimated). Therefore, it seems that some reduction in the number of parameters is needed from a pragmatic point of view.

Secondly, and more importantly, in many situations one may want to impose substantive restrictions on the parameters, which will in effect lead to a reduction in the number of parameters. An obvious example of such a restriction is the requirement that a certain parameter equals a known constant. For example, it can be hypothesized that the range of nondecision time, s_t, equals zero for all conditions ($s_{t(c)} = 0$ for $c = 1, \ldots, C$). In this way, s_t has been dropped from the model (below it will be shown how it can be evaluated whether this restriction makes sense). Another popular substantive restriction in the context of the diffusion model is the requirement of a symmetric diffusion process ($z_{(c)} = a_{(c)}/2$ for $c = 1, \ldots, C$).

However, we can go a step further by carrying out a regression of the parameters onto a set of predictors. To elucidate this concept, assume that a researcher has set up a brightness discrimination task (RatcliffRouder1998; see also Example 2 in this paper). Suppose furthermore that there are 33 levels of brightness defined by increasing the number of white pixels in each step with an equal number. For the moment, the focus will be on the drift rates. Not restricting the drifts in any way will lead to 33 drift parameters to be estimated. However, the researcher may want to test the hypothesis that the drift rate varies linearly with brightness level:

$$v_{(c)} = v^*_{(1)} + B_{(c)} v^*_{(2)}.$$

where $B_{(c)}$ refers to the brightness level in condition c and $c = 1, \ldots, C$. In this example, we have reduced the number of parameters to be estimated from 33 to 2. (Note also that we have introduced a new notation here: basic or design parameters are marked with a star.)

In general, the drift rate in condition c can be decomposed into a weighted linear combination of M known predictor values:

$$v_{(c)} = \sum_{j=1}^{M} d_{(cj)} v^*_{(j)} \tag{1.1}$$

where $d_{(cj)}$ is the value of the j^{th} predictor in condition c. In the aforementioned example, $M = 2$, $d_{(c1)} = 1$ and $d_{(c2)} = B_{(c)}$. Because we have C linear equations as in Equation 1.1 (one for each drift rate), we can make

use of matrices and vectors to represent them all at once:

$$
\begin{aligned}
\boldsymbol{\Psi}_v &= \begin{bmatrix} v_{(1)} \\ \vdots \\ v_{(c)} \\ \vdots \\ v_{(C)} \end{bmatrix} = \begin{bmatrix} \sum_{j=1}^{M} d_{(1j)} v^*_{(j)} \\ \vdots \\ \sum_{j=1}^{M} d_{(cj)} v^*_{(j)} \\ \vdots \\ \sum_{j=1}^{M} d_{(Cj)} v^*_{(j)} \end{bmatrix} \\
&= \begin{bmatrix} d_{(11)} & \cdots & d_{(1M)} \\ \vdots & \ddots & \vdots \\ d_{(c1)} & d_{(cj)} & d_{(cM)} \\ \vdots & \ddots & \vdots \\ d_{(C1)} & \cdots & d_{(CM)} \end{bmatrix} \times \begin{bmatrix} v^*_{(1)} \\ \vdots \\ v^*_{(j)} \\ \vdots \\ v^*_{(M)} \end{bmatrix} \\
&= \mathbf{D}_v \times \mathbf{v}^*.
\end{aligned}
$$

The design matrix $\mathbf{D}$ is a $C \times M$ matrix where each column represents a predictor (e.g., an intercept, an experimental treatment, a measured variable, etc.). The design matrix $\mathbf{D}$ is then multiplied with an $M \times 1$ design parameter vector, to recover a $C \times 1$ model parameter vector $\boldsymbol{\Psi}$.

The idea of regressing the parameters onto a set of predictors can be applied to all parameters in the model and is by no means restricted to the drift rates. Because a different design matrix can be used for each parameter, $\mathbf{D}$ is indexed with the parameter symbol in order to make it clear to which parameter the design corresponds. The entire parameter matrix $\mathbf{P}$ can be described in terms of only the seven (known) design matrices $\mathbf{D}$ and the seven design parameter vectors $\boldsymbol{\Psi}$. The result is that, when fitting the model to the data, only the elements of the parameter vectors (as opposed to all the diffusion parameters) have to be estimated.

Two special and interesting cases of design matrices $\mathbf{D}$ are worth men-

tioning. The first special case is where $\mathbf{D}$ consists of a column of ones. This can be illustrated for the parameter T_{er} as follows:

$$\begin{bmatrix} T_{er(1)} \\ \vdots \\ T_{er(c)} \\ \vdots \\ T_{er(C)} \end{bmatrix} = \begin{bmatrix} 1 \\ \vdots \\ 1 \\ \vdots \\ 1 \end{bmatrix} \times T_{er(1)}^{*}.$$

The result of this is that the C conditions have the same T_{er}. In a second special case, $\mathbf{D}$ equals the $C \times C$ identity matrix such that each of the C conditions has a different value for a certain parameter. In the case of an identity matrix as the design matrix, there is no restriction of parameters across conditions.

To illustrate the usefulness of the design matrix method, let us consider a final example. Suppose we want to fit a drift rate to the first condition and allow the drift rates of the other conditions to deviate from the first condition (but all in the same way). This can be implemented by defining the design matrix

$$\mathbf{D}_v = \begin{bmatrix} 1 & 0 \\ 1 & 1 \\ \vdots & \vdots \\ 1 & 1 \end{bmatrix}$$

with $\mathbf{\Psi}_v = \mathbf{D}_v \times \begin{bmatrix} v_{(1)}^{*} \\ v_{(2)}^{*} \end{bmatrix}$, and therefore $v_{(1)} = v_{(1)}^{*}$ and $v_{(2)} = v_{(1)}^{*} + v_{(2)}^{*}$ for all $c \neq 1$ (see Chapter 6 in Littell, Stroup, & Freund, 2002, for more details on the construction of design matrices).

In general, we formulate the parameter matrix $\mathbf{P} =$

$$\{\mathbf{D}_a \times \mathbf{a}^*, \mathbf{D}_{T_{er}} \times \mathbf{T_{er}}^*, \mathbf{D}_\eta \times \eta^*, \mathbf{D}_z \times \mathbf{z}^*, \mathbf{D}_{s_z} \times \mathbf{s_z}^*, \mathbf{D}_{s_t} \times \mathbf{s_t}^*, \mathbf{D}_v \times \mathbf{v}^*\}.$$

Then, all the elements of $\mathbf{a}^*$, $\mathbf{T_{er}}^*$, η^*, $\mathbf{z}^*$, $\mathbf{s_z}^*$, $\mathbf{s_t}^*$, and $\mathbf{v}^*$ are the parameters over which we want to optimize the fit to data.

Creative use of design matrices allows one to impose substantive restrictions on parameter sets, and will enable researchers to test specific substantive hypotheses. Extending the diffusion model with the design matrix methodology, it becomes possible to build a type of "analysis of variance/multiple regression"-diffusion model.

Using the design matrix method entails two restrictions, however. Firstly, only linear decompositions (i.e., linear in the basic parameters) can be represented by matrices. Secondly, only restrictions across conditions are possible, while restrictions across parameters (e.g., restricting z to be equal to $a/2$) requires a different strategy. Nonetheless, implementing restrictions using design matrices is a very flexible and powerful tool which has gained some attention as well in other areas (e.g., see De Boeck & Wilson, 2004, for a wide variety of applications in psychometrics).

1.4 Statistical inference: Estimation

Finding the parameters of the Ratcliff diffusion model, given a data set, is something of a challenge. Before starting, several nontrivial choices need to be made, in particular regarding how to deal with outliers and other contaminant reaction times, the objective function to use in the estimation step, and the precise method of optimization of the latter function. In this section, we discuss each of these choices, but for details we will refer the reader to Appendices 1.A and 1.B. A crucial part of any algorithm to fit the

diffusion model is the efficient computation of its cumulative distribution function (CDF). For this, we rely heavily on the methods described in Tuerlinckx (2004).

1.4.1 Outlier handling strategies

An important issue to consider when applying a statistical model to reaction time data is that of contaminants—data points that appear in the data sets but that are somehow not germane to the research question. A well-known class of contaminants is outliers (data points that are outside the range of normal observations), but other examples are random guesses (data from trials where the participant somehow missed the stimulus and guessed), delayed start-ups (where the participant was somehow inappropriately delayed in responding), and fast guesses (where the participant executed a response before having actually inspected the stimulus).

Each of these types of contaminants can severely muddy the data (Ratcliff, 1993; Ratcliff & Tuerlinckx, 2002; Ulrich & Miller, 1994), possibly resulting in biased parameter estimates and incorrect standard errors of estimation. A fitting procedure for a reaction time model such as the one considered in this paper should therefore always be equipped with a proper strategy for handling these contaminants. We opt for a combination of two methods: First, the data are preprocessed with an exponentially weighted moving average (EWMA) control method that gives the minimal reaction time necessary for inclusion in the data analysis and second, a mixture model is fitted to the data.

The EWMA method is an optional new method that is used in a preprocessing step in order to filter out reaction times that are suspected of being fast guesses. The idea behind this method is that the identification of fast guesses is made possible because they tend to have a specific signa-

ture, being responses with a very short RT and chance level performance. A method suggested by this property of fast guesses sorts the data points according to the response times and finds the minimal response time at which the responses begin to deviate from what we expect when guessing. This minimal response time is used as a lower cut-off value such that all observations with shorter RTs are censored. More technical detail is provided in Appendix 1.A.

The mixture model approach to handle outliers was first described in Ratcliff and Tuerlinckx (2002) but it is extended here to cope with a larger variety of contaminants. The basic idea is that each trial has a probability of $(1-\pi)\gamma$ of being a guess, a probability of $(1-\pi)(1-\gamma)$ of being a 'delayed startup', and a remaining probability π of being an actual realization of a diffusion process (note that this model reduces to the original RDM if π is 1). Each trial can then be represented by the decision tree shown in Figure 1.2. The first step leads to either the diffusion process (with probability π) or to a contamination process (with probability $1-\pi$). A contamination process can in turn be a 'guess' (with probability γ) or a 'delayed startup' response (with probability $1-\gamma$). Note that this treatment adds two free parameters to the model (π and γ), yielding an extended RDM. Details concerning the component distributions and the mixture distribution are provided in Appendix 1.A.

1.4.2 The loss function

To estimate the best fitting parameters of the RDM (or the extended RDM), given a data set, we have to find the maximum of a likelihood function, or the minimum of some deviance function. For our loss function, we use a negative multinomial log-likelihood function (MLF), but other options are available in the program (see below, The Diffusion Mo-

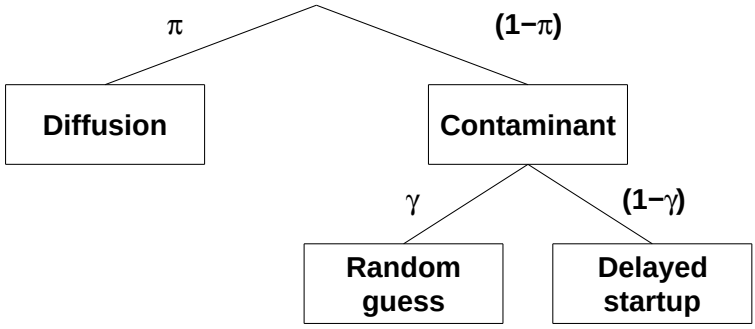


Figure 1.2: *A decision tree structure illustrating the mixture model.*

del Analysis Toolbox). We opt for the MLF for several reasons, chiefly among them its computational tractability compared to continuous likelihood and its robustness in the face of a small amount of contaminants and outliers (see Ratcliff & Tuerlinckx, 2002). Briefly, the loss function is defined as -2 times the natural logarithm of the joint likelihood of observing the observed number of data points in each of a set of predefined "reaction time bins". We call this statistic Λ. Details regarding Λ and its optimization are provided in Appendix 1.B.

1.5 Statistical inference: Testing and model selection

After having estimated the parameters of one or more models, the researcher may want to test hypotheses about the parameters and/or compare models. We distinguish between testing a hypothesis about a single parameter with the Wald test, comparing two nested models, and comparing non-nested models.

1.5.1 The Wald test for a hypothesis about single parameter

The univariate Wald test can be used to test the null hypothesis that $\theta = \theta_0$ (versus the alternative $\theta \neq \theta_0$). It starts from the Wald statistic $Z = \frac{\hat{\theta}-\theta_0}{s_{\hat{\theta}}}$, where $\hat{\theta}$ is the point estimate of some parameter θ and $s_{\hat{\theta}}$ the standard error. Under the null hypothesis and under some regularity conditions, Z follows approximately a standard normal distribution (or, equivalently, Z^2 follows a χ^2_1-distribution; Bishop, Fienberg, & Holland, 1975; the univariate Wald test is equivalent to a "Z-test").

Although the regularity conditions are fairly general, one of them is noteworthy. The Wald statistic should not be used if the test value θ_0 is at a boundary of the parameter space (Bishop et al., 1975, but see also Stram & Lee, 1994, for an adaptation of the reference distribution). As a consequence, it cannot be used to test the null hypothesis that, for example $\eta = 0$, since η is bounded at 0.

Note also that a multivariate Wald test is possible to test a composite null hypothesis about several parameters (Bishop et al., 1975).

1.5.2 Comparing two nested models

A model, called the reduced model, is nested in another model, called the full model, if the reduced model can be reached by setting restrictions on the parameters of the full one (e.g., setting some of the parameters to zero). Such nested models can be compared through the likelihood ratio test (LRT). In this way, joint hypotheses about several parameters simultaneously can be tested. The LRT is very helpful in combination with the design matrix approach because Model 1 is nested in Model 2 if for a given parameter the columns of the design matrix of Model 1 ($\mathbf{D_1}$) lie in the space spanned by the columns of the design matrix of Model 2 ($\mathbf{D_2}$) (where we assume that the design matrices for the other parameters are

kept constant). That is, the models are nested if each column of $\mathbf{D_1}$ can be represented as a linear combination of the columns of $\mathbf{D_2}$.

For example, a researcher might want to test whether an experimental manipulation has had some influence on drift rate. To that end, one could compare a model in which all drift rates are constrained to be equal to a model where they are free to vary over conditions. The former model—the reduced model—could be formulated as

$$\mathbf{P_{reduced}} = \left\{\mathbf{1_C}\times\mathbf{a}^*, \mathbf{1_C}\times\mathbf{T_{er}}^*, \mathbf{1_C}\times\eta^*, \mathbf{1_C}\times\mathbf{z}^*, \mathbf{1_C}\times\mathbf{s_z}^*, \mathbf{1_C}\times\mathbf{s_t}^*, \mathbf{1_C}\times\mathbf{v}^*\right\},$$

where $\mathbf{1_C}$ indicates a $C \times 1$ vector with all elements equal to 1 (C being the number of conditions in the experiment). This model restricts all parameters to be equal across conditions, while the latter model—the full model—is then:

$$\mathbf{P_{full}} = \left\{\mathbf{1_C}\times\mathbf{a}^*, \mathbf{1_C}\times\mathbf{T_{er}}^*, \mathbf{1_C}\times\eta^*, \mathbf{1_C}\times\mathbf{z}^*, \mathbf{1_C}\times\mathbf{s_z}^*, \mathbf{1_C}\times\mathbf{s_t}^*, \mathbf{I_{C\times C}}\times\mathbf{v}^*\right\},$$

where the drift rates are now determined by the multiplication of the $C\times C$ identity matrix $\mathbf{I_{C\times C}}$ and the design parameter matrix $\mathbf{v}^*$. Thus, the restriction on drift rate v is now released and $C-1$ parameters have been added to the model (because $\mathbf{v}^*$ now contains C elements instead of 1). Note that the columns of the drift design matrix in the reduced model lie in the column space of the design matrix of the full model.

The LRT statistic $\Delta\Lambda$ is the difference between the negative of twice the log-likelihood of the reduced model and the negative of twice the log-likelihood of the full model: $\Delta\Lambda = \Lambda_{\mathbf{P_{reduced}}} - \Lambda_{\mathbf{P_{full}}}$. Under the null hypothesis (i.e., that the reduced model is true), $\Delta\Lambda$ follows approximately

a chi-square distribution with number of degrees of freedom equal to the difference in number of parameters between the full model and the reduced model: $\Delta\Lambda = \Lambda_{\mathbf{P_{reduced}}} - \Lambda_{\mathbf{P_{full}}} \sim \chi^2_{\Delta df}$. In the case of the example given in the previous paragraph, $\Delta df = C - 1$.

Two things should be noted about the LRT. First, the same boundary condition applies here as for the Wald statistic: if the reduced model parameter set $\mathbf{P_{reduced}}$ is at an edge of the parameter space of the full model, this statistic should not be used. Second, the distribution of $\Delta\Lambda$ assumes that fixed reaction time bins were used (see Appendix 1.B). In the case where percentile-based bins were used, the chi-square assumption does not hold (see Speckman & Rouder, 2004; see also Appendix 1.B).

1.5.3 Comparing non-nested models

If two models are not nested, then model selection may be carried out by using information criteria such as the Akaike Information Criterion (in this paper we use the small sample version $AICc$; Hurvich & Tsai, 1989) or the Bayesian Information Criterion (BIC; G. Schwarz, 1978). The two measures are defined as $AICc = \Lambda + 2dN/(N - d - 1)$ and $BIC = \Lambda + d\log(N)$, where N is the total number of observations and d indicates the number of free parameters in the model in question. In both cases, the model with the lower value on the criterion is preferred. Of course, these criteria can equally validly be applied to nested models.

1.6 The Diffusion Model Analysis Toolbox

In an attempt to further popularize the diffusion model, we have developed a MATLAB (version 2006a; ©1994 The MathWorks, Inc.) application, which should allow researchers with less technical background to use the diffusion model in practice. The program, which is called DMAT (for

Diffusion Model Analysis Toolbox) can be freely downloaded from the website of the K.U.Leuven Research Group for Quantitative and Personality Psychology (`http://ppw.kuleuven.be/okp/dmatoolbox`).

In creating DMAT, we had two main goals in mind. The program should be (1) accurate and efficient and (2) user-friendly. We believe that we have achieved both goals to a satisfactory degree. Regarding accuracy and efficiency, DMAT performs well in simulations (see below) testing the recovery of model and design parameters from simulated data (estimation biases are generally low and standard errors small). In addition, on our desktop PCs, the algorithm typically converges in less than one minute's time. The program is developed to make use of all fitting and modeling strategies we have discussed above (and more).

Regarding flexibility and ease of use, we have added a graphical user interface (GUI). (Note that a MATLAB command interface is also available and offers more flexibility.) Also, wherever we could, we have provided default settings that we believe will perform well in most cases, and we have written an instructional primer to the use of the toolbox (Vandekerckhove & Tuerlinckx, 2008).

1.7 Simulations

To evaluate aspects of the tools described above, we performed many Monte Carlo simulations, of which we report here a selection. We discuss the results of three simulation studies in which the performance of the estimation method is tested and two more simulations are carried out to evaluate properties of the inferential statistics associated with using the RDM.

Throughout, we use six standard parameter sets (A through F), which are reported in Table 1.1. Note that in each of these parameter sets, there

Table 1.1: *Standard parameter sets used in the simulations. See text for details.*

	Parameter Set					
	A	B	C	D	E	F
a	0.08	0.08	0.16	0.16	0.16	0.16
T_{er}	0.30	0.30	0.30	0.30	0.30	0.30
η	0.08	0.16	0.08	0.16	0.08	0.16
z	0.04	0.04	0.08	0.08	0.08	0.08
s_z	0.02	0.02	0.02	0.02	0.10	0.10
s_t	0.02	0.02	0.02	0.02	0.10	0.10
v_1	0.40	0.40	0.30	0.30	0.30	0.30
v_2	0.25	0.25	0.20	0.20	0.20	0.20
v_3	0.10	0.10	0.10	0.10	0.10	0.10
v_4	0.00	0.00	0.00	0.00	0.00	0.00

are four conditions, across which all parameters are identical, except for drift rate, which systematically varies (in Table 1.1, there are ten columns with parameters: six with the parameters that are constant across conditions, and one for each condition's drift rate). We borrow these parameter sets from Ratcliff and Tuerlinckx (2002). Our simulation method was based on the rejection method described in Tuerlinckx, Maris, Ratcliff, and De Boeck (2001).

1.7.1 Asymptotic parameter recovery

As a first test of the estimation algorithm, we used it to estimate RDM parameters, given the true (i.e., expected) proportions in each of the bins of the likelihood function (see Equation 1.5 in Appendix 1.B). In other words, as input we use the exact proportions of observations that each reaction time bin would have, given a certain set of parameters. Under this condition, there should be perfect recovery of the parameter values. This test was carried out under many different parameter sets, including the ones in Table 1.1. In each case, the algorithm returned the exact

parameter values to the requested accuracy (this was the case for each objective function DMAT allows).

1.7.2 Preasymptotic parameter recovery

As a second test of the estimation algorithm, we performed a series of simple simulations to investigate biases and standard errors of the parameter estimates. We define the relative bias of each parameter as $\frac{\bar{\hat{\theta}}-\theta}{\theta} \times 100\%$, and the standard error as $\sqrt{\frac{1}{R-1}\sum_{j=1}^{R}\left(\hat{\theta}_j - \bar{\hat{\theta}}\right)^2}$, with R the number of replications, and $\hat{\theta}$ and $\bar{\hat{\theta}}$ respectively the estimate and the mean estimate of the parameter θ.

From each of the six parameter sets shown in Table 1.1, we generated 100 data sets with 250 data points in each condition (without outliers). We used DMAT to find parameter estimates and calculated relative biases and standard errors within each parameter set. As can be seen from Table 1.2, the simulation parameters are generally well recovered. The a, T_{er}, and z estimates tend to be within 10% of their simulation values. The relative biases of the v estimates are slightly larger for large values of the "true" v. As already seen in Ratcliff and Tuerlinckx (2002), the variance parameters tend to be slightly more difficult to estimate, in particular when they are small compared to the means of the distributions.

We repeated this simulation for three different sample sizes: instead of having 250 observations per condition, we used simulated data sets with 100, 500, and 2500 observations per condition. In Table 1.3 we show relative biases and standard errors for each sample size. To conserve space, we report average (absolute) values over the six standard parameter sets. As can be seen in Table 1.3, biases and standard errors are somewhat higher for the $N = 100$ condition, especially those regarding the starting point range and larger drift rates. As expected, biases decrease strongly

Table 1.2: *Recovery of simulation parameters by DMAT. Relative biases are in the upper half, standard errors (multiplied by 1000) in the lower half.*

Parameter Set	a	T_{er}	η	z	s_z	s_t	v_1	v_2	v_3	$v_4^\dagger$
A	2	1	1	3	7	-38	11	4	5	-2
B	2	1	8	2	6	-49	8	5	5	-1
C	3	4	2	5	91	-3	14	7	5	-3
D	7	4	23	7	127	1	21	17	18	0
E	3	-1	1	3	-1	-1	11	5	4	-0
F	4	0	9	4	-1	2	10	7	8	-1
A	4	7	72	2	21	13	78	44	28	23
B	4	7	65	2	22	13	64	46	31	25
C	14	18	48	7	43	33	70	45	26	13
D	26	21	92	13	58	31	128	77	46	20
E	13	24	47	7	35	36	64	44	25	13
F	25	27	89	13	56	32	113	71	42	20

† Biases for v_4 are not relative biases, but the actual recovered values multiplied by 1000 (since the true values were zero)

as N increases, and standard errors decrease with a factor $\sqrt{5}$ (i.e., the standard errors are roughly proportional to the square root of the sample size).

1.7.3 Outlier handling strategies

To test the outlier treatment procedure applied by our algorithm (see Appendix 1.A), we performed four more simple simulation runs to evaluate the combined EWMA / mixture model approach. In each simulation run, we again generated 100 data sets from each of the six parameter sets shown in Table 1.1, with 250 data points in each condition.

We employed a simple two-by-two design: we either added outliers to the simulated data or did not, and we either enabled the outlier treatment or did not. When we did add outliers, 2.5% were fast guesses (RTs were draws from a uniform distribution between 200 and 400ms and accuracy

Table 1.3: *Recovery of simulation parameters by DMAT, when the number of observations in each of four conditions is varied over three levels (100, 500, and 2500). The simulation was run with each of the six standard parameter sets, but the results are averaged here (for the relative biases, we averaged over absolute values). Relative biases are in the upper half, standard errors (multiplied by 1000) in the lower half.*

Sample size (N)	a	T_{er}	η	z	s_z	s_t	v_1	v_2	v_3	$v_4^\dagger$
100	4	2	12	6	49	5	23	15	4	4
500	1	1	4	1	11	20	4	3	2	1
2500	1	0	3	1	12	12	2	2	2	1
100	18	24	89	10	48	42	126	86	45	31
500	6	11	38	3	25	18	41	29	19	13
2500	3	5	16	1	13	9	17	12	8	6

† Biases for v_4 are not relative biases, but the actual recovered values multiplied by 1000 (since the true values were zero)

was about 50%) and an additional 2.5% were delayed startups (RT draws from a uniform distribution between 500 and 3000ms, but with accuracy as expected under the diffusion model). We then estimated the parameters for each data set with DMAT and compared parameter recovery. In Table 1.4, the results are shown for parameter set A. As can be seen, if the data set did contain outliers with the aforementioned properties, and they are not accounted for, estimation biases increase dramatically, to over 100% for some drift values. When the combined EWMA / mixture model method is applied, relative biases return to the same magnitude as in the condition where no outliers existed.

To conserve space, we do not report results for the other parameter sets here, but as it turns out, our outlier treatment succeeds in alleviating the influence of outliers and contaminants on parameter estimates: biases and standard errors of the parameters that the adapted algorithm returned from the contaminated data set are closer to those of the parameters that the original algorithm returns from a "clean" data set, and they are lower

Table 1.4: *DMAT's recovery of simulation parameters from either "clean" or outlier-contaminated data (generated from parameter set A in Table 1.1), with our outlier correction method either enabled or disabled. Relative biases are in the left half, standard errors (multiplied by 1000) in the right half.*

	Relative bias				Standard errors			
Outliers added:	No	Yes	No	Yes	No	Yes	No	Yes
Outliers treated:	No	No	Yes	Yes	No	No	Yes	Yes
a	2	62	-5	0	4	6	4	3
T_{er}	1	3	3	2	7	8	8	8
η	4	513	-60	-14	64	24	46	63
z	3	55	-3	1	2	4	2	2
s_z	25	471	12	44	21	18	18	20
s_t	-40	191	14	-3	13	11	16	23
v_1	10	112	15	6	71	49	72	60
v_2	4	110	3	2	40	116	32	37
v_3	4	166	0	0	23	35	18	25
$v_4^{\dagger}$	0	0	0	0	21	47	16	16
π			.96	.94			29	6
γ			.22	.05			347	90

† Bias values for this parameter are actual recovered values, not relative biases.

than those from the original algorithm on the contaminated data set.

It should be noted, however, that other processes than the ones assumed here might generate contaminant reaction times. If that is the case, then parameter estimates might still be biased, in spite of the correction mechanisms proposed in this paper.

1.7.4 Power analyses

1.7.4.1 Power analysis 1

In another series of simulations, we evaluated the power of the likelihood ratio test. From each of the six parameter sets shown in Table 1.1, we again generated 100 data sets with 250 data points in each condition. Then we allowed DMAT to find the best parameter estimates, imposing

two different models. In the first model, we allowed for no parameter changes across conditions:

$$\mathbf{P_1} = \{\mathbf{1_C} \times \mathbf{a}^*, \mathbf{1_C} \times \mathbf{T_{er}}^*, \mathbf{1_C} \times \eta^*, \mathbf{1_C} \times \mathbf{z}^*, \mathbf{1_C} \times \mathbf{s_z}^*, \mathbf{1_C} \times \mathbf{s_t}^*, \mathbf{1_C} \times \mathbf{v}^*\}.$$

In the second model, we allowed drift rates to vary across conditions:

$$\mathbf{P_2} = \{\mathbf{1_C} \times \mathbf{a}^*, \mathbf{1_C} \times \mathbf{T_{er}}^*, \mathbf{1_C} \times \eta^*, \mathbf{1_C} \times \mathbf{z}^*, \mathbf{1_C} \times \mathbf{s_z}^*, \mathbf{1_C} \times \mathbf{s_t}^*, \mathbf{I_{C \times C}} \times \mathbf{v}^*\}.$$

It can be seen that the first model is nested in the second. Therefore, the first is the reduced model and the second the full model.

Then, we computed the test statistic $\Delta\Lambda = \Lambda_{\mathbf{P_1}} - \Lambda_{\mathbf{P_2}} \sim \chi^2_{df=3}$ and its significance level. If the likelihood ratio test has sufficient power, a large proportion of these statistics should be larger than the critical chi-square value. Failing to reject a null hypothesis that is truly false, on the other hand, would be a Type II error. In fact, as it turns out, in 99% of these cases did a significant result emerge at all significance levels (down to $\alpha = 10^{-6}$). In only four cases (once in parameter set B and thrice in C) was the null hypothesis not rejected at the $\alpha = .01$ level. It can be concluded that, at least for these parameter sets and this sample size, the likelihood ratio test has very high power.

1.7.4.2 Type I error

However, the previous result begs the question of selectivity: Is it possible that the analysis would yield significant results, even where none were present? To test this possibility, we repeated the same kind of simulations, but now we changed the true parameters such that all drift rates remained constant across conditions. The drift rates were all equal to -.25, -.15, -.05, .05, .15, and .25 for parameter sets A through F, respectively. If the likeli-

hood ratio test statistic follows a chi-square distribution, we would expect the p-values resulting from this test to follow a uniform distribution under the null hypothesis. Figure 1.3 shows the empirical cumulative distribution functions of p-values for each of the six parameter sets together with the straight line which is the theoretical CDF of a uniform random variable. The empirical distribution of p-values deviates from the theoretical but not much. The deviation indicates that the likelihood ratio test is slightly too liberal, and that it may be prudent to test at more restrictive significance levels in practical settings.

1.7.4.3 Power analysis 2

Following up on the previous simulations, we set up a new series to determine how well DMAT is able to detect small differences in parameters. To that end, we simulated data sets with 250 data points in each of two conditions. In both conditions, all parameters were equal (and taken from parameter set A in Table 1.1), except for drift rate. Drift rate was always zero in the first condition, and in the second condition it was either also 0, or .02, .04, .06, .08, or .10. With each of those values for the second drift rate, we generated 10000 data sets and allowed DMAT to recover the parameter estimates, once with a model allowing no differences across the two conditions and once allowing drift rate to differ between the conditions. Then we calculated the likelihood ratio test statistic and the associated p-values (found from a χ^2-distribution with one degree of freedom). Figure 1.4 shows the proportion of rejected null hypotheses as a function of the threshold value α. There it can be seen that if the true drift rate in the second condition is 0.1, there are a lot of rejections of $\mathcal{H}_0$, even with very small values for α. Table 1.5 shows the proportion of rejections of $\mathcal{H}_0$ for common values of α. When $\alpha = .05$, a drift rate of 0.1 is detected more

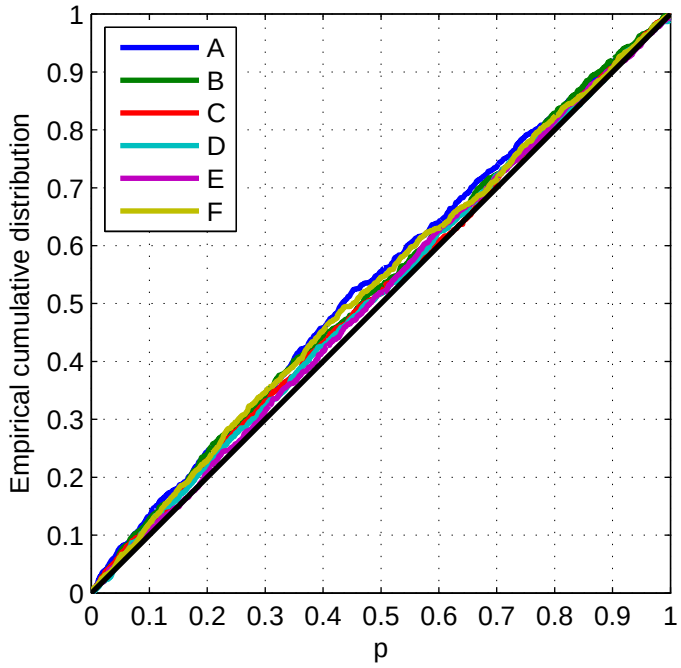


Figure 1.3: *The selectivity of the DMAT algorithm is illustrated. Different lines indicate different parameter sets. The full black line on the diagonal is the expected cumulative distribution function of the p-values (a uniform CDF). Lines above the diagonal indicate a liberal decision, while lines under it indicate a conservative decision. In general, DMAT produces CDFs that are close to the uniform CDF.*

than 96.3% of the time, but at a 6.1% risk of getting a "false alarm". With $\alpha = 10^{-6}$, there are no false alarms, but the test is much less powerful, detecting a drift rate difference of 0.1 in only 18.5% of cases.

1.8 Applications

To demonstrate the application of the methods described in this paper, we use two data sets containing both accuracy and reaction time data (Expe-

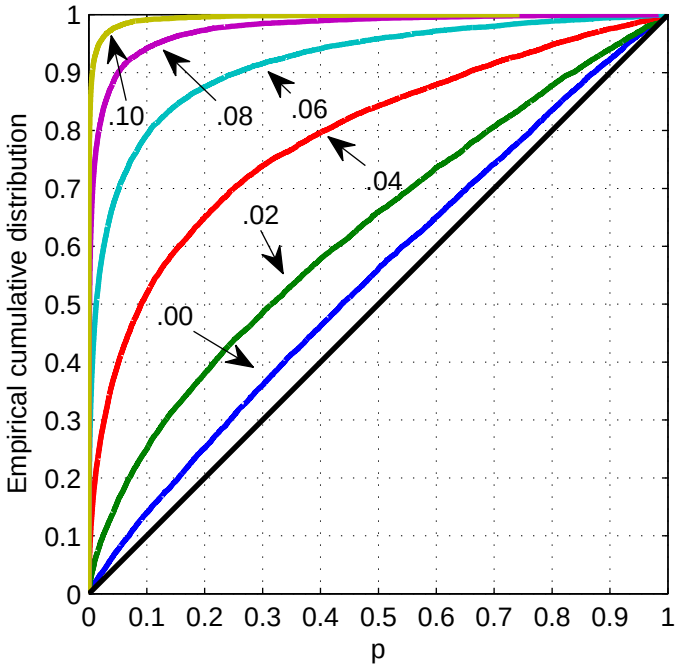


Figure 1.4: *The sensitivity of DMAT is illustrated. Different lines indicate data sets with larger or smaller simulated differences in drift rate. As the simulated difference increases, the CDF of the p-value departs from that of the uniform.*

Table 1.5: *Proportion of rejected null hypotheses for different levels of the α criterion, and with different 'real' effects in the data. Real differences in drift rate of 0.1 are detected in 96.28% of cases at the .05 significance level, but at that α-level there are also 6.07% false alarms.*

	Simulated difference in drift rate					
α	0	0.02	0.04	0.06	0.08	0.1
.05	.0607	.1454	.3778	.6649	.8769	.9628
.01	.0151	.0503	.1848	.4452	.7297	.9062
.0001	.0002	.0023	.0153	.0819	.2586	.5388
.00001	.0000	.0009	.0044	.0285	.1244	.3360
.000001	.0000	.0005	.0007	.0091	.0536	.1848

riment 3 in Vandekerckhove, Panis, & Wagemans, 2007, and Experiment 1 from Ratcliff & Rouder, 1998; used with permission). For substantive details on the studies, the interested reader is referred to Vandekerckhove et al. (2007) and Ratcliff and Rouder (1998).

1.8.1 Example 1: An incomplete factorial ANOVA design

The experiment by Vandekerckhove et al. (2007) is in the domain of visual shape perception and change detection in particular. The basic effect of interest is that if observers are shown a succession of two 2D shapes which are different in only one vertex (an angle or a curvature extreme), this difference is easier to detect if it is adding or removing a concavity than if it is adding or removing a convexity (Barenholtz, Cohen, Feldman, & Singh, 2003). The substantive research question in this experiment is: Does the effect occur when the change is not adding or removing a new vertex, but increasing or decreasing an existing one? The paradigm is a two-interval forced choice task.

In the experiment, three variables are manipulated: (a) change: was there any difference between the two shapes? (b) quality: did the number of vertices change? (c) type: if there was a change, was it in a concavity (curvature with negative sign) or in a convexity (positive sign)? As is obvious from variables (b) and (c), this is not a fully crossed design (properties of the change cannot be manipulated if there was none; as a result each 'change' condition had 80 data points, but each 'no-change' condition had 320). Table 1.6 lists all the conditions between which we would want to differentiate. Because the manipulations are all intended to affect the quality of the stimulus, we expect changes in drift rate, but not in any other variable. Writing the design as we do in Table 1.6 simplifies construction of a design matrix: The complete design matrix is simply the

Table 1.6: *Design of Experiment 3 in Vandekerckhove et al. (2007).*

Condition	Change	Quality	Type
1	1	1	-1
2	1	1	1
3	1	-1	-1
4	1	-1	1
5	-1	0	0

last three columns in the table, plus one column with ones for an intercept.

The goal of this experiment (and thus of the data-analysis) is twofold. Primarily, it was to find out whether the type variable contributes anything above and beyond the quality variable. Additionally, if type has an effect, we would want to know whether it is independent of quality (i.e., is there an interaction?). To this end, we defined a series of five models, each an extended version of the former. In Model 1, all parameters were constant across conditions:

$$\mathbf{P_1} = \{\mathbf{1_5} \times \mathbf{a}^*, \mathbf{1_5} \times \mathbf{T_{er}}^*, \mathbf{1_5} \times \eta^*, \mathbf{1_5} \times \mathbf{z}^*, \mathbf{1_5} \times \mathbf{s_z}^*, \mathbf{1_5} \times \mathbf{s_t}^*, \mathbf{1_5} \times \mathbf{v}^*\}\,.$$

In Model 2, we let drift rates vary according to the design of the experiment, without the type variable, and in Model 3, we let drift rates vary according to the complete design of the experiment. These design matrices were

$$\mathbf{D_{v2}} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & 1 & -1 \\ 1 & -1 & 0 \end{bmatrix}$$

Table 1.7: *Fit statistics from the model queue for one participant (Example 1).*

Model	Λ	df	$\Delta\Lambda$	Δdf	p	$AICc$	BIC
1	2414.32	9				2433	2472
2	2401.95	11	12.37	2	.0021	2424	2472
3	2387.49	12	14.46	1	.0001	2412	2464
4	2387.13	13	0.37	1	.5435	2414	2470
5	2373.70	37	13.42	24	.9587	2453	2609

and

$$\mathbf{D_{v3}} = \begin{bmatrix} 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & 1 \\ 1 & -1 & 0 & 0 \end{bmatrix},$$

respectively. In Model 4, we allowed drift rates to vary freely across conditions:

$$\mathbf{D_{v4}} = \mathbf{I_{C\times C}},$$

and finally, in Model 5, we allowed all diffusion parameters to vary freely across all five conditions:

$$\mathbf{P_5} = \{\mathbf{I_5}\times\mathbf{a}^*, \mathbf{I_5}\times\mathbf{T_{er}}^*, \mathbf{I_5}\times\eta^*, \mathbf{I_5}\times\mathbf{z}^*, \mathbf{I_5}\times\mathbf{s_z}^*, \mathbf{I_5}\times\mathbf{s_t}^*, \mathbf{I_5}\times\mathbf{v}^*\}.$$

Note that Model 1 has 9 free parameters, Model 2 has 11, Model 3 has 12, and Model 4 has 13, while Model 5 has 37 (we keep the mixture model parameters π and γ constant in all models). Note also that each model is nested in the next. We report the analysis for one participant in the experiment. Table 1.7 displays the fit statistics of each model together with the deviance Λ and the likelihood ratio test statistic $\Delta\Lambda$ for two consecutive models (together with the appropriate number of degrees of freedom).

From the table, we can conclude that the increase in model freedom from Model 1 to Model 2 was a success—the badness-of-fit measure significantly decreases ($p = .0021$). Going from Model 2 to Model 3 (adding the variable type) also significantly improved model fit ($p = .0001$). However, going to Model 4 (allowing deviation from the experimental design) was not a worthwhile step: allowing drift rates to vary freely does not significantly improve the fit of the model ($p = .5435$). Finally, while the step from Model 4 to Model 5 decreased the chi-square value, this decrease was again not significant when we take into consideration that 24 parameters had been added to the model ($p = .9587$). Considering this, and inspecting the $AICc$ and BIC values, Model 3 earns our preference. Furthermore, the estimated basic drift parameters were $\hat{\mathbf{v}}^* = \begin{bmatrix} 0.065 & -0.039 & -0.050 & -0.077 \end{bmatrix}^{\mathrm{T}}$. The standard errors of these basic parameters, obtained from inverting the Hessian matrix, are 0.012, 0.010, 0.014, and 0.015, respectively. Thus, the Wald test for $\mathcal{H}_0 : v_4 = 0$ (the effect of type) is $Z = \frac{\hat{v}_4 - 0}{\sigma_{v_4}} = \frac{-0.077}{0.015} = -5.10$, $p < 10^{-6}$. The values of the other parameter estimates and their standard errors of estimation under Model 3 are given in Table 1.8. In the table, it can be seen that $\hat{s_z}^*$ has converged to a boundary value (zero), so we might consider removing it from the model in a second stage of the data-analysis (but doing so does not dramatically affect the other parameter estimates).

Regarding outlier treatment, the parameter π converged to a boundary value (one) as well, meaning that there appear to be very few contaminants in this data set. As a result, the γ parameter becomes unidentified and we might later also consider dropping both π and γ from the model. (When we did this, the outcome of the analysis still remained the same.) The cut-off value estimated by the EWMA method was 373.3 ms, meaning that about 9% of data points (in this case, evenly distributed across conditions) were censored.

As for the substantive hypotheses, the finding that adding the type

Table 1.8: *Recovered basic parameters and associated standard errors (SEs) for Example 1, under Model 3, for one participant. Wald tests are for* $\mathcal{H}_0$*:* $\theta = 0$*.*

Parameter	Estimate	SE	Z	p
$a^*_{(1)}$	0.1747	0.0105	16.63	$<.0001^\dagger$
$T_{er}{}^*_{(1)}$	0.3406	0.0054	62.74	$<.0001^\dagger$
$\eta^*_{(1)}$	0.2542	0.0346	7.36	$<.0001^\dagger$
$z^*_{(1)}$	0.0888	0.0053	46.88	$<.0001^\dagger$
$s_{z}{}^*_{(1)}$	0.0807	0.0535	1.50	$.0607^\dagger$
$s_{t}{}^*_{(1)}$	0.0000	0.0318	0.00	$.5000^\dagger$
$v^*_{(1)}$	0.0649	0.0117	5.56	$<.0001$
$v^*_{(2)}$	-0.0385	0.0104	-3.71	.0002
$v^*_{(3)}$	-0.0501	0.0135	-3.72	.0002
$v^*_{(4)}$	-0.0769	0.0151	-5.10	$<.0001$

† Using Stram and Lee's (1994) corrected reference distribution $Z^2 \sim .5\chi^2_0 + .5\chi^2_1$.

variable to the analysis significantly increased model fitness indicates that it influences drift rate, above and beyond the effect of the quality variable. Additionally, allowing for a more complex pattern than the experimental design (e.g., with interactions) did not lead to a better fit, indicating that for this participant, the experimental variables did not interact.

The reported results were not identical for all participants in the experiment. Five out of ten showed the pattern discussed above. In two other cases, Model 3 did not provide a significantly better fit than Model 2, indicating no significant effect of type. In a further two cases, Model 4 did provide a significant improvement relative to Model 3, indicating interactions between experimental variables. In one final case, Model 4 performed better than Model 3, but Model 5 also performed better than Model 4, indicating influences on other parameters beside drift rate (with this participant, there was a large across-condition difference in boundary separation).

1.8.2 Example 2: A linear regression design

The first experiment in (Ratcliff & Rouder, 1998) is a brightness discrimination experiment. There were two manipulations of interest. First, there was a speed-accuracy instruction (participants were either instructed to be fast or to be accurate) and second, there was a manipulation of brightness, which increased with equal steps over 33 levels. However, in order to ensure a sufficient number of trials in each cell, we collapsed the five darkest and five brightest levels into a single level each, leaving 25 levels of the brightness variable and the number of trials varying across conditions from 61 to more than 200, with an average of about 150. The two variables were completely crossed in a 2×25 design, yielding 50 conditions (conditions 1-25 have an 'accuracy' instruction and 26-50 have a 'speed' instruction). The task was a 2AFC procedure, whereby each subject was shown a stimulus and had to judge whether this stimulus was drawn from a 'bright' distribution or from a 'dark' distribution (the two distributions overlapped significantly, so subjects could not be highly accurate). Feedback was given after each trial.

From the manipulations, we can expect two things. Firstly, we expect that the speed-accuracy instruction will have an effect on boundary separation. Secondly, we expect that as brightness of the stimulus increases, the drift towards the 'bright' response increases. Note that for this analysis, we change the interpretation of the model's upper and lower boundaries. We will now say that a hit on the upper boundary leads to a 'bright' response, and on the lower to a 'dark' response.[1] The drift rate is hence no longer a measure of ability to respond correctly, but of a tendency to respond

[1]Changing the response coding is in general only necessary if there is a very high proportion of correct - or error - responses, so that one of the marginal distributions of the model is represented by only a few data points. Here there is no substantive or statistical reason that compels us to do this; we merely take this approach for illustrative reasons.

'bright' (and a negative drift rate can now reasonably occur, indicating a tendency to respond 'dark'). Because the responses are not classified as correct or wrong, the assumption for the EWMA method that guesses are equally distributed across responses (i.e., that 50% of the guesses are 'bright') no longer holds. Accordingly, we switched off the EWMA preprocessing. For the same reason, the 'fast guesses' component of the mixture model is no longer a valid representation, thus we will assume that the weight γ is equal to zero.

To perform the analysis, we defined a series of three models, each a more complex version of the former. In all models, we defined that there should be two different levels of the parameters a, z, and s_z: one for the conditions with accuracy instruction and one for those with speed instruction. To do this, we constructed the following design matrix for these parameters: $\mathbf{D}_a = \mathbf{D}_z = \mathbf{D}_{s_z} = \begin{bmatrix} \mathbf{1_{25}} & \mathbf{0_{25}} \\ \mathbf{0_{25}} & \mathbf{1_{25}} \end{bmatrix}$, which has two columns with 25 ones and 25 zeros each. Additionally, in Model 1 we will allow v to evolve linearly with the brightness manipulation, while allowing different regression slopes and intercepts for different speed-accuracy instructions: $\mathbf{D}_v = \begin{bmatrix} \mathbf{1_{25}} & \mathbf{L} & \mathbf{0_{25}} & \mathbf{0_{25}} \\ \mathbf{0_{25}} & \mathbf{0_{25}} & \mathbf{1_{25}} & \mathbf{L} \end{bmatrix}$, where $\mathbf{L} = \begin{bmatrix} 3 & 6 & 7 & \dots & 27 & 28 & 31 \end{bmatrix}^{\mathrm{T}}$ represents the 25 brightness levels (with the first and last values adapted to reflect the average of the five groups that were pooled there). The other design matrices impose the requirement that there is no change across conditions: $\mathbf{P_1} = \{\mathbf{D}_a \times \mathbf{a}^*, \mathbf{1_{50}} \times \mathbf{T_{er}}^*, \mathbf{1_{50}} \times \eta^*, \mathbf{D}_z \times \mathbf{z}^*, \mathbf{D}_{s_z} \times \mathbf{s_z}^*, \mathbf{1_{50}} \times \mathbf{s_t}^*, \mathbf{D}_v \times \mathbf{v}^*\}$.

However, the restriction that drift rates should increase linearly with the brightness manipulation is hardly tenable, both on theoretical grounds (because performance has upper and lower bounds) and due to opportunistic inspection of Ratcliff and Rouder's (1998) results. In fact, in

Table 1.9: *Fit statistics from the model queue for one participant (Example 2).*

Model	Λ	df	$\Delta\Lambda$	Δdf	p	$AICc$	BIC
1	23516.8	14				23545	23642
2	23213.64	20	303.14	6	<.0001	23254	23393
3	23153.26	60	60.38	40	.0202	23274	23692
4	23086.12	351	67.14	291	.9999	23821	26236

their article, drift rate increases with brightness like a sigmoid function. Thus, in Model 2, we add a quadratic, cubic and quartic component to the design, to mimic an S-shaped function. Now, $\mathbf{D}_v = \begin{bmatrix} \mathbf{1_{25}} & \mathbf{L} & \mathbf{L}^2 & \mathbf{L}^3 & \mathbf{L}^4 & \mathbf{0_{25}} & \mathbf{0_{25}} & \mathbf{0_{25}} & \mathbf{0_{25}} & \mathbf{0_{25}} \\ \mathbf{0_{25}} & \mathbf{0_{25}} & \mathbf{0_{25}} & \mathbf{0_{25}} & \mathbf{0_{25}} & \mathbf{1_{25}} & \mathbf{L} & \mathbf{L}^2 & \mathbf{L}^3 & \mathbf{L}^4 \end{bmatrix}$, where the exponents indicate the element-wise power function (i.e., each element of the vector $\mathbf{L}$ is taken to that power). The other design matrices are the same as in Model 1. Note that, for numerical reasons, we rescaled each column of $\mathbf{D}_v$ such that the values were in the range (00.5).

In Model 3, we allowed drift rates to vary freely: $\mathbf{P_3} = \{\mathbf{D}_a \times \mathbf{a}^*, \mathbf{1_{50}} \times \mathbf{T_{er}}^*, \mathbf{1_{50}} \times \eta^*, \mathbf{D}_z \times \mathbf{z}^*, \mathbf{D}_{s_z} \times \mathbf{s_z}^*, \mathbf{1_{50}} \times \mathbf{s_t}^*, \mathbf{I_{50\times 50}} \times \mathbf{v}^*\}$.

Finally, in Model 4, all diffusion parameters can vary freely across conditions.

Models 1 to 4 have 1420, 60, and 351 free parameters, respectively, and each model is nested in the next. The models were fitted to the data of one participant. Table 1.9 displays the fit statistics of each model, and shows the Λ and $\Delta\Lambda$-statistics with their degrees of freedom. Figure 1.5 shows the drift rates according to models 1, 2, and 3 as a function of brightness.

As can be seen from the table, Model 2 outperforms Model 1 greatly, indicating deviations from linearity (as is obvious from the figure). Moreover, Model 3 performs slightly better than Model 2. Finally, Model 4 does not perform significantly better than Model 3, indicating that it is not necessary to free all parameters in the model across conditions. The

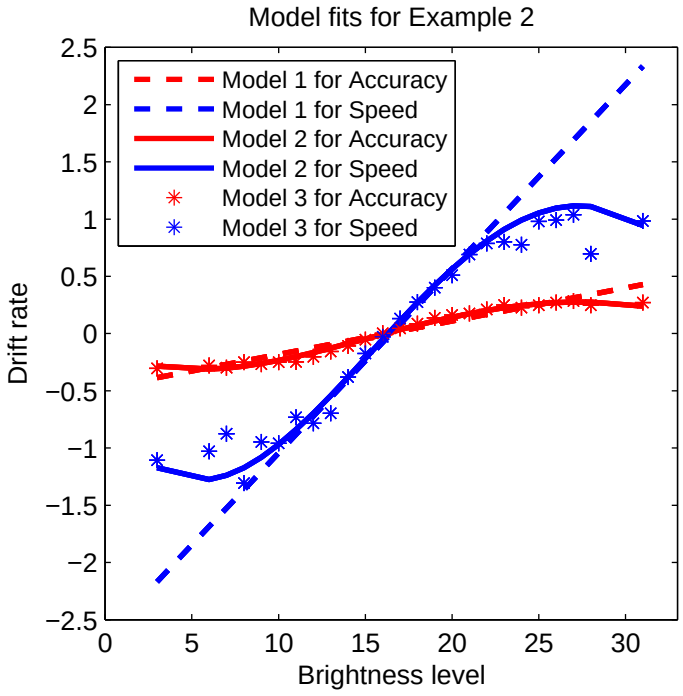


Figure 1.5: *Drift rates of one participant in Experiment 1 of Ratcliff and Rouder (1998). Drifts recovered by Model 1 are shown as dashed lines, with the steeper line indicating the Speed condition. Drifts from Model 2 are full curves, and drifts from Model 3 are stars. As can be seen, Model 1 provides a poor fit, while Model 2 is much closer to the separate drift rates, though with still some deviation left.*

$AICc$ and BIC statistics, in Table 1.9, show a preference for Model 2, where a polynomial regression was imposed on the drift rates. In this case, we would opt for Model 2, since the both information criteria point in that direction and the likelihood ratio test does not give convincing evidence against Model 2.

The recovered basic parameters and their standard errors of estimation under Model 2 are given in Table 1.10. Unlike in Example 1, here all parameters significantly deviate from zero (or from 1, in the case of π).

Table 1.10: *Recovered basic parameters and associated standard errors (SEs) for Example 2, under Model 2, for one participant. Wald tests are for unless indicated otherwise. Basic drift parameters (1), (2), and (3) refer to the accuracy condition, and (4), (5), and (6) refer to the speed condition. Other parameters indexed with a (1) apply to the accuracy condition and with a (2) to the 'speed' condition.*

Parameter	Estimate	SE	Z	p
$a^*_{(1)}$	0.1688	0.0016	103.50	$<.0001^\dagger$
$a^*_{(2)}$	0.0436	0.0000	$>10^6$	$<.0001^\dagger$
$T_{er}{}^*_{(1)}$	0.3065	0.0008	369.61	$<.0001^\dagger$
$\eta^*_{(1)}$	0.0252	0.0085	2.95	$.0016^\dagger$
$z^*_{(1)}$	0.0821	0.0012	71.13	$<.0001^\dagger$
$z^*_{(2)}$	0.0218	0.0000	$>10^6$	$<.0001^\dagger$
$s_{z}{}^*_{(1)}$	0.0476	0.0078	6.12	$<.0001^\dagger$
$s_{z}{}^*_{(2)}$	0.0426	0.0000	$>10^6$	$<.0001^\dagger$
$s_{t}{}^*_{(1)}$	0.1427	0.0023	62.27	$<.0001^\dagger$
$v^*_{(1)}$	-0.5892	0.0155	-37.96	<.0001
$v^*_{(2)}$	-3.9174	0.2829	-13.85	<.0001
$v^*_{(3)}$	0.8681	0.0336	25.86	<.0001
$v^*_{(4)}$	6.4215	0.6207	10.35	<.0001
$v^*_{(5)}$	-0.1671	0.0188	-8.90	<.0001
$v^*_{(6)}$	-2.0014	0.3373	-5.93	<.0001
$\pi^*_{(1)}$	0.9582	0.0041	$-10.28^\ddagger$	$<.0001^\dagger$

† Using Stram and Lee's (1994) corrected reference distribution $Z^2 \sim .5\chi^2_0 + .5\chi^2_1$. ‡ Testing $\mathcal{H}_0$: $\pi^*_{(1)} = 1$.

With $\hat{\pi} \approx .9606$ about 4% of the data (across conditions) are estimated to be contaminants.

For the two other participants, $AICc$ and BIC values did not agree, but the pattern of significance between models was identical to that in Table 1.9.

1.9 Conclusion

In the present paper, we investigated and enhanced the practical applicability of the diffusion model for reaction time and accuracy data and explored several avenues of improvement. We suggested the use of design matrices in order to regress diffusion model parameters onto covariates from an experiment and discussed the use of the likelihood ratio statistic for statistical inference and model selection. With this statistical framework to complement diffusion modeling, the simultaneous analysis of reaction time and accuracy data is moved closer to the realm of well-known statistical procedures such as ANOVA and multiple linear regression. We presented simulation studies where the small-sample behavior of the likelihood ratio statistic was found suitable. We also presented outlier treatment methods and showed that they perform well. Furthermore, we have implemented these methods in a freely available software tool (DMAT; Vandekerckhove & Tuerlinckx, 2008).

Some further extensions of the Ratcliff diffusion model now present themselves. A first extension that readily flows from the present study is to implement other (non-linear) constraints on model parameters than the ones permitted by the design matrix method. For example, in our second application we imitated a sigmoid function in an ad-hoc way (with a polynomial of a high degree), whereas a better solution would be to simply use a non-linear link function (such as a logit or probit link). A second possibility for advancement is to move from classical frequentist parameter estimation to a Bayesian framework (as in Lee, Fuss, & Navarro, 2007). Finally, further research is needed to investigate the statistical qualities of Quantile Probability Products estimators (Brown & Heathcote, 2003; Heathcote & Brown, 2004; Heathcote, Brown, & Mewhort, 2002; Speckman & Rouder, 2004), since this estimation method seems preferable

if the range of the reaction time distribution is unknown, but its qualities for statistical inference are not yet explored.

References

Barenholtz, E., Cohen, E., Feldman, J., & Singh, M. (2003). Detection of change in shape: an advantage for concavities. *Cognition*, *89*, 1–9.

Bishop, Y., Fienberg, S., & Holland, P. (1975). *Discrete multivariate analysis: Theory and practice.* Cambridge: MIT Press.

Brown, S., & Heathcote, A. (2003). QMLE: Fast, robust and efficient estimation of distribution functions based on quantiles. *Behavior Research Methods, Instruments & Computers*, *35*, 485–492.

Chandra, M. (2001). *Statistical quality control.* New York: CRC Press.

De Boeck, P., & Wilson, M. (2004). *Explanatory item response models: A generalized linear and nonlinear approach.* New York: Springer.

Donders, F. C. (1969). On the speed of mental processes. *Acta Psychologica*, *30*, 412–431.

Heathcote, A., & Brown, S. (2004). Reply to speckman and rouder: A theoretical basis for QML. *Psychonomic Bulletin & Review*, *11*, 577.

Heathcote, A., Brown, S., & Mewhort, D. (2002). Quantile maximum likelihood estimation of response time distributions. *Psychonomic Bulletin & Review*, *9*, 394–401.

Hurvich, C. M., & Tsai, C.-L. (1989). Regression and time series model selection in small samples. *Biometrika*, *76*, 297–307.

Laming, D. R. J. (1968). *Information theory of choice-reaction times.* London, UK: Academic Press.

Lee, M. D., Fuss, I. G., & Navarro, D. J. (2007). A Bayesian approach to diffusion models of decision-making and response time. In B. Schölkopf, J. Platt, & T. Hoffman (Eds.), *Advances in neural*

information processing systems 19 (pp. 809–816). Cambridge, MA: MIT Press.

Link, S. W., & Heath, R. A. (1975). A sequential theory of psychological discrimination. *Psychometrika*, *40*, 77–105.

Littell, R. C., Stroup, W., & Freund, R. J. (2002). *SAS system for linear models, fourth edition.* Cary, NC: SAS Institute Inc.

Luce, R. D. (1986). *Response times.* New York: Oxford University Press.

Nelder, R., J. A. Mead. (1965). A simplex method for function minimization. *Computer Journal*, *7*, 308–313.

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, *85*, 59–108.

Ratcliff, R. (1981). A theory of order relations in perceptual matching. *Psychological Review*, *88*, 552–572.

Ratcliff, R. (1988). Continuous versus discrete information processing: Modeling the accumulation of partial information. *Psychological Review*, *95*, 238–255.

Ratcliff, R. (1993). Methods for dealing with reaction time outliers. *Psychological Bulletin*, *114*, 510–532.

Ratcliff, R. (2002). A diffusion model account of response time and accuracy in a brightness discrimination task: Fitting real data and failing to fit fake but plausible data. *Psychonomic Bulletin & Review*, *9*, 278–291.

Ratcliff, R., Gomez, P., & McKoon, G. (2004). Diffusion model account of lexical decision. *Psychological Review*, *111*, 159–182.

Ratcliff, R., & Rouder, J. N. (1998). Modeling response times for two-choice decisions. *Psychological Science*, *9*, 347–356.

Ratcliff, R., & Rouder, J. N. (2000). A diffusion model account of masking in two-choice letter identification. *Journal of Experimental Psychology: Human Perception and Performance*, *26*, 127–140.

Ratcliff, R., Thapar, A., & McKoon, G. (2001). The effects of aging on reaction time in a signal detection task. *Psychology and Aging*, *16*, 323–341.

Ratcliff, R., & Tuerlinckx, F. (2002). Estimating parameters of the diffusion model: Approaches to dealing with contaminant reaction times and parameter variability. *Psychonomic Bulletin & Review*, *9*, 438–481.

Ratcliff, R., Van Zandt, T., & McKoon, G. (1999). Connectionist and diffusion models of reaction time. *Psychological Review*, *102*, 261–300.

Read, T., & Cressie, N. (1988). *Goodness-of-fit statistics for discrete multivariate data*. New York: Springer.

Roberts, S. (1959). Control chart tests based on geometric moving averages. *Technometrics*, *1*, 239–250.

Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, *6*, 461–464.

Schwarz, W. (2001). The ex-wald distribution as a descriptive model of response times. *Behavior Research Methods, Instruments, & Computers*, *33*, 457–469.

Speckman, P. L., & Rouder, J. N. (2004). A comment on heathcote, brown, and mewhorts qmle method for response time distributions. *Psychonomic Bulletin & Review*, *11*, 574–576.

Stram, D. O., & Lee, J. W. (1994). Variance components testing in the longitudinal mixed effects model. *Biometrics*, *50*, 1171–1177.

Strayer, D., & Kramer, A. (1994). Strategies and automaticity: I. *Basic findings and conceptual framework. Journal of Experimental Psychology: Learning, Memory, & Cognition*, *20*, 318–341.

Thapar, A., Ratcliff, R., & McKoon, G. (2003). A diffusion model analysis of the effects of aging on letter discrimination. *Psychology & Aging*,

18, 415–429.

Tuerlinckx, F. (2004). The efficient computation of the distribution function of the diffusion process. *Behavior Research Methods, Instruments, & Computers*, *36*, 702-716.

Tuerlinckx, F., Maris, E., Ratcliff, R., & De Boeck, P. (2001). A comparison of four methods for simulating the diffusion process. *Behavior Research Methods, Instruments, & Computers*, *33*, 443–456.

Ulrich, R., & Miller, J. (1994). Effects of truncation on reaction time analysis. *Journal of Experimental Psychology: General*, *123*, 34–80.

Vandekerckhove, J., Panis, S., & Wagemans, J. (2007). The concavity effect is a compound of local and global effects. *Perception & Psychophysics*, *69*, 1253–1260.

Vandekerckhove, J., & Tuerlinckx, F. (2007). The Diffusion Model Analysis Toolbox. [Software and manual] [Computer software manual]. (Retrieved from `http://ppw.kuleuven.be/okp/dmatoolbox`)

Vandekerckhove, J., & Tuerlinckx, F. (2008). Diffusion model analysis with MATLAB: A DMAT primer. *Behavior Research Methods*, *40*, 61–72.

Voss, A., Rothermund, K., & Voss, J. (2004). Interpreting the parameters of the diffusion model: An empirical validation. *Memory & Cognition*, *32*, 1206–1220.

Voss, A., & Voss, J. (2007). Fast-dm: A free program for efficient diffusion model analysis. *Behavior Research Methods*, *39*, 767–782.

Voss, A., & Voss, J. (2008). A fast numerical algorithm for the estimation of diffusion-model parameters. *Journal of Mathematical Psychology*, *52*, 1–9.

Wagenmakers, E.-J., Ratcliff, R., Gomez, P., & McKoon, G. (2008). A diffusion model account of criterion shifts in the lexical decision task. *Journal of Memory and Language*, *58*, 140–159.

Wagenmakers, E.-J., van der Maas, H. J. L., & Grasman, R. P. P. P. (2007). An EZ-diffusion model for response time and accuracy. *Psychonomic Bulletin & Review*, *14*, 3–22.

1.A Outlier Treatment Methods

Exponentially weighted moving average filter

The exponentially weighted moving average method (EWMA; Chandra, 2001; Roberts, 1959) is a statistical quality control method that can detect shifts in performance as reaction times increase. A cut-off threshold is set where the performance is judged to be above chance level.

The first step in the application of the method is sorting the reaction times from short to long. In effect, we will then look at our data set as if it described a binary process that unfolds (and changes) over time. As time progresses (i.e., RT increases), the process will start to shift away from its 'control state' (with 50% accuracy) and tend toward a biased process (with accuracy $> 50\%$). The control process describes our expectation regarding fast guesses, which is straightforward: Guesses are draws from a Bernoulli process at chance level. Formally, if the s^{th} observation (that is, the response $X_{(s)}$, corresponding to the s^{th} sorted reaction time $T_{(s)}$) is a guess, then $X_{(s)} \sim \text{Bernoulli}(0.5)$. The control process should be a credible representation of fast guesses, otherwise this method will not work. However, usually trials in an experiment are counterbalanced and randomized in such a way that participants cannot significantly exceed chance level accuracy without paying proper attention to the stimuli presented. If measures have been taken to avoid participants from being 'cued' to a correct (or error) response even when guessing, then it is reasonable to expect accuracy to be around 50% for fast guesses in a two-alternative

forced choice task.

To determine the minimal response time at which the system no longer follows this control process, we take the reaction times from all conditions (all RTs still sorted fast to slow), and then analyze their corresponding responses. Of these responses, we iteratively compute the EWMA statistic $c_s = \lambda x_s + (1 - \lambda)c_{s-1}$, where $x_s = 1$ if the response corresponding to the s^{th} sorted reaction time was correct and 0 otherwise, and $\lambda \in (01]$ is a weight parameter which controls how many of the last observations are used. If λ is 1 only the s^{th} observation is used and if approaches 0, all observations from the first to the s^{th} are weighted equally. We will then, at each iteration, calculate the upper control limit (UCL) of this process, and check if the EWMA statistic c_s exceeds this value.

In practice, some constants need to be defined. The first is the in-control mean of the process, which in this context represents the expected average performance of a fast guess. We denote this parameter c_0, and initialize it to 0.5. Second is the in-control standard deviation σ_0 (standard deviation of X), which is also equal to 0.5 (this follows from the properties of the Bernoulli distribution). A third constant for EWMA is the weight parameter λ. We choose $\lambda = 0.01$, thereby accounting for many previous data points. The final constant is the width of the control limits (in standard deviations). To ensure a sensitive test, we set L to 1.5 (a relatively low value).

Given these parameters, we now compute $c_s < \text{UCL}_s = c_0 + L\sigma_0\sqrt{\frac{\lambda}{2-\lambda}\left[1 - (1-\lambda)^{2s}\right]}$ and check if it is smaller than the upper control limit: . If this inequality is true, then the process is judged to be within the limits of the control model, and we label observation s as a 'fast guess'. When the upper control limit is exceeded, we decide that the probability of giving a correct response significantly exceeds 0.5 from this reaction time

on, and stop the iteration process. The reaction time at which the UCL was breached is then taken as the threshold, and all reaction times below it are censored.

The EWMA method is commonly illustrated with a control chart, which depicts the evolution of c_s as a function of increasing reaction time. Figure 1.6 shows an example control chart, with the EWMA statistic indicated by a full line, the control state by a dotted line and the control limits by a shaded region around the control state. This control chart is based on data that were generated from the parameters shown in Table 1.1 (set A), with 250 data points in each condition, and 5% fast outliers added to the 200-400ms domain, uniformly distributed and with 50% accuracy. The EWMA algorithm returns a cut-off value of 322ms, which is reasonable considering that the diffusion process with these parameters starts around 300ms, but there are contaminants between 200 and 400ms.

Mixture model approach

The CDF of the diffusion model extended with the mixture model approach is

$$\begin{aligned} F_{XT}(xt\theta) &= \pi \mathrm{Diff}(xt\theta) + (1-\pi)\gamma\frac{1}{2}\mathrm{U}(tT_{-}T_{+}) && (1.2)\\ &+ (1-\pi)(1-\gamma)\Pr(X=x|\theta)\mathrm{U}(tT_{-}T-+) && (1.3) \end{aligned}$$

where $\mathrm{U}(tAB)$ indicates the cumulative density function of a uniform distribution from A to B, evaluated at t. $\mathrm{Diff}(xt\theta)$ is the joint probability that the response equals x ($x = 0$ for an error and $x = 1$ for a correct response) and that the response is given at time t or before, under a Ratcliff diffusion model with parameter vector θ (thus, $\mathrm{Diff}(xt\theta) = \Pr(X = xT \leq t|\theta)$). The exact formula for this joint probability is provided in Tuerlinckx (2004; Equations 1, 2, and 3). Further, T_{-} and T_{+} are the minimum and maxi-

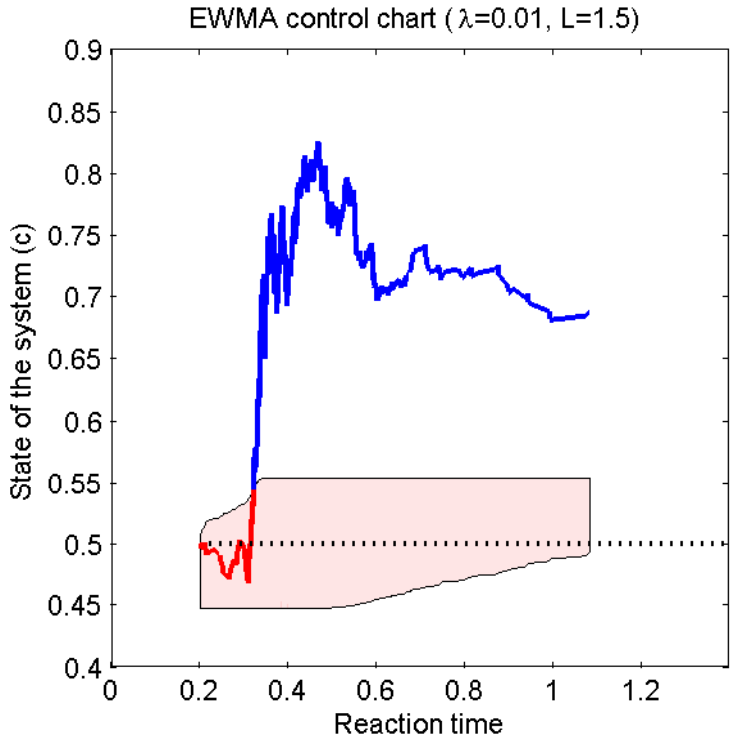


Figure 1.6: *An EWMA control chart showing guessing for reaction times lower than approximately 322ms. See text for details.*

mum of the assumed response time distributions for contaminants. Technically, T_- and T_+ are parameters, but in the remainder of this paper we will not treat them as such. They are not included in the parameter estimation routine, but are directly estimated with the observed minimum and maximum response times (for each condition and each participant), respectively.

1.B Minimizing the Multinomial Log-Likelihood Function

Loss function

DMAT uses a multinomial likelihood function (MLF), which expresses the likelihood of observing a certain proportion of responses in a given number of reaction time bins, and should therefore be maximized in order to find good parameter estimates.

To define B reaction time bins, we select $B-1$ monotonically increasing bin edges $q_1, \ldots, q_{B-1}$ and define $q_0 = 0$ and $q_B = +\infty$. The observed frequency in bin b, in condition c, for response x, is then simply $O_{cxb} = \sum_{j=1}^{n_{cb}} I\left(q_{b-1} < t_{cxj} \leq q_b\right)$, with n_{cx} being the number of data points with response x in condition c. $I(\cdot)$ is the indicator function (which takes the value 1 if its argument is true and 0 otherwise). The predicted (or expected) proportion of x responses in bin b of condition c equals $P_{cxb} = F_{XT}(xq_b\theta_c) - F_{XT}(xq_{b-1}\theta_c)$, where θ indicates the parameter vector for condition c, and F_{XT} is the CDF of the RDM (or of the extended Ratcliff diffusion model, see Equation 1.3).

The negative log of the MLF that needs to be minimized is then defined as:

$$\Lambda = -2(l) = -2\log\left(\prod_{c=1}^{C}\prod_{x=0}^{1}\prod_{b=1}^{B} P_{cxb}^{O_{cxb}}\right) \tag{1.4}$$

$$= -2\sum_{c=1}^{C}\sum_{x=0}^{1}\sum_{b=1}^{B} O_{cxb}\log\left(P_{cxb}\right). \tag{1.5}$$

We will hereafter refer to Equation (B1) as 'the' multinomial (log)likelihood function (MLF). During parameter estimation this will be the loss function we will be minimizing. An alternative to the MLF is the more common chi-square loss function as described by Ratcliff and

Tuerlinckx (2002). It is shown by Read and Cressie (1988) that both are intimately related. DMAT allows the user to choose between these two, but the MLF is the default option.

We give two final remarks about the loss function. First, in light of the recent discussion about the appropriateness of fixed versus percentile based boundaries to define the bins (see Heathcote et al., 2002; Speckman & Rouder, 2004; Brown & Heathcote, 2003; Heathcote & Brown, 2004), in DMAT we have left the choice to the user. To briefly summarize this discussion: A statistic computed using percentile based (data-dependent) bin edges is not a true chi-square statistic. The practical consequence of this is that the (p-value of the) test statistic $\Delta\Lambda$ cannot be used for inference. There is also the third option of using fixed bin edges calculated from the average percentile-based bins over a wide range of parameter values (and this is the default choice in DMAT). The fits reported in this paper are based on these realistic bin boundary values but as said before, the user can opt for another method.

A second remark is that some parameters in the model cannot take all possible real values and this may lead to numerical difficulties during the optimization. There are several possible strategies to avoid these problems, but in the estimation algorithm we simply define that $\Delta\Lambda = 10^{10}$ if the parameter set θ is outside the allowable parameter space. This penalty value ensures that the optimization algorithm will remain inside the parameter space.

Optimization

In this subsection, we discuss three aspects of our optimization procedure: the starting points, the algorithm and ways to avoid local minima.

First, we use a method of moments to produce a good initial estimate

of the parameters. If the user of DMAT did not provide the algorithm with a starting guess, DMAT uses the estimates from the EZ-diff model (Wagenmakers et al., 2007) to generate a plausible starting point for the first model in the queue. For subsequent models in a queue, DMAT uses the final estimates of the previous model as starting point (if necessary, a linear transformation or regression will be applied to ensure that this initial guess does not violate any restrictions of the present model). For the parameters of the RDM that are not represented in the EZ-diff model, we make the following (arbitrary) guesses: $\eta = 0.2$, $z = a/2$, $s_z = \frac{9}{20}a$, and $s_t = \frac{9}{10}T_{er}$.

Second, the algorithm we use to find the optimum of the loss function is the Nelder-Mead Simplex algorithm (NMS algorithm; Nelder, 1965), with a few adaptations. In our algorithm, we allow a single NMS run to proceed for 200 steps, after which the size of the simplex shape is reset to its original size. We do this because we have observed that (due to numerical issues) the optimizer sometimes converges in a local minimum. Resetting the simplex size allows the algorithm to escape from such local minima. Usually, the simplex size is reset three times, thus performing four runs with maximally 200 steps. When these are finished, we start a fifth, longer, NMS run with maximally 5000 iterations. Usually, however, the last NMS run converges before that. DMAT users can change the number of NMS runs, as well as the maximum number of iterations allowed. The final phase of the algorithm is a single quasi-Newton step, where the first and second derivatives of the objective are numerically approximated and used to find the local minimum near the point where the NMS run converged. This provides us with a numerical approximation to the Hessian matrix (the matrix of second derivatives) at the minimum, which is then used to verify that the solution point is in fact a minimum (the Hessian should be positive definite), and to calculate estimates of parameter standard errors.

Third, we incorporate another strategy for identifying and escaping suspected local minima. The fact that the algorithm has converged to a minimum is still no guarantee that we have in fact found optimal parameter estimates. A better parameter set might still exist in a region that our algorithm has not visited. This is a very difficult problem and it is not possible in general to give strong guarantees about the optimality of a set of estimated parameters. Local minima seem to exist near those boundaries of the parameter space where any of the variability parameters η, s_t, or s_z are zero (although it is possible that this is a true minimum for some data sets).

To avoid local minima, the algorithm performs a jump when the variance parameters are estimated to zero (s_z and s_t will be changed to half of their maximal value, and η will be, arbitrarily, reset to 0.2). After making this jump, the NMS is restarted. We have experienced that with this identify-and-jump strategy the algorithm often succeeds in locating a better point in the parameter space.

A MATLAB toolbox

CHAPTER 2

Diffusion Model Analysis with MATLAB: a DMAT primer

Abstract

The Ratcliff diffusion model has proved to be a useful tool in reaction time analysis. However, its use has been limited due to the practical difficulty of estimating the parameters. We present a software tool, the *Diffusion Model Analysis Toolbox* (DMAT) intended to make the Ratcliff diffusion model for reaction time and accuracy data more accessible to experimental psychologists. The tool takes the form of a MATLAB toolbox and can be freely downloaded from `http://ppw.kuleuven.be/okp/dmatoolbox`. Using the program does not require a background in mathematics, nor any advanced programming experience (but familiarity with MATLAB is useful). We demonstrate the basic use of DMAT with two examples.

2.1 Introduction

The Ratcliff diffusion model (RDM) has garnered significant attention in recent years as a model for the simultaneous analysis of reaction time and accuracy data. There are three main reasons for its popularity. Firstly, it can be applied in many different fields (Ratcliff, 1978, 1981, 1988, 2002; Ratcliff, Gomez, & McKoon, 2004; Ratcliff & Rouder, 1998, 2000; Ratcliff, Thapar, & McKoon, 2004; Ratcliff, Van Zandt, & McKoon, 1999; Strayer & Kramer, 1994; Thapar, Ratcliff, & McKoon, 2003). Secondly, it performs extraordinarily well in terms of parsimony and description of interesting patterns in reaction time data (e.g., Ratcliff, 1987; Ratcliff & Rouder, 1998). Thirdly and finally, its main parameters have interesting process interpretations that allow for substantive insights (Voss, Rothermund, & Voss, 2004). If the RDM has one significant drawback, it is that it is prohibitively difficult to to apply in practice, to the point where methods for fitting the RDM to experimental data are a research topic in their own right (Ratcliff & Tuerlinckx, 2002; Tuerlinckx, Maris, Ratcliff, & De Boeck, 2001; Tuerlinckx, 2004; Vandekerckhove & Tuerlinckx, 2007b; Voss & Voss, 2008; Wagenmakers, van der Maas, & Grasman, 2007). Only recently, attempts are being made at making the RDM more applicable in research practice (Vandekerckhove & Tuerlinckx, 2007b, 2007a; Voss & Voss, 2007). This paper presents a MATLAB toolbox that is exactly such an attempt.

In the next four sections, we will (1) briefly describe the RDM, (2) repeat the basics of matrix methods in statistical modeling, (3) provide some practical information regarding a new tool, the *Diffusion Model Analysis Toolbox* (DMAT), and (4) give two didactic examples with code. For conceptual details regarding the RDM, the fitting procedures, and associated statistical treatment, the reader is referred to Vandekerckhove and Tuerlinckx (2007b).

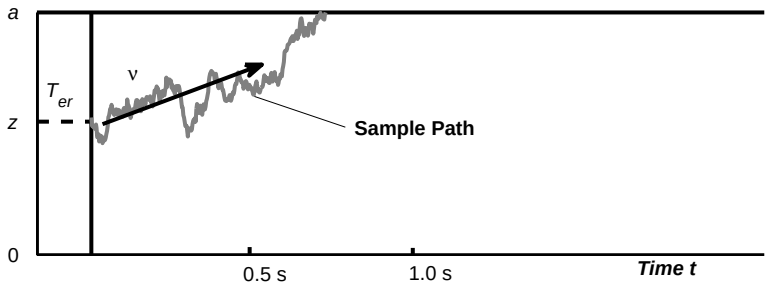


Figure 2.1: *A graphical representation of the diffusion process. The curved line indicates the amount of evidence for the 'upper' response as it evolves over time. At about 700ms, the upper boundary is crossed and the process ends. See text and Table 2.1 for more detail.*

2.2 The Ratcliff diffusion model

The basic principle behind the RDM is that of integration of noisy evidence over time. It is assumed that, in order to make a speeded choice between two options, evidence is accumulated sequentially over time. As soon as sufficient evidence towards either option has gathered, the process stops and outputs a decision (*absorbing boundaries*). The accumulation process is governed by two distinct forces, namely a tendency to drift towards either boundary (*drift rate*), and a stochastic component in the step size and direction on the decision dimension. The process itself is not assumed to be necessarily unbiased: the starting point of the process may be closer to one boundary than the other, increasing the a-priori probability for one response. Figure 2.1 shows a graphical representation of the diffusion process.

In all, the RDM as implemented in DMAT has nine free parameters for each condition. Table 2.1 lists them, their notation, and their common

Table 2.1: *The nine free parameters of the RDM, as implemented by DMAT.*

	Symbol	Parameter	Interpretation
Decision process	a	Boundary separation	Speed-accuracy trade-off (high a means high accuracy)
	z	Starting point	Bias for either response ($z = a/2$ is neutral)
	v	Drift rate	Amount of input information; Quality of the stimulus
Nondecision	T_{er}	Nondecision time	Sum of all other processes involved (motor response time, encoding...)
Intertrial variability	s_z	Intertrial range of z	Participant's variability in bias
	s_t	Intertrial range of T_{er}	Participant's variability in nondecision time
	η	Intertrial SD of v	Spurious differences in stimulus quality, or variability in attention or motivation
Mixture model	π	Proportion non-outliers	Proportion of data resulting from a diffusion process
	γ	Proportion guesses	Proportion of outliers that is a guess (and not a delayed startup)

interpretation.

2.3 Matrix notation and design of experiments

In order to impose restrictions on parameters across conditions, DMAT makes use of a matrix modeling method that is similar to the standard technique of general linear modeling (see Vandekerckhove & Tuerlinckx, 2007b, for a more detailed explanation and examples of this method). In particular, if there are c conditions, a vector $\underset{c\times 1}{\mathbf{\Psi}}$ of a given type of parameters across conditions is assumed to be the result of the matrix product $\underset{c\times m}{\mathbf{D}} \times \underset{m\times 1}{\Theta}$, given that $\mathbf{D}$ is a design matrix and that $\mathbf{\Theta}$ is a vector with free parameters that remain. Crucially, $\mathbf{\Theta}$ contains no more elements than does **Psi** ($m \leq c$), often resulting in a more parsimonious model with fewer parameters to estimate.

Depending on how the design matrix is formulated, the restrictions change. If, for example, $\mathbf{D}$ is a column of ones, then for any value of $\mathbf{\Theta}$,

the product $\boldsymbol{\Psi}$ will be a vector with all elements equal. On the other hand, if $\mathbf{D}$ is the identity matrix, then $\boldsymbol{\Psi}$ will be identical to $\boldsymbol{\Theta}$, and no restrictions will have been applied. If $\mathbf{D}$ contains a column of ones and a second column of covariate values, restrictions in the form of a linear regression are applied. The $\hat{\boldsymbol{\Theta}}$ vector will then contain an estimate for the intercept as its first element and the regression weight of the linear model as its second element.

Of course it is possible to construct even more complex models, such that a parameter may be made dependent on more than one covariate, linear or categorical; on their interactions; on participants; and so on. Also, different designs may be implemented for different parameters.

DMAT requires that a design matrix is formulated for each parameter of the model. In most cases, however, the design matrix will be the column of ones, indicating no variability across conditions. (This is also the default setting for DMAT models.)

2.4 The Diffusion Model Analysis Toolbox

2.4.1 Requirements

The DMA Toolbox requires that you have MATLAB 7.2 (R2006a) or a more recent version installed. The Optimization Toolbox also needs to be installed. If they are available, DMAT will make use of the Statistics and Symbolic Math toolboxes, but these are not required.

The toolbox was developed and tested on Windows and Linux platforms.

2.4.2 Installation

The software tool we are presenting is a MATLAB toolbox that can be freely downloaded via `http://ppw.kuleuven.be/okp/dmatoolbox`. Upon filling out a form, you will be e-mailed a link where a ZIP archive is available. The archive contains some 300 files, 70 of which are MATLAB functions. Unpackage the file to the
toolbox folder of your MATLAB install, and then execute the included installer function from the MATLAB command window. The installer will guide you from there. If you did not unpackage the ZIP archive in the toolbox directory (e.g., because you do not have write access to it), the installer will ask you to locate it first. When the installation is finished, you can test the toolbox by calling the function test_main. Since DMAT is constantly under development, its most recent version, bug reports and fixes can be found on the website.

2.4.3 End User License Agreement

While the installer runs, you will be asked to read and accept an End User License Agreement. Please note that, while DMAT and its source code may be downloaded at no cost, it is *not* permitted to redistribute the code or derived code without the authors' consent. We welcome cooperation from third parties in further developing DMAT, but in order to maintain transparency regarding exactly which methods an end user has implemented, we want the distribution of this toolbox to remain centralized.

2.5 Usage and examples

2.5.1 Two interfaces

DMAT features both a graphical user interface (GUI) and a command interface (CI). To start the GUI, simply type "dmatgui" in the MATLAB command window. In the examples, we will focus mainly on the CI, which requires some coding. Using the GUI should be largely self-evident if the commands for the CI are known. Where it isn't, we will explicitly mention how the GUI expects and handles user input.

2.5.2 Data sets

For either interface, the data set should be provided in a proper format, meaning that it should be a three-column matrix where each row indicates a trial in the experiment. Of the three columns, the first contains the condition, the second contains the response type (0 or 1, usually meaning incorrect and correct) and the third contains the reaction time in seconds. If the data matrix contains only two columns, all trials will be assumed to be in the same condition. If you use the CI, the data should be contained in a *double array*. For the GUI, it can be a double array in a MATLAB file (.mat), or ASCII data in a tab-delimited file (.tab or .dat), a comma-delimited file (.csv), or a space-delimited file (.txt).

2.5.3 General usage of the toolbox: Command interface

2.5.3.1 Input

The most important function in the DMA Toolbox is called "multiestv4." This function accepts as its first input a data set (as a three-column double array) and as second input an optional *options structure*. A large part of using DMAT is constructing this options structure: a MATLAB variable

with many fields containing different possible settings. Table 2.2 gives an overview of the settings (field names), their default values, and their effects. The default options structure can be obtained by calling the function multiestv4 without input arguments. Then, you can edit the fields of this structure to fit your needs. The standard syntax for this is:

```
opts = multiestv4();
opts.fieldname = value;
```

And to estimate the parameters, then:

```
output = multiestv4(data,options);
```

This will return an *output structure*, which contains information about the model fit and the optimization algorithm.

2.5.3.2 Output

Like the options structure, the output structure is a MATLAB variable which has many different fields, each one containing information about the model fit or the algorithm. Table 2.3 contains information about the available fields and what they mean. Output fields can be browsed from the command window. The following syntax will display the contents of the field *fieldname*:

`output(model).fieldname`,

where *model* refers to the numerical index of the model you are investigating. for example, output(2).Fitvalue will return the value of the deviance function of the second model, and output(2).Df will return the number of free parameters in the model.

Table 2.2: *The fields of DMAT's options structure, with default values and effects.*

Field name	Default value (multiestv4)	Effect / Use
DesignMatrix	<Columns of ones>	Parameters vary across conditions?
Display	`'off'`	How much output should DMAT give to the command window?
EWMA	<structure>	Provide parameters for the EWMA procedure. Defaults are $\lambda = .01$, $L = 1.5$, $s = .5$.
EstimationMethodScalar	5 (Multinomial likelihood with fixed bin edges)	Objective function to minimize (Multinomial likelihood or χ^2? Fixed bin edges or percentiles?)
FixedBinEdges	$\begin{bmatrix} .30 & .36 & .42 & .52 & .80 \\ .38 & .47 & .56 & .70 & 1.0 \end{bmatrix}$	If fixed bin edges, which values to use (in seconds, first row for corrects, second for errors).
FixedValues	`[]`	Provide a condition-by-parameter matrix with NaN for free parameters and a specific value for fixed ones.
Guess	`[]`	Starting position for the optimization (condition-by-parameter matrix). If none given, DMAT finds one.
GuessMethodScalar	1	If DMAT has to generate a guess, 1 causes it to use ezdiff* and 2 uses a slight perturbation.
LongSimplexRuns	1	The number of times the long simplex run should be repeated.
MaxIter	5000	The maximum number of iterations for long simplex runs.
Name	`'No name given'`	A name for the model.
NoFitting	0	If set to 1, only construct objective function, no actual fitting.
NonparametricBootstrap	0	Nonparametric bootstrap iterations.
ObjectiveDecimals	7	Number of significant decimals for the objective.
OutlierMax	`[]`	Maximum RT for inclusion.
OutlierMin	`[]`	Minimum RT for inclusion.
OutlierTreatment	`'None'`	Which outlier treatment to use.
ParameterDecimals	7	Significant decimals for parameters.
ParametricBootstrap	0	Parametric bootstrap iterations.
Percentiles	$\begin{bmatrix} 10 & 30 & 50 & 70 & 90 \\ 10 & 30 & 50 & 70 & 90 \end{bmatrix}$	If estimation with percentiles, which ones to use (values between 0 and 100; first row for corrects, second for errors).
ShortSimplexRuns	3	The number of times the short simplex run (200 iterations) should be repeated.
SpecificBias	`[]`	Per condition, value of B, where $B = z/a$. If NaN, z and a are estimated separately.

* The EZDIFF algorithm is described in Wagenmakers et al. (2007) and in Vandekerckhove and Tuerlinckx (2007b).

Table 2.3: *The fields of DMAT's output structure, with brief explanation of their contents.*

Field name	Contents
DesignVector	Point estimates of the free parameters
Df	Number of free parameters
FitInfo	Fit indices of the model
Fitvalue	Deviance of the model
Hessian	Estimate of the Hessian matrix at the minimum
Minimum	Estimate of the entire parameter set
Name	Name of the model
NonparametricBootstrapMean	If requested, nonparametric bootstrap estimate of entire parameter set
NonparametricBootstrapStdErr	If requested, nonparametric bootstrap estimate of parameters' standard errors
NonparametricBootstraps	An output structure for each nonparametric bootstrap iteration
Options	The options structure that the user provided
OutlierReport	If requested, information regarding outlier treatment
ParametricBootstrapMean	If requested, parametric bootstrap estimate of entire parameter set
ParametricBootstrapStdErr	If requested, parametric bootstrap estimate of parameters' standard errors
ParametricBootstraps	An output structure for each parametric bootstrap iteration
Simplex	Information regarding the simplex runs (convergence time, number of steps...)
StdErr	Estimate of the parameters' standard errors (based on the Hessian matrix)
Time	Total time needed for fitting this model
Warnings	Anything the user might need to know (e.g., if the Hessian indicates poor model fit)

2.5.3.3 Further processing

If two models are nested, then the difference in their deviances follows a chi-square distribution with a number of degrees of freedom equal to the difference in number of parameters, under the null hypothesis that the models are equal. Thus, the following syntax (using DMAT's chi2test function) will give the p-value of the difference between two models:

```
>> x2 = output(1).Fitvalue-output(2).Fitvalue;
>> df = output(2).Df-output(1).Df;
>> p = chi2test(x2,df);
```

A convenient function in this regard is qtable, which shows different fit values of models, and also shows the p-value of the difference in fit each pair of consecutive models. (Note that this implies that the p-values reported in a certain row are only correct if that model is nested in the preceding model.)

2.5.4 General usage of the toolbox: Grapical user interface

2.5.4.1 Input

In the GUI, you first need to load the data by clicking the "Browse" button and finding the data set (that you saved somewhere). When you have set all the options to your liking, the model can be added to the model queue by clicking the "Current Model" button. The model queue stores series of models that can be fitted in a batch submission. This is often advantageous, since each set of parameter estimates can be used as an initial guess of the next model, resulting in an increase in efficiency. In particular, if subsequent models are nested, then the parameter estimates of the more restrictive model will often be a good starting point for the less restrictive model. DMAT will perform the necessary linear transformations automatically.

Then you can start to define the next model and click "Current Model" again when you are finished. Click "Run" to start parameter estimation, or save the model queue with the "Save" button. Note that in the "Save as..." window you can choose to save the queue in a DMAT native format (*.dmq), or as an ASCII M-file that can be viewed, edited, and run from the command line. Both *.dmq files and generated M-files can be re-loaded into the GUI at a later time time by clicking the "Load" button (but you have to load a data set first, and the models in the loaded queue have to be appropriate for that data set, i.e., have the same number of

conditions). Note that DMAT, if able, automatically makes emergency back-ups (both of the most recent queue and of intermediate estimation results), and if something should go wrong, calling the dmatrescue function from the command interface might bring relief.

2.5.4.2 Output

As soon as the algorithm has started, an output window will replace the DMAT main window, allowing you to browse some descriptive statistics. The window will be updated as results from the queue become available. You can simply select the model and output type from two lists on the left hand side.

2.5.5 Simulating data

In the examples that follow, we will use simulated data. DMAT contains several functions that allow you to simulate data sets that are ready for use. Appendix 2.A shows a simple sequence of commands that will produce a data set with three conditions, which differ only in drift rate. In the GUI, simply click the "Simulator" button. After you enter the number of conditions, click "Set" and then input the parameter set and the number of data points points desired. Then click "Simulate and Save," select a file name and location, and click "Close." You cannot input a seed for the random number generator within the GUI.

2.5.6 Example 1: A simple design

2.5.6.1 Data set

For our first example, we will imagine an experiment with three conditions. The difference between conditions is supposed to be in the quality of a presented stimulus; hence we are interested in the difference in drift

rates. For our data set, we will use the one that results from the code in Appendix 2.A.

2.5.6.2 Input

To analyze these data, we will attempt to fit two models. In one model, we will make the assumption that all drift rates are equal (the "reduced model" or "null model"). In the second model, we will relax that assumption. By comparing the goodness-of-fit of these models, we can investigate the effect of condition on drift rate (much as we would in ANOVA). As explained in the section on matrix representations, constraining equality across conditions implies a design matrix which is a column of ones. Applying no constraints on a given parameter can be achieved by using the appropriate identity matrix as design matrix.

Appendix 2.B shows code for fitting these two models. Note that the only setting we have adapted is the DesignMatrix field. Usually, the rest of the default settings provided by multiestv4 are well suited.

In the GUI, you can simply click "Current Model" as soon as the data set is loaded, since the default model doesn't need to be changed for Model 1. For Model 2, the design matrix for drift rate (v) needs to be changed to an identity matrix. Click the drop-down menu next to "View/Edit design matrix" and select "v." In the window that pops up, you can choose to manually input an identity matrix, or simply select "Identity" in the drop-down menu. Confirm with OK and click "Current Model" to add the second model to the queue.[1] Finally, click "Run" to start parameter estimation.

[1] Actually, we have built some short-cuts into the DMAT GUI that make this example even easier. Load the data, click "Predefined," select "No effects" and "Effect on v only," and click "Add" and then "Run."

2.5.6.3 Output

We can inspect the best fitting parameter set of the second model in the usual way:

```
>> output(2).Minimum
ans =  0.152  0.300  0.061  0.112  0.015  0.102  0.403
       0.152  0.300  0.061  0.112  0.015  0.102  0.175
       0.152  0.300  0.061  0.112  0.015  0.102  0.003
```

2.5.6.4 Statistical processing

We can find the p-value of the difference in deviance between the two models (we know the number of parameters to be estimated increased by 2 from Model 1 to Model 2):

```
>> x2 = output(1).Fitvalue-output(2).Fitvalue
x2 = 945.317
>> p = chi2test(x2,2)
p = 0.000
```

This result indicates that there is a highly significant effect of condition on the drift rate—which is as we expected. To get a quick summary of the model fits, use the qtable function (see Appendix 2.B for the output that would yield).

2.5.7 Example 2: A more complicated design

2.5.7.1 Data set

For our second example, we simulate data with eight conditions. Conditions 1 through 4 contain an accuracy instruction and 5 through 8 contain a speed instruction (influencing boundary separation). A second manipulation, within these groups, pertains to the quality of the stimulus (influencing drift rate). This time, however, the manipulation is a continuous

covariate $\mathbf{X}$, which takes the values 0.2, 0.8, −0.4, and 0.6. In Appendix 2.C we present code that will generate such data.

2.5.7.2 Input

We will construct a queue of three models, differing only in the design matrices. In the first model, we will apply no design (all parameters constrained to be equal across conditions). In the second model, the experimental design is implemented (allowing the first four conditions to have a different boundary separation from the last four, and constraining drift rate to be a linear function of the covariate $\mathbf{X}$). In the third model, we allow both boundary separation and drift rate to vary freely across conditions (to test for deviation from the design). Furthermore, we will tweak some settings of the fitting algorithm. Firstly, we no longer want to use fixed reaction time bins, but rather use Quantile Probability Products (Brown & Heathcote, 2003; Heathcote & Brown, 2004; Heathcote, Brown, & Mewhort, 2002; see also Vandekerckhove & Tuerlinckx, 2007b). To this end, we we give the EstimationMethodScalar option the value 6 (see DMAT's included documentation for a table of valid values for this setting). We will choose classical percentiles 10, 30, 50, 70, and 90, and add the first and fifth percentile to achieve a better fit of the left slope of the RT distribution. Finally, we choose not to estimate the starting point z, but rather fix it to $a/2$, since nothing in the experiment has given us cause to assume an a-priori bias on behalf of the fictional participant. Appendix 2.C shows all the code needed to provide DMAT with this input. In the GUI, the estimation method and percentiles can be set in the Advanced window, and the bias can be set via the "Bias: Set" button in the main window. The matrices have to be input manually (or pasted from an external editor).

Two things may be mentioned regarding the code in Appendix 2.C.

Firstly, note the use of two short-cuts that are available in coding the design matrices. Supplying `'1'` (of data type *char array*) for a design matrix will be interpreted as a column of ones and supplying `[]` (the empty array) will be interpreted as an identity matrix. (Which are short-cuts in the sense that you don't need to figure out exactly how large these matrices ought to be.) Secondly, note that in order for DMAT to recognize the "specified bias" (the restriction that $z = a/2$), we need to supply a non-restrictive design matrix for z. Otherwise, the design matrix restrictions will override the specified-bias restriction (DMAT will print a warning when this happens). The non-restrictive design of course means the identity matrix (or its short-cut, the empty matrix).

2.5.7.3 Output

We can see the difference between the three models by calling the qtable function. In the resulting table, shown as Table 2.4, we we can see that Model 2 outperforms Model 1 [$\chi^2(2) = 4,087.38$, $p < 10^{-10}$] indicating a significant effect of the experimental design), but Model 3 does not outperform Model 2 [$\chi^2(2) = 16.99$, $p < .1501$]. Model 2 also has the lowest $AICc$ and BIC values. Finally, Figure 2.2 is a screenshot of the GUI, showing drift rate as a function of condition. condition. The error bars are in this case based on the Fisher information matrix (which is derived from the Hessian matrix). If bootstrap analyses had requested, bootstrap confidence intervals would be shown.

2.6 Summary

In this paper, we have presented DMAT, an application with a low ease of use threshold that enables fitting and evaluation of the Ratcliff diffusion model. We briefly described the model and the basic matrix methods used

Table 2.4: *Output of the qtable function in Example 2.*

Deviance	*d-Deviance*	*df*	*d-df*	*AICc*	*BIC*	*p*
20206.10	NaN	6	NaN	20218.12	20255.24	NaN
16118.72	4087.38	8	2	16134.75	16184.23	0.00
16101.73	16.99	20	12	16141.95	16265.51	0.15

Note—*Deviance* is the badness-of-fit measure, *d-Deviance* is the difference between consecutive models, *df* is the number of parameters in each model, *d-df* is the difference with the previous model, *AICc* is the small-sample Akaike Information Criterion (Hurvich & Tsai, 1989), *BIC* is the Bayesian Information Criterion, and *p* is the significance of the difference between consecutive models (based on a χ^2 test of *d-Deviance* with *d-df* degrees of freedom).

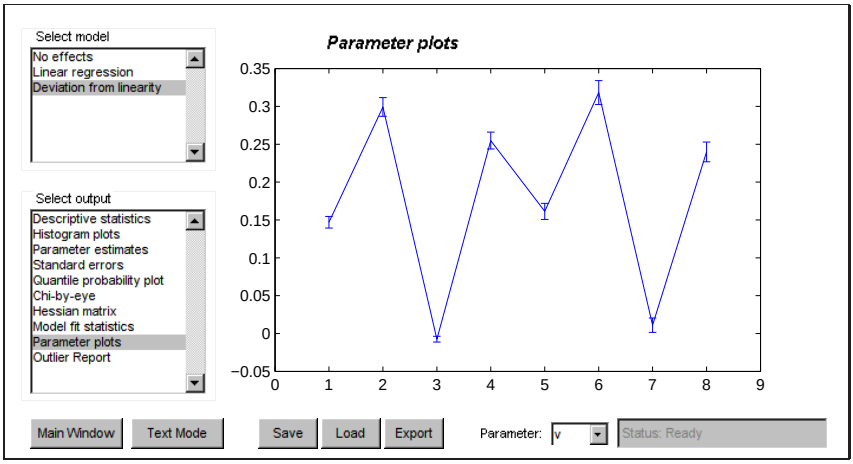


Figure 2.2: *A screenshot from the DMA Toolbox. Parameter estimation is complete and the user can browse through the output. The graph shows estimated drift rates (and error bars of one standard error, obtained from the Hessian matrix) as a function of condition, as obtained under the model that allows deviation from the linear design.*

in general linear modeling, which we have extended for use in diffusion models. With two didactic examples, we demonstrated the use of DMAT.

References

Brown, S., & Heathcote, A. (2003). QMLE: Fast, robust and efficient estimation of distribution functions based on quantiles. *Behavior*

Research Methods, Instruments & Computers, *35*, 485–492.

Heathcote, A., & Brown, S. (2004). Reply to speckman and rouder: A theoretical basis for QML. *Psychonomic Bulletin & Review*, *11*, 577.

Heathcote, A., Brown, S., & Mewhort, D. (2002). Quantile maximum likelihood estimation of response time distributions. *Psychonomic Bulletin & Review*, *9*, 394–401.

Hurvich, C. M., & Tsai, C.-L. (1989). Regression and time series model selection in small samples. *Biometrika*, *76*, 297–307.

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, *85*, 59–108.

Ratcliff, R. (1981). A theory of order relations in perceptual matching. *Psychological Review*, *88*, 552–572.

Ratcliff, R. (1987). More on the speed and accuracy of positive and negative responses. *Psychological Review*, *94*, 277–280.

Ratcliff, R. (1988). Continuous versus discrete information processing: Modeling the accumulation of partial information. *Psychological Review*, *95*, 238–255.

Ratcliff, R. (2002). A diffusion model account of response time and accuracy in a brightness discrimination task: Fitting real data and failing to fit fake but plausible data. *Psychonomic Bulletin & Review*, *9*, 278–291.

Ratcliff, R., Gomez, P., & McKoon, G. (2004). Diffusion model account of lexical decision. *Psychological Review*, *111*, 159–182.

Ratcliff, R., & Rouder, J. N. (1998). Modeling response times for two-choice decisions. *Psychological Science*, *9*, 347–356.

Ratcliff, R., & Rouder, J. N. (2000). A diffusion model account of masking in two-choice letter identification. *Journal of Experimental Psychology: Human Perception and Performance*, *26*, 127–140.

Ratcliff, R., Thapar, A., & McKoon, G. (2004). A diffusion model analysis

of the effects of aging on recognition memory. *Journal of Memory and Language*, *50*, 408–424.

Ratcliff, R., & Tuerlinckx, F. (2002). Estimating parameters of the diffusion model: Approaches to dealing with contaminant reaction times and parameter variability. *Psychonomic Bulletin & Review*, *9*, 438–481.

Ratcliff, R., Van Zandt, T., & McKoon, G. (1999). Connectionist and diffusion models of reaction time. *Psychological Review*, *102*, 261–300.

Strayer, D., & Kramer, A. (1994). Strategies and automaticity: I. *Basic findings and conceptual framework. Journal of Experimental Psychology: Learning, Memory, & Cognition*, *20*, 318–341.

Thapar, A., Ratcliff, R., & McKoon, G. (2003). A diffusion model analysis of the effects of aging on letter discrimination. *Psychology & Aging*, *18*, 415–429.

Tuerlinckx, F. (2004). The efficient computation of the distribution function of the diffusion process. *Behavior Research Methods, Instruments, & Computers*, *36*, 702-716.

Tuerlinckx, F., Maris, E., Ratcliff, R., & De Boeck, P. (2001). A comparison of four methods for simulating the diffusion process. *Behavior Research Methods, Instruments, & Computers*, *33*, 443–456.

Vandekerckhove, J., & Tuerlinckx, F. (2007a). The Diffusion Model Analysis Toolbox. [Software and manual] [Computer software manual]. (Retrieved from `http://ppw.kuleuven.be/okp/dmatoolbox`)

Vandekerckhove, J., & Tuerlinckx, F. (2007b). Fitting the Ratcliff diffusion model to experimental data. *Psychonomic Bulletin & Review*, *14*, 1011–1026.

Voss, A., Rothermund, K., & Voss, J. (2004). Interpreting the parameters of the diffusion model: An empirical validation. *Memory & Cogni-*

tion, *32*, 1206–1220.

Voss, A., & Voss, J. (2007). Fast-dm: A free program for efficient diffusion model analysis. *Behavior Research Methods*, *39*, 767–782.

Voss, A., & Voss, J. (2008). A fast numerical algorithm for the estimation of diffusion-model parameters. *Journal of Mathematical Psychology*, *52*, 1–9.

Wagenmakers, E.-J., van der Maas, H. J. L., & Grasman, R. P. P. P. (2007). An EZ-diffusion model for response time and accuracy. *Psychonomic Bulletin & Review*, *14*, 3–22.

2.A Annotated code for generating random data in DMAT

Define a parameter set from which to sample

Parameter sets are matrices with one row for each condition, and one column for each parameter. The parameters are `a`, `Ter`, `eta`, `z`, `sz`, `st`, `v`:

```
parameter_set = [.16 .30 .08 .12 .02 .10 .40
                 .16 .30 .08 .12 .02 .10 .20
                 .16 .30 .08 .12 .02 .10 .00];
```

Define simulation constants

There are two more constants that the simulation will need. First is the number of data points in each condition, `N`. Note that you can choose between providing `N` as a scalar, like here, or as a vector with one number for each condition, like: `N = [500 500 1000]`. Secondly, since the simulator makes use of random number generators, a 'seed' is needed. If you provide none, a seed is selected with the help of the system clock. For reproducibility, we choose a seed here:

```
N = 1000;
seed = 0;
```

Simulate data

With the **multisimul** function:

```
data = multisimul(parameter_set,N,seed);
disp(data(1:10,:))
```

```
    3.0000    1.0000    1.2130
    2.0000    1.0000    0.4999
    2.0000    1.0000    0.3589
    2.0000    1.0000    0.3912
    1.0000    1.0000    0.5518
    1.0000    1.0000    0.4063
    3.0000    1.0000    0.4475
    2.0000    1.0000    0.3741
    1.0000    1.0000    0.4261
    2.0000    1.0000    0.4417
```

2.B Annotated code for fitting two nested diffusion models in DMAT

Prepare the options structure

Get default options structure:

```
options = multiestv4
```

```
options =

             DesignMatrix: 1x7 cell
                  Display: 'off'
                     EWMA: [1x1 struct]
    EstimationMethodScalar: 5
              FixedBinEdges: [2x5 double]
                FixedValues: []
                   FitBnotZ: 0
```

```
                    Guess: []
          GuessMethodScalar: 1
            LongSimplexRuns: 1
                    MaxIter: 5000
                       Name: 'No name given'
                  NoFitting: 0
      NonparametricBootstrap: 0
           ObjectiveDecimals: 7
                 OutlierMax: []
                 OutlierMin: []
           OutlierTreatment: 'None'
           ParameterDecimals: 7
         ParametricBootstrap: 0
                Percentiles: [2x5 double]
           ShortSimplexRuns: 3
               SpecificBias: []
```

Copy it, because we want to fit more than one model

```
options = repmat(options,2,1);
```

Prepare the design matrix for each model

In Model 1, all parameters are kept equal across conditions. In Model 2, drift rate is allowed to vary freely:

```
O = ones(3,1);
I = eye(3);
design_matrix1 = {O,O,O,O,O,O,O,O,O};
design_matrix2 = {O,O,O,O,O,O,I,O,O;}
```

Insert the design matrices into the options structure

```
options(1).DesignMatrix = design_matrix1;
options(1).Name = 'No effect';
options(2).DesignMatrix = design_matrix2;
```

```
options(2).Name = 'Effect on v only';
```

Invoke DMAT

Use the **multiestv4** function to get parameter estimates:

```
output = multiestv4(data,options);
```

```
    Starting model 1 of 2: No effect

      Guess     :   13482.80335984     (25-Dec-2008 20:40:24)
      Simplex  1:   11644.02574703     (25-Dec-2008 20:40:28)
      Simplex  2:   11635.47143151     (25-Dec-2008 20:40:32)
      Simplex  3:   11635.46000666     (25-Dec-2008 20:40:37)
      Simplex  4:   11635.46000546     (25-Dec-2008 20:40:44)
      Final loss:   11635.46000546     (25-Dec-2008 20:40:47)

      The recovered sZ parameters are suspect.
      Trying again.

      Guess     :   11654.27977576     (25-Dec-2008 20:40:47)
      Simplex  1:   11635.63165838     (25-Dec-2008 20:40:54)
      Simplex  2:   11635.47863373     (25-Dec-2008 20:40:59)
      Simplex  3:   11635.46420571     (25-Dec-2008 20:41:04)
      Simplex  4:   11635.46000551     (25-Dec-2008 20:41:14)
      Final loss:   11635.46000551     (25-Dec-2008 20:41:16)

      The last convergence point was still a suspect result.
      Returning to the best point found and giving up.

Warning: Hessian is not positive definite.
    Starting model 2 of 2: Effect on v only

      Guess     :   11635.46000558     (25-Dec-2008 20:41:18)
      Simplex  1:   10679.01165606     (25-Dec-2008 20:41:23)
      Simplex  2:   10676.02448910     (25-Dec-2008 20:41:27)
      Simplex  3:   10675.98667395     (25-Dec-2008 20:41:32)
      Simplex  4:   10675.98344462     (25-Dec-2008 20:41:42)
      Final loss:   10675.98344462     (25-Dec-2008 20:41:44)

      The recovered sZ parameters are suspect.
```

```
Trying again.

Guess     :   10698.82098705     (25-Dec-2008 20:41:44)
Simplex  1:   10674.04939412     (25-Dec-2008 20:41:49)
Simplex  2:   10673.30388143     (25-Dec-2008 20:41:53)
Simplex  3:   10673.18461789     (25-Dec-2008 20:41:58)
Simplex  4:   10673.14120078     (25-Dec-2008 20:42:11)

Final loss:   10673.14120078     (25-Dec-2008 20:42:14)
```

Process results

Use the **qtable** function to get a concise summary of the model queue:

```
qtable(output)
```

```
------------------------------------------------------------------------
   Deviance  d-Deviance   df   d-df        AICc         BIC         p
------------------------------------------------------------------------
 11635.4600         NaN    7    NaN  11649.4974  11691.5046       NaN
 10673.1412    962.3188    9      2  10691.2014  10745.1985   0.00000
------------------------------------------------------------------------
```

2.C Annotated code for generating the data set and estimating the models described in example 2

Simulating the data

Define a parameter set from which to sample

We will define a more complex design for this example. We will suppose eight different conditions. Conditions 1 through 4 contain an accuracy instruction and 5 through 8 contain a speed instruction. A second manipulation, within these groups, again pertains to the quality of the stimulus. This time, however, it is a continuous variable which takes the values .2 .8

-.4 .6. The following parameter set is roughly what we would expect from such an experiment:

```
parameter_set = [.24 .30 .08 .12 .02 .10 .15
                 .24 .30 .08 .12 .02 .10 .30
                 .24 .30 .08 .12 .02 .10 .00
                 .24 .30 .08 .12 .02 .10 .25
                 .08 .30 .08 .04 .02 .10 .15
                 .08 .30 .08 .04 .02 .10 .30
                 .08 .30 .08 .04 .02 .10 .00
                 .08 .30 .08 .04 .02 .10 .25];
```

Define simulation constants

Let's say that the number of data points per condition wasn't equal here:

```
N = [470 440 500 450 430 460 400 450];
seed = 0;
```

Simulate data

```
data = multisimul(parameter_set,N,seed);
disp(data(1:10,:))
```

```
    6.0000    1.0000    0.3451
    4.0000    1.0000    0.7723
    3.0000    1.0000    1.7779
    3.0000         0    0.6231
    1.0000    1.0000    0.5785
    1.0000    1.0000    1.3063
    6.0000    1.0000    0.3371
    8.0000    1.0000    0.3294
    3.0000         0    1.3768
    8.0000    1.0000    0.3747
```

Fitting the model

Prepare the options structure

Get default options structure

```
options = multiestv4;
```

Copy it, because we want to fit more than one model

```
options = repmat(options,3,1);
```

Adapt all of the options structures at once to change some settings

We want to use a percentile-based method instead of a fixed-bins method. (The **deal** function changes all fields with a given name in an array of structures simultaneously. Note the required use of `[]` at the left hand side here.)

```
[options.EstimationMethodScalar] = deal(6);
```

Setting this field to 6 indicates that we want a multinomial likelihood estimation based on percentiles (Quantile Maximum Probability Estimation). A table with possible values for EstimationMethodScalar is given in the DMAT documentation.

Since we indicated that we want to estimate on the basis of percentiles, we need to indicate which percentiles to use. Classically, percentiles 10, 30, 50, 70, and 90 are used, but adding some smaller values increases recovery of the left slope of the RT distribution.

```
[options.Percentiles] = deal([1 2 5 10 30 50 70 90
                              1 2 5 10 30 50 70 90]);
```

We also don't want to estimate the starting point, but want to fix it to exactly half of boundary separation, in all conditions:

```
[options.SpecificBias] = deal([.5 .5 .5 .5 .5 .5 .5 .5]);
```

Prepare the design matrix for each model

In Model 1, all parameters are kept equal across conditions. However, we do need to make a change in the design matrices. Since we supplied an extra restriction (namely that all z = a/2), we need to remove the design matrix restriction to avoid a conflict. Thus:

```
options(1).DesignMatrix = ...
            {'1','1','1',[],'1','1','1','1','1'};
options(1).Name = 'No effects';
```

Note also that here we make use of two short-cuts built into the code: supplying `'1'` instead of a design matrix restricts that parameter to be equal across conditions. Supplying an empty matrix (`[]`) allows it to vary without restriction.

In Model 2, drift rate is allowed to vary as a linear function of the covariate, and boundary separation is allowed to vary between the two manipulations.

```
v_covariate = [.2 .8 -.4 .6 .2 .8 -.4 .6]';
v_intercept = ones(8,1);
v_dm = [v_intercept v_covariate]
```

```
v_dm =

    1.0000    0.2000
    1.0000    0.8000
    1.0000   -0.4000
    1.0000    0.6000
    1.0000    0.2000
    1.0000    0.8000
    1.0000   -0.4000
    1.0000    0.6000
```

```
a_dm = [ones(4,1) zeros(4,1);zeros(4,1) ones(4,1)];
```

```
options(2).DesignMatrix = ...
            {a_dm,'1','1',[],'1','1',v_dm,'1','1'};
options(2).Name = 'Linear regression';
```

In Model 3, drift rate and boundary separation are allowed to vary freely across conditions (to check for deviations from the design).

```
options(3).DesignMatrix = ...
            {[],'1','1',[],'1','1',[],'1','1'};
options(3).Name = 'Deviation from linearity';
```

Invoke DMAT

Use the **multiestv4** function to get parameter estimates

```
output = multiestv4(data,options);
```

```
    Starting model 1 of 3: No effects
Warning: Automatically generated guess was outside
parameter space. Generating new guess.

      Guess     :   21129.87060030     (25-Dec-2008 20:46:39)
      Simplex  1:   20325.38699337     (25-Dec-2008 20:46:59)
      Simplex  2:   20269.33154165     (25-Dec-2008 20:47:31)
      Simplex  3:   20269.33152623     (25-Dec-2008 20:48:06)
      Simplex  4:   20269.33152623     (25-Dec-2008 20:48:40)
      Final loss:   20269.33152623     (25-Dec-2008 20:48:53)

Warning: Hessian is not of full rank.
    Starting model 2 of 3: Linear regression

      Guess     :   20269.33152623     (25-Dec-2008 20:49:01)
      Simplex  1:   17346.55024075     (25-Dec-2008 20:49:24)
      Simplex  2:   17322.22852869     (25-Dec-2008 20:49:43)
      Simplex  3:   17322.21130713     (25-Dec-2008 20:50:04)
      Simplex  4:   17322.21130712     (25-Dec-2008 20:50:33)
      Final loss:   16167.60784974     (25-Dec-2008 20:51:05)

    Starting model 3 of 3: Deviation from linearity
```

```
Guess     :   16167.60784974     (25-Dec-2008 20:51:09)
Simplex  1:   16155.85066211     (25-Dec-2008 20:51:24)
Simplex  2:   16155.32897820     (25-Dec-2008 20:51:40)
Simplex  3:   16155.18402990     (25-Dec-2008 20:51:56)
Simplex  4:   16152.05264367     (25-Dec-2008 20:57:56)
Final loss:   16151.98518856     (25-Dec-2008 21:00:52)
```

Display summary output

```
qtable(output)
```

```
-------------------------------------------------------------------------
   Deviance  d-Deviance   df   d-df        AICc         BIC          p
-------------------------------------------------------------------------
 20269.3315         NaN    6    NaN  20281.3549  20318.4637        NaN
 16167.6078   4101.7237    8      2  16183.6479  16233.1174    0.00000
 16151.9852     15.6227   20     12  16192.2199  16315.7590    0.20914
-------------------------------------------------------------------------
```

Bayesian

CHAPTER 3

A Bayesian Approach to Diffusion Process Models of Decision-Making

Abstract

The Wiener diffusion model, and its extension to the Ratcliff diffusion model, are powerful and well developed process accounts of the time course of human decision-making in two-choice tasks. Typically these models have been applied using standard frequentist statistical methods for relating model parameters to behavioral data. Although this approach has achieved notable successes, we argue that the adoption of Bayesian methods promises to broaden the scope of the psychological problems the models can address. In a Bayesian setting, it is straightforward to include linear, non-linear, and categorical covariates of the basic model parameters, and so provide a much richer characterization of individual differences, the properties of stimuli, the effects of task instructions, and a range of other

important issues. In this paper, we provide an example of the Bayesian possibilities by applying the Ratcliff diffusion model to a benchmark data set involving a brightness discrimination task. We simultaneously use a categorical covariate and nonlinear regression to model the psychophysical function in a theoretically satisfying way. We also use Bayesian inference on latent class assignment variables to identify and accommodate contaminant data at the level of individual trials, categorizing them as 'diffusion' trials, 'guesses,' and 'delayed startup' trials. Using our application as a concrete example, we discuss the potential benefits of applying the Bayesian framework to process models in the cognitive sciences.

3.1 Introduction

One area of the cognitive sciences that has many formal models is that of choice reaction time (RT), particularly when the number of choices is restricted to two (Bogacz, Brown, Moehlis, Holmes, & Cohen, 2006). The practical application of many of the available models, however, has historically been hampered by computational difficulties (see e.g., Vandekerckhove & Tuerlinckx, 2007). This is particularly the case for one prominent class of models based on diffusion processes, including the Wiener diffusion model and its popular extension, the Ratcliff diffusion model (Ratcliff, 1978; Wagenmakers, in press).

For the latter model, several pieces of software have been published to aid in fitting these models to data (Vandekerckhove & Tuerlinckx, 2008; Voss & Voss, 2007). Nevertheless, the application of statistical models to the diffusion parameters with these programs is, at present, restricted to the application of linear constraints (such as ANOVA or polynomial regression). The spectrum of possible applications of the diffusion model is much broader than that. In this paper we employ psychophysical curve

fitting and latent class assignments in a Bayesian[1] treatment of the diffusion model, as an example of how Bayesian methods can broaden the class of psychological problems diffusion models can address.

The structure of the paper is as follows. We first describe the Wiener and Ratcliff diffusion models as process accounts for two-choice RT. We then report an example diffusion model analysis using Bayesian methods—based on previously studied data relating to a brightness discrimination task—that would be highly challenging to implement in a classical frequentist context. We also demonstrate using this example that the Bayesian approach can be successfully applied to relatively small sample sizes. Finally, we discuss the power and generality of the framework for extending the potential of process models in the cognitive sciences.

3.2 Diffusion models

3.2.1 The Wiener diffusion model

The Wiener diffusion model as a process for speeded decisions starts from the basic principle of *accumulation of information*. When an individual is asked to make a binary choice on the basis of an available stimulus, the assumption is that evidence from the stimulus accumulates over time and a decision is made when an upper or lower boundary is reached. Which boundary is reached determines which response is given, and the number of accretion steps taken is related to the RT. The model can also be seen as a continuous-time version of the Sequential Probability Ratio Test (SPRT; Laming, 1968), and is used frequently in neuroscience (Smith & Ratcliff, 2004).

[1]We want to emphasize that we are using Bayesian methods as a framework for statistical inference, and *not* as a set of theoretical assumptions about how humans make inferences. This means we are not proposing a 'rational' or 'computational-level' model of cognition, despite our reliance on Bayesian methods of inference.

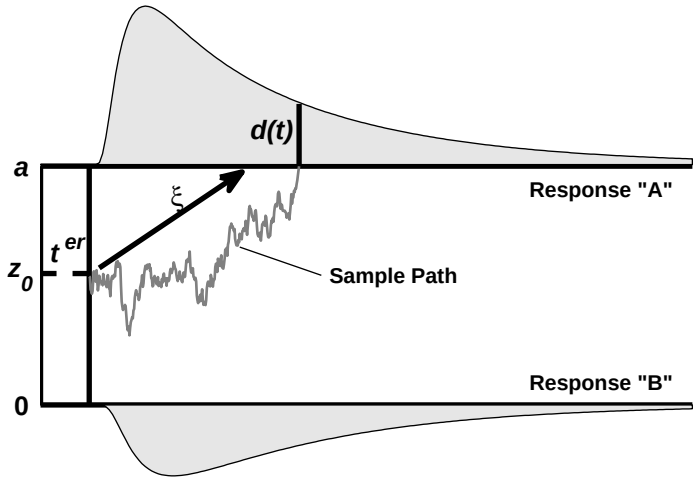


Figure 3.1: *A graphical illustration of the diffusion model. Note that* $z_0 = a \times b$. *In the Ratcliff diffusion model,* b, t^{er}, *and* ξ *vary from trial to trial. The probability density for a correct response given at time* t *is shown as* $d(t)$.

Figure 3.1 depicts the diffusion process, and shows the main parameters of the process. On the vertical axis there are the *boundary separation* a, indicating the level of evidence required to make a response (i.e., speed-accuracy trade-off) and the *starting point* z_0, indicating the a priori status of the evidence counter. The arrow represents the average rate of information uptake or *drift rate* ξ, which indicates the average amount of evidence that the observer receives from the stimulus at each sampling. Finally, the short dashed line indicates the *nondecision time* t^{er}, the time used for everything except making a decision (i.e., encoding the stimulus and physically executing the response).

It is important to note that, considering the Bayesian statistical context of this article, it will be more convenient to use a different parametrization of the process. We will therefore not consider the starting point z_0, but rather use the *initial bias* b, defined as $b = z_0/a$.

3.2.2 The Ratcliff diffusion model

Despite the elegance of the basic Wiener process as an account of the time course of decision-making, the evolution of diffusion models has involved a series of additional assumptions. These have all been intended to address shortcomings in the ability of the basic model to capture empirical regularities observed in data from human decision-making experiments.

One important change has been the introduction of additional noise processes to capture cross-over effects. 'Cross-over effects' refer to the observation that errors can sometimes be, on average, faster than correct decisions, but other times are as slow or slower. These possibilities are not accommodated by the basic model in Figure 3.1 without allowing for variation in the parameters. Accordingly, to predict fast errors, the basic model is extended by assuming that the starting point is subject to between-trial variation according to a Gaussian or uniform distribution. Similarly, to predict slow errors, it is assumed that the mean drift rate is also subject to between-trial variation according to a Gaussian distribution.

Additionally, for empirical reasons the nondecision time is assumed to vary from trial to trial, usually according to a uniform distribution. These three noise processes are parameterized with the standard sufficient statistics (mean and variance of a Gaussian or mean and range of a uniform), which become additional parameters of the model. When the Wiener diffusion model is extended with trial-to-trial variabilities such as these, it is often called the Ratcliff diffusion model (Wagenmakers, in press). This extended model comes with a much greater computational burden (see Tuerlinckx, 2004).

Notation In this paper, we will use X and T to refer to the accuracy and RT variables, and x and t for specific instances of these variables.

We will sometimes write T^* and t^* to refer to response vectors (X, T) and (x, t), respectively. We will use indices i $(i = 1, \ldots, I)$ and j $(j = 1, \ldots, J)$ to indicate conditions and k $(k = 1, \ldots, K)$ for trials within conditions. To indicate a vector, we will use a bold font, so that $\mathbf{a}$ is the matrix of boundary separations, in all conditions and all trials. We use the symbol $\sim$ to denote "is distributed according to," so that $t^*_{ijk} \sim \text{Wiener}_{X,T}(a_{ijk}, t^{er}_{ijk}, b_{ijk}, \xi_{ijk})$ and the proportionality symbol $\propto$ to denote "is proportional to".

3.3 Application to benchmark data

To illustrate the advantages and the potential of approaching diffusion models from a Bayesian perspective, we revisit a benchmark data set (Ratcliff & Rouder, 1998). We estimate five parameters of the Ratcliff diffusion model (for simplicity, we assume an unbiased diffusion process, so that $b = 0.5$) and perform a non-linear regression and a latent class assignment.

3.3.1 Data set

In the experiment by Ratcliff and Rouder (1998), there were two manipulations of interest. First, there was a speed-accuracy instruction (participants were either instructed to be fast or to be accurate) and second, there was a manipulation of brightness. The task was a 2AFC procedure, whereby each participant was shown a stimulus and had to judge whether this stimulus was drawn from a 'bright' distribution or from a 'dark' distribution (the two distributions overlapped significantly, so subjects could not be highly accurate; in total, there were 33 different levels of brightness, '1' being completely dark and '17' being completely ambiguous). Feedback was given after each trial. There were three participants (labeled KR, JF, and NH), and the experiment ran over the course of 11 days. After prepro-

cessing[2], there were a varying number of trials in each cell of the design, but the total was around 8,000 for each participant.

From the manipulations, we can expect two things. First, we expect that the speed-accuracy instruction will have an effect on boundary separation. Secondly, we expect that the brightness of the stimulus influences the drift rate. Furthermore, it is likely that the data set will contain at least some contaminant data, which we define as data points that are not generated by the process of interest and are hence not completely germane to the research question. In line with previous work, we will consider two types of contaminants: *guesses* and *delayed startups* (Vandekerckhove & Tuerlinckx, 2007).

3.3.2 Bayesian modeling

We implemented a Bayesian analysis of the brightness discrimination task data using the graphical model presented in Figure 3.2. Graphical models (see Griffiths, Kemp, & Tenenbaum, in press, and Lee, 2008, for psychological introductions) are a convenient language for describing the probabilistic relationship between parameters and data. In a graphical model, variables of interest are represented by nodes in a graph, with children depending on their parents. Circular nodes represent continuous variables, square nodes discrete variables, shaded nodes observed variables, and unshaded nodes unobserved variables. In addition, plates enclose parts of a graph to denote independent replication.

An important practical advantage of adopting the graphical model formalism is that it allows our modeling to be implemented using WinBUGS (Lunn, Thomas, Best, & Spiegelhalter, 2000). This makes it straightfor-

[2]We applied similar preprocessing as Ratcliff and Rouder, removing all trials from the first day, the first 20 trials of the other days, and the first trial of each block. In contrast to their analysis, we did not remove any trials based on an extreme RT.

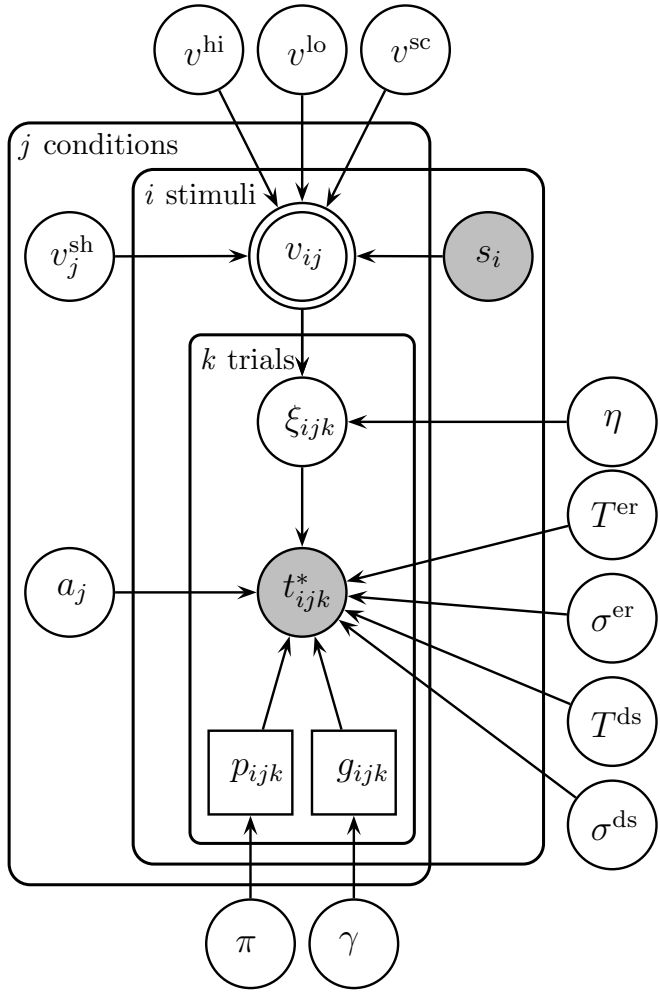


Figure 3.2: *Graphical model representation of our Bayesian analysis of the Ratcliff diffusion model against the benchmark brightness discrimination data.*

ward to perform full Bayesian inference computationally, using standard MCMC methods to sample from the posterior distribution.

We now explain the graphical model in Figure 3.2, highlighting the way in which it addresses important psychological problems, including accounting for contaminants in data, relating the physical and psychological

properties of stimulus, and allowing for trial-to-trial variability in performance.

3.3.2.1 Latent classes

The Bayesian approach makes it easy to apply latent predictors to data. In the model in Figure 3.2, we have assumed that there are three types of experimental trials: (1) Diffusion trials (with probability $1-\pi$), (2) guesses (probability $\pi(1-\gamma)$), and (3) delayed startups (probability $\pi\gamma$). A similar distinction was applied by Vandekerckhove and Tuerlinckx (2007). This categorical distinction is latent because we have no direct measures of class membership. Most powerfully, the Bayesian approach allows us to estimate each trial's probability of membership to each of these (mutually exclusive) classes, so that we can identify specific trials that might be contaminants. Class memberships are indicated by two binary variables, $p_{ijk} \sim \text{Bernoulli}(\pi)$ and $g_{ijk} \sim \text{Bernoulli}(\gamma)$.

Applying this latent class assignment implies that we assume that three distinct psychological processes account for the data. The first is a typical diffusion process. The second is a diffusion process devoid of relevant information (a *guess*); That is, the participant has not gained any information from the stimulus and the response is therefore at chance level. In terms of diffusion model parameters, this translates to the assumption that all $\xi_{ijk} = 0$ if $p_{ijk}(1 - g_{ijk}) = 1$. The third psychological process is one of *delayed startups*, where trials have a different non-decision time.

3.3.2.2 The non-linear regression of drift rate

In psychophysics, it is common to perform nonlinear regression to model the effect of stimulus dimensions on accuracy, often using a Weibull link function. Because it seems natural that drift rates have similar asymptotic

behavior as a function of stimulus intensity, our model applies a Weibull. For the i^{th} brightness condition ($i = 1, \ldots, 33$) and j^{th} speed/accuracy instruction, then

$$v_{ij} = v^{lo} + \left(v^{hi} - v^{lo}\right) \times \left(1 - \exp\left[-\left(i/v^{sc}\right)^{v_j^{sh}}\right]\right). \tag{3.1}$$

Note that we allow the shape parameter v^{sh} to be different between the speed/accuracy conditions. This is contrary to Ratcliff and Rouder (1998), who assumed mean drift rates to be equal for equal stimulus intensities.

3.3.2.3 Variability in performance

In order to extend the Wiener distribution to the Ratcliff diffusion model, the graphical model in Figure 3.2 implements a mixed-model version of the Wiener distribution. This means that, from trial to trial, some parameters are conditionally independent draws from a mixing distribution. By conceptualizing the Ratcliff diffusion model in this way, we can avoid the computationally intense integrals described in Tuerlinckx (2004), and approximate the integrals using standard MCMC computational methods used to integrate over the posterior. The simplification offered by this approach allows us to choose theoretically plausible mixing distributions, so we choose a Gaussian mixing distribution for drift rate and a truncated Gaussian for nondecision time.

This combination of assumptions in the graphical model can be for-

mally stated as follows:

$$\begin{cases} \text{if } p_{ijk} = 0 & \begin{cases} t^*_{ijk} \sim \text{Wiener}(a_j, t^{er}_{ijk}, a_j/2, \xi_{ijk}) \\ \xi_{ijk} \sim \text{N}(v_{ij}, \eta^2) \\ t^{er}_{ijk} \sim \text{TN}_{(0,+\infty)}\left(T^{er}, (\sigma^{er})^2\right) \end{cases} \\ \text{if } \begin{matrix} p_{ijk} = 1 \\ g_{ijk} = 1 \end{matrix} & \begin{cases} t^*_{ijk} \sim \text{Wiener}(a_j, t^{ds}_{ijk}, a_j/2, \xi_{ijk}) \\ \xi_{ijk} \sim \text{N}(v_{ij}, \eta^2) \\ t^{er}_{ijk} \sim \text{TN}_{(0,+\infty)}\left(T^{ds}, (\sigma^{ds})^2\right) \end{cases} \\ \text{otherwise} & \begin{cases} t^*_{ijk} \sim \text{Wiener}(a_j, t^{er}_{ijk}, a_j/2, 0) \\ t^{er}_{ijk} \sim \text{TN}_{(0,+\infty)}\left(T^{er}, (\sigma^{er})^2\right) \end{cases} \end{cases},$$

where N and TN stand for normal and truncated normal distributions, respectively.

3.3.3 Results

All of our analyses are based on 40,000 posterior samples collected after a burn-in of 10,000 samples. First, we investigate recovery of the model by inspecting posterior predictive samples (we limit ourselves to data of participant KR, but results were similar for the others). The two panels in Figure 3.3 show the proportion of 'bright' responses in the data (open circles) and as recovered by the model (grey dots; the full line connects the mean predictions). Similarly, in Figure 3.4, we show posterior predictives of the 10^{th}, 30^{th}, 50^{th}, 70^{th}, and 90^{th} percentiles of each RT distribution. In all panels, it is clear that the model recovers the patterns in the data quite well. The exception is the 10^{th} RT percentile in the speed condition, which the model consistently overestimates. This may be due to our restriction that $b = .5$ and does not vary from trial to trial.

Looking at the posterior means and standard deviations for the standard Ratcliff diffusion model parameters (for participant KR) in column

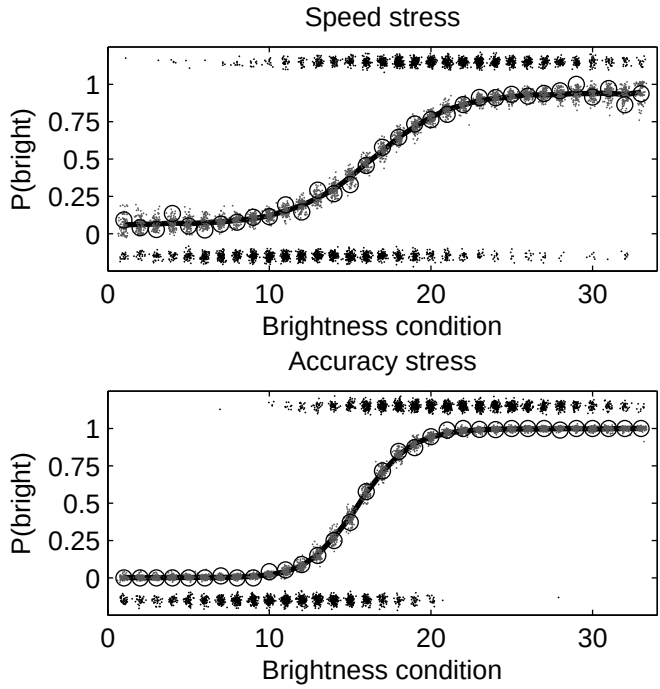


Figure 3.3: *Posterior predictive proportions of 'bright' responses, as a function of stimulus intensity (participant KR). Grey dots indicate* 100 *posterior samples, open circles indicate the proportions in the data set. Black dots at top and bottom indicate observed data points (jittered). Thick dark lines connect the posterior mean estimate of the response probabilities in each condition.*

A of Table 3.1, we see that the boundary separation parameter a is much smaller in the speed-instruction condition ($a_1 = 0.05$), as expected. The Weibull asymptote parameters, as well as the scale parameter, get sensible mean posterior values. Interestingly, the shape parameter is somewhat different between the two instruction conditions, with steeper Weibull functions in the speed-stress condition.

Figure 3.5 shows posterior distributions of the π, γ, and v^{sh}, for each participant. The difference in v^{sh}s is small in two participants, but large

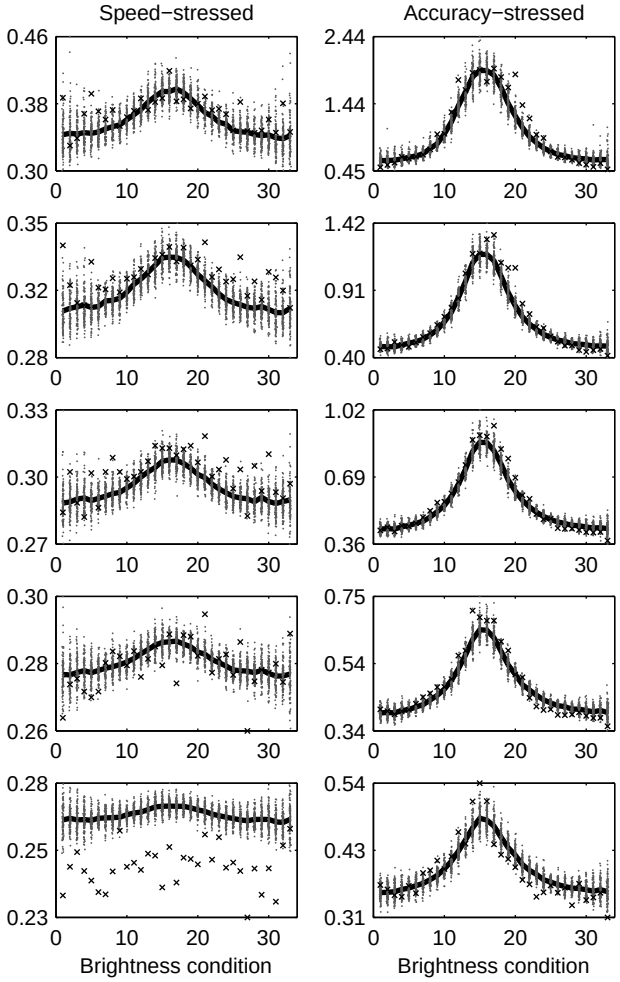


Figure 3.4: *Posterior predictive RT percentiles (KR). Left panels are for the speed-stressed condition. The first through fifth row are for the* 90^{th}, 70^{th}, 50^{th}, 30^{th}, *and* 10^{th} *percentiles, respectively. The values of the RT quantiles are shown on the vertical axes. Grey dots indicate* 100 *posterior samples, the dark lines connect posterior means. Empirical percentiles are shown by crosses.*

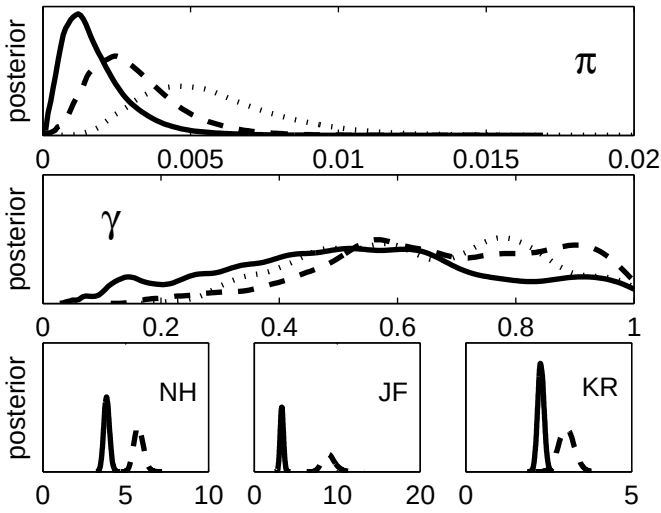


Figure 3.5: *Posterior density plots for parameters π, γ, and v_j^{sh}. In the top two panels (π and γ), line styles indicate participants (KR: dotted, JF: dashed, NH: full). In the bottom panels, v_1^{sh} (dashed, for the speed condition) and v_2^{sh} (full line, for the accuracy condition) are shown for each participant separately. Significant differences are visible.*

for JF, and it seems consistent between participants.[3]

Participant KR has the highest π parameter—the posterior mean is about .006. Looking at this participant's γ parameter, we see that there is much uncertainty regarding the proportion of guesses (because this parameter pertains to only .6% of the data—43 trials), but there are likely more delayed start-ups (28) than guesses (15). The delayed start-up trials are on average 858 ms slower than regular trials. If we compare the first two columns in Table 3.1, it appears that accounting for contaminants in this data set makes little difference for the mean estimates of the parame-

[3]The change in the Weibull shape can be interpreted as an effect of an adaptation in the participant's strategy. However, the present data do not allow us to speculate further on the cause of this effect.

ters. The posterior uncertainty of the drift-related parameters, however, is higher in the more complicated contaminant-mixture model.

3.4 Sample size

Typically, applying the Ratcliff diffusion model requires "a fair amount [sic] of data for accurate estimation of its parameters" (Wagenmakers, in press). By constraining parameters across conditions and using a Bayesian approach with modern computational sampling methods, we expect the need for large data sets to be alleviated. To test this possibility, we conducted analyses based on subsampling from the benchmark data, and comparing the results with the results from the full data set.

To subsample from the original data set, we sampled—without replacement—either 2, 5, 10, or 20% of the data points for participant KR; thus approximately preserving the relative number of data points in each condition. We then applied a model that is similar to the one described in the previous section (see Fig. 3.2), but we leave out the contaminant modeling ($\pi = 0$) because of the low proportions of contaminants found. We drew $5,000$ samples from the joint posterior, after a burn-in of $5,000$. For each parameter, we compute the posterior mean. This procedure was repeated 20 times for each proportion, with new subsamples each time. Then, with the mean posterior estimates resulting from each of the smaller data sets, we computed the squared relative bias R^2 for each parameter: $R^2_\theta = \left((\theta - \hat{\theta})/\theta\right)^2$, where θ indicates the parameter as estimated from the full data set (with the same model containing no contaminant component) and $\hat{\theta}$ as estimated from the smaller data set. We summed the R^2s for each parameter set to obtain a measure of how close the recovered parameters of each subsample were to the ones found from the full data set. From each downsampling proportion (2, 5, 10, and 20%), we then chose the results

that gave the median recovery under the R^2 criterion, and report those results in Table 3.1.

As can be seen, most of the estimates from the reduced data sets are very similar to those inferred from the full data set, and they certainly preserve all of the important order relations and trends in the parameter values across conditions. With few data, posterior uncertainty is very large. As expected from statistical theory, the posterior standard deviations scale up with a factor $\sqrt{N_t/N_s}$, where N_t is the total sample size and N_s the size of the subsample.

3.5 Conclusions

In this paper, we demonstrated a Bayesian extension of the popular Ratcliff diffusion model. In a single example, we combined a psychophysical link function and latent class assignment to revisit the benchmark data set of Ratcliff and Rouder (1998). As part of the Bayesian method, we employed posterior predictive checks (shown in figures 3.3 and 3.4) of the model. We found that few of the data points are contaminants. Interestingly—and in contrast to previous analyses—we also found differences in drift rate as an effect of task instruction. In particular, drift rate as a function of stimulus quality increases more steeply under speed-stress than it does under accuracy-stress. In addition, we reported a simple numerical experiment that showed that relatively small samples can yield satisfactory parameter estimates. This suggests that the Ratcliff diffusion model may, using Bayesian methods, be applied to smaller data sets than was previously practicable.

We think many of the specific demonstrations in our example correspond to general points regarding the usefulness of Bayesian statistical methods for understanding process models in the cognitive sciences. At the

Table 3.1: *Some results for participant KR. Posterior means in top half; Posterior standard deviations in bottom half. The rightmost column describes the distrbitution of the R^2_θ criterion for the 5% samples. R^2_θ is generally small with little variability, except for η and v_2^{sh}. All SDs have been multiplied by 100.*

	A^*	B^*	2%	5%	10%	20%	$R^2_{\theta;5\%}$
T^{er}	0.25	0.25	0.23	0.26	0.25	0.25	0.00
a_1	0.05	0.06	0.07	0.05	0.05	0.05	0.04
a_2	0.21	0.21	0.20	0.22	0.24	0.23	0.01
η	0.11	0.12	0.08	0.14	0.16	0.16	0.21
σ^{er}	0.03	0.03	0.04	0.04	0.03	0.04	0.04
v^{hi}	0.59	0.57	0.43	0.67	0.77	0.71	0.05
v^{lo}	-0.55	-0.53	-0.63	-0.74	-0.62	-0.60	0.05
v^{sc}	0.57	0.56	0.48	0.56	0.63	0.60	0.00
v_1^{sh}	3.02	3.07	4.89	2.39	2.20	2.75	0.05
v_2^{sh}	2.26	2.33	1.83	1.87	1.95	2.32	0.58
T^{er}	0.14	0.14	1.11	0.73	0.42	0.34	0.17
a_1	0.08	0.08	0.67	0.41	0.25	0.20	4.80
a_2	0.32	0.32	2.10	1.56	1.18	0.89	2.12
η	0.69	0.66	5.07	3.40	2.18	1.83	20.66
σ^{er}	0.08	0.08	0.70	0.36	0.25	0.19	4.22
v^{hi}	3.72	2.88	8.73	12.75	8.71	9.45	4.88
v^{lo}	1.95	1.80	12.83	8.66	6.70	4.95	5.50
v^{sc}	1.34	1.04	3.22	4.89	3.59	3.09	0.43
v_1^{sh}	24.84	22.80	241.43	71.83	40.83	54.36	5.78
v_2^{sh}	9.77	9.40	49.30	29.10	25.52	24.62	94.62

* A is with outlier treatment; B is without outlier treatment.

most general level, the Bayesian framework for scientific inference allows enormous freedom in building process models. All that is required is a formal probabilistic account of how observed data are generated. Once this modeling has been done, and data are available, making inferences is the (conceptually) easy process of reversing the generative process, and inferring which combinations of parameters are likely to have given rise to the data. Important issues like balancing goodness-of-fit with complexity, assessing sensitivity to prior information, conditioning on nuisance variables,

and so on, are all dealt with completely and coherently because Bayesian inference has a principled basis in probability theory.

More practically, Bayesian methods, especially through the use of graphical models or other languages that permit the use of modern computational methods for posterior sampling, make it straightforward to undertake analyses that are psychologically rich, but otherwise difficult to implement. For example, mixture models—including especially latent assignment models—allow data in a task to be modeled as having been generated by more than one psychological process. Complex regression structures are straightforward to implement and variability across trials is easily formalized in a Bayesian account.

The ability of Bayesian methods in our example to extend the scope of well-developed and widely-used diffusion process accounts of decision-making is very promising. It suggests that Bayesian methods can be applied widely to process models throughout the cognitive sciences, broadening the set of psychological questions these models can be used to answer.

References

Bogacz, R., Brown, E., Moehlis, J., Holmes, P., & Cohen, J. D. (2006). The physics of optimal decision making: A formal analysis of models of performance in two-alternative forced choice tasks. *Psychological Review*, *113*, 700–765.

Griffiths, T. L., Kemp, C., & Tenenbaum, J. B. (in press). Bayesian models of cognition. In R. Sun (Ed.), *Cambridge handbook of computational cognitive modeling.* Cambridge, UK: Cambridge University Press.

Laming, D. R. J. (1968). *Information theory of choice-reaction times.* London, UK: Academic Press.

Lee, M. D. (2008). Three case studies in the Bayesian analysis of cognitive models. *Psychonomic Bulletin & Review*, *15*, 1–15.

Lunn, D., Thomas, A., Best, N., & Spiegelhalter, D. (2000). WinBUGS — a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing*, *10*, 325–337.

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, *85*, 59–108.

Ratcliff, R., & Rouder, J. N. (1998). Modeling response times for two-choice decisions. *Psychological Science*, *9*, 347–356.

Smith, P. L., & Ratcliff, R. (2004). The psychology and neurobiology of simple decisions. *Trends in Neurosciences*, *27*, 161–168.

Tuerlinckx, F. (2004). The efficient computation of the distribution function of the diffusion process. *Behavior Research Methods, Instruments, & Computers*, *36*, 702-716.

Vandekerckhove, J., & Tuerlinckx, F. (2007). Fitting the Ratcliff diffusion model to experimental data. *Psychonomic Bulletin & Review*, *14*, 1011–1026.

Vandekerckhove, J., & Tuerlinckx, F. (2008). Diffusion model analysis with MATLAB: A DMAT primer. *Behavior Research Methods*, *40*, 61–72.

Voss, A., & Voss, J. (2007). Fast-dm: A free program for efficient diffusion model analysis. *Behavior Research Methods*, *39*, 767–782.

Wagenmakers, E.-J. (in press). Methodological and empirical developments for the Ratcliff diffusion model of response times and accuracy. *European Journal of Cognitive Psychology*.

Cognitive psychometrics

CHAPTER 4

Hierarchical diffusion models for two-choice response times

Abstract

Two-choice response times are a common type of data, and much research has been devoted to the development of process models for such data. However, the practical application of these models is notoriously complicated and flexible methods are largely nonexistent. We combine a popular model for choice response times—the Wiener diffusion process—with techniques from psychometrics in order to construct a *hierarchical diffusion model*. Chief among these techniques is the application of random effects, with which we allow for unexplained variability among participants, items, or other experimental units. These techniques lead to a modeling framework that is highly flexible and easy to work with. Among the many novel models this statistical framework provides are a *multilevel diffusion model*, *regression diffusion models*, and a large family of *explanatory diffusion*

models. We provide examples and the necessary computer code.

4.1 Introduction

In his 1957 Presidential Address before the APA Meeting, Lee Cronbach drew a captivating sketch of the state of psychology at the time. He focused on the two distinct disciplines that existed in the field of scientific psychology at the time. On the one side, there was the experimental discipline that concerned itself with the systematic manipulation of conditions in order to observe the consequences. On the other side, there was the correlational discipline, which focused itself on the study of preexisting differences between individuals or groups. Cronbach saw many potential contributions of these disciplines to one another, and argued that the time and opportunity had come for the two dissociated fields to crossbreed: "We are free at last to look up from our own bedazzling treasure, to cast properly covetous glances upon the scientific wealth of our neighbor discipline. Trading has already been resumed, with benefit to both parties" (Cronbach, 1957, p. 675). Two decades onward, Cronbach (1975) saw the hybrid discipline flourishing across several domains.

In the area of measurement of psychological processes, a similar schism still exists today. Psychological measurement and individual differences are studied in the domain of psychometrics, while cognitive processes are the stuff of the more nomothetic mathematical psychology. In both areas, statistical models are used extensively. There are the common models based on the (general) linear model such as ANOVA and regression but we will focus on more advanced, nonlinear techniques.

Experimental psychology has, for a long time, made use of process models to describe interesting psychological phenomena in various fields. Some famous examples are Sternberg's (1966) Sequential Exhaus-

tive Search Model for visual search and memory scanning, Atkinson and Shiffrin's (1968) Multistore Model for memory, multinomial processing tree models for categorical responses (Batchelder & Riefer, 1999; Riefer & Batchelder, 1988) and the general family of sequential sampling models for choice response times (Laming, 1968; Link & Heath, 1975; Ratcliff & Smith, 2004). A property shared by these process models is that they give detailed accounts of underlying response processes. Such models are typically applied to data from single participants, and they are very successful in fitting empirical data.

In the correlational area, however, measurement models are dominant. Most well-known among these is the factor analysis (FA) model, but models from item response theory (IRT) belong to this class as well. In the past decade, a lot of work has appeared showing the relationships between FA, IRT, and multilevel models. Rijmen, Tuerlinckx, De Boeck, and Kuppens (2003) show that many IRT models are generalized linear mixed models (GLMM) and the rest are nonlinear mixed models (NLMM; see also De Boeck & Wilson, 2004). Skrondal and Rabe-Hasketh (2004) offer an encompassing framework for FA models, IRT models, and multilevel models (called generalized linear latent and mixed models or GLLAMM). The models that originated in correlational research are used to model individual differences. Often such models are less detailed and more general than the models discussed in the previous paragraph, but they are able to locate the main sources of individual differences.

Recently, some convergence between the experimental and the correlational areas has emerged. Batchelder and Riefer (1999; see also Batchelder, 1998; Riefer, Knapp, Batchelder, Bamber, & Manifold, 2002) have introduced the concept of *cognitive psychometrics*. In cognitive psychometrics, models from cognitive psychology are used to capture specific interesting aspects of the data. These models typically assume that the data have

been gathered with a specific paradigm (e.g., that they are binary choice response times). While this necessarily makes the models less general than multipurpose statistical models, it provides the advantage of providing substantive insight in the data. Furthermore, ideas of hierarchical modeling have recently been introduced into the area of cognitive modeling, most notably by Rouder and colleagues (e.g., Rouder & Lu, 2005; Rouder, Lu, Speckman, Sun, & Jiang, 2005; Rouder et al., 2007), who use hierarchical models as a statistical framework for inference, and also by Tenenbaum and colleagues (e.g., Chater, Tenenbaum, & Yuille, 2006; Griffiths, Kemp, & Tenenbaum, 2008), who use hierarchical models as an account of the organization of human cognition.

Extending cognitive models to hierarchical models (or vice versa) is an important part of the trading between disciplines that Cronbach (1957) advocated. The benefits of the trade do go both ways: By extending process models hierarchically, experimental psychologists who use these models can take individual differences into account and are in a better position to explain such individual differences. Correlational psychologists, on the other hand, could apply measurement models that are built upon firmly validated process models, often grounded in substantive theory.

In the present paper, we aim to integrate both traditions further by extending hierarchically an important and popular process model, the diffusion model for two-choice response times. Even though choosing the diffusion model as our measurement level bears with it a number of implementation difficulties, we choose this model because of the interesting psychological interpretability of its parameters, which we will explain in the next section. Additionally, choice response times—the combination of reaction time and accuracy data—are ubiquitous in experimental psychology, and we believe that a hierarchical extension of the diffusion model could be of considerable value to the field. In addition, a Bayesian ap-

proach is taken to fit the hierarchical extension of the diffusion model. Details on the practical implementation will provided as well.

In the sections that follow, we will introduce diffusion model for two-choice response times, and then provide a detailed account of the hierarchical extension to the diffusion model. Then, we will describe three, progressively more complex, example applications. We conclude with a discussion of our approach and of further possible applications.

4.2 The diffusion model

The diffusion model as a process for speeded decisions starts from the basic principle of *accumulation of information* (Laming, 1968; Link & Heath, 1975). When an individual is asked to make a binary choice on the basis of an available stimulus, the assumption is that evidence from the stimulus is accumulated over (continuous) time and a decision is made as soon as an upper or lower boundary is reached. Which boundary is reached determines which response is given. The basic form of this model is often referred to as the *Wiener diffusion model with absorbing boundaries*.

Figure 4.1 depicts the Wiener diffusion process, and shows the main parameters of the process. On the vertical axis there are the boundary separation α,[1] indicating the evidence required to make a response (i.e., speed-accuracy trade-off) and the initial bias β, indicating the a-priori status of the evidence counter as a proportion of α. If β is less than 0.5, this indicates bias for the response represented by the lower boundary. The absolute value of the starting position is $\alpha\beta = \zeta_{init}$, but we will generally not use this value. The arrow represents the average rate of information uptake or drift rate δ, which indicates the average amount of evidence that

[1] Throughout, we will use Greek letters to indicate unobserved parameters and Latin letters for running indexes or observed variables.

the observer receives from the stimulus at each sampling. (The amount of variability in these samples, which makes the process stochastic, is a scaling constant that is typically set to 0.1 in the literature) Finally, the short dashed line indicates the nondecision time τ, the time used for everything except making a decision (i.e., encoding the stimulus and physically executing the response). Probability density functions associated with the diffusion model can be found in Appendix 4.A.

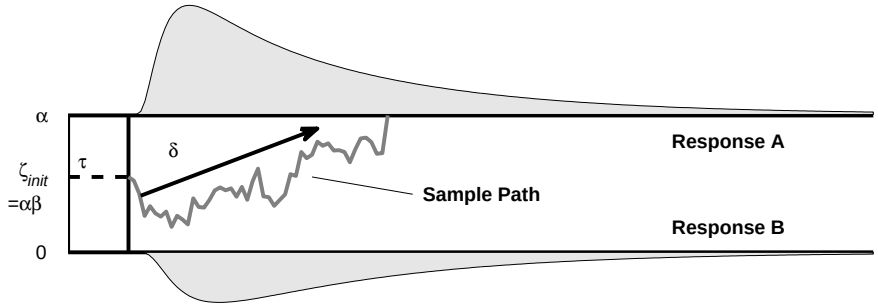


Figure 4.1: *A graphical illustration of the Wiener diffusion model.*

The diffusion model owes much of its current popularity to the work of Ratcliff and colleagues (e.g., Ratcliff, 1978; Ratcliff & Rouder, 1998; Ratcliff, Van Zandt, & McKoon, 1999; Ratcliff & Smith, 2004). An important contribution Ratcliff made was to incorporate trial-to-trial variance into the Wiener diffusion model, so that the parameters β, δ, and τ are not constant but vary from trial to trial. This conceptually significant extension has performed so remarkably well in the analysis of two-choice response time data that it is now sometimes referred to as the Ratcliff diffusion model (Vandekerckhove & Tuerlinckx, 2007; Wagenmakers, in press). It has successfully been applied to data from experiments in many different fields, such as memory (Ratcliff, 1978; Ratcliff & McKoon, 1988), letter

matching (Ratcliff, 1981), lexical decision (Ratcliff, Gomez, & McKoon, 2004; Wagenmakers, Ratcliff, Gomez, & McKoon, 2007), signal detection (Ratcliff & Rouder, 1998; Ratcliff, Thapar, & McKoon, 2001; Ratcliff et al., 1999), visual search (Strayer & Kramer, 1994), and perceptual judgment (Eastman, Stankiewicz, & Huk, 2007; Ratcliff, 2002; Ratcliff & Rouder, 2000; Thapar, Ratcliff, & McKoon, 2003; Voss, Rothermund, & Voss, 2004). The Ratcliff diffusion model is also one of very few models that succeed in explaining all of the "benchmark" characteristic aspects of two-choice response time data—such as different response time distributions for correct and error responses, both of them positively skewed and the relation between their means dependent on parameters, with some minimum value below which there is no mass (Brown & Heathcote, in press). In addition, the model has passed selective influence tests for its main parameters (Voss et al., 2004), in which experimental manipulations are shown to affect only the relevant model parameters (e.g., changing from speed to accuracy instructions affects only the boundary separation parameter).

4.3 A hierarchical framework for the diffusion model

4.3.1 Motivation

There are several motivations for making a hierarchical extension of a substantively generated model such as the diffusion model. The first and most important motivation involves the type of data set to which the diffusion model has typically been applied. Traditional applications of the diffusion model have been restricted to single participants (see, e.g., Ratcliff & Rouder, 1998), and there has generally been no motivation to model inter-individual differences in the decision process. The dearth of investigation into individual differences in when applying process models is reminiscent

of the schism between the experimental and correlational subdisciplines that Cronbach (1957, cf. supra) pointed out.

More recently, however, the diffusion model has been applied to study individual differences (e.g., Klauer, Voss, Schmitz, & Teige-Mocigemba, 2007; Ratcliff et al., 2004; Wagenmakers et al., 2007). The typical approach in such cases is to run multi-step analyses: In a first step a specific model is fitted to data from each individual, and then inferences regarding individual differences are made on the basis of summary measures of the parameter estimates. An example of this approach can be found in Klauer et al. (2007), where individual participants' parameter estimates are subjected to second-stage analysis using analysis of variance.

However, data do not always allow for separate analyses per individual: Estimating the diffusion model's parameters typically requires a large number of data points (Wagenmakers, in press) and in many experimental contexts it may be impractical or even impossible to obtain many data points within each participant. In particular, when studying higher-level cognitive processes or emotions the stimulus material may simply not allow for the generation of hundreds of trials or for presenting stimuli more than once (e.g., Brysbaert, Van Wijnendaele, & De Deyne, 2000; Klauer et al., 2007). Often, however, there are many participants in the sample. In cases such as these, it is natural to be interested in individual differences but it is impossible to analyze the data separately for each participant, and the multi-step procedure cannot be applied.

Another problem with the multi-step procedures is that one may want to constrain parameters to be equal across participants. In this case, an analysis needs to involve all subjects simultaneously, allowing some of the parameters to differ and others to be equal. However, such an approach may lead to a prohibitively large number of parameters. As will be argued below, a hierarchical approach may offer a solution by formalizing

individual differences in a specific process model framework.

4.3.2 Uses of the hierarchical diffusion model

In a hierarchical model, it is assumed that participants are a randomly drawn sample from some partly specified population (see also Gelman & Hill, 2007). Each individual participant has their own set of parameters, and since these participants are typically randomly selected from some larger population, the differences in parameter values between participants can be seen as a random effect in the statistical sense.[2] In this way, individual differences can be explicitly permitted in a hierarchical model. However, not only the person-specific parameters are important, but also the unknown characteristics of their population distributions, such as the means, variances, and covariances, the latter two of which are indications of the magnitude (i.e., importance) of individual differences.[3] In a hierarchical framework, it is relatively easy to construct models in which some parameters are constrained to be equal across participants, while others may vary from individual to individual. Hierarchical models are ideally suited to handle the data sets with few trials per participant (discussed above), even in the case where single individuals do not provide enough information to estimate all model parameters. Hierarchically extending

[2] A *random effect* occurs when experimental units are randomly drawn, interchangeable samples from a larger population. This applies not only to participants, but may apply to items, trials, blocks, and other units as well, as long as they are interchangeable samples. If the selected units comprise the entirety of the relevant population (about which we want to make inferences), then a *fixed effect* is appropriate.

[3] While it may seem that such an approach leads to even more parameters than when no population assumptions are made, invoking the population assumption actually reduces the number of *effective parameters* because it acts as a constraint on the person-specific parameters (this effect is in some cases also called *shrinkage to the mean*). A limiting case is when the variance of the population distribution is zero such that there are no individual differences and all person-specific parameters are exactly equal to the mean. Moreover, shrinkage is stronger for parameters of individuals that provide less information. For more information on hierarchical modeling and shrinkage we refer to Gelman and Hill (2007).

the diffusion model leads to what we call the hierarchical diffusion model (HDM).[4]

Hierarchical models have proven useful in many areas of research. Some selected domains include psychological measurement where item response models have been used (e.g., De Boeck & Wilson, 2004), educational measurement and school effectiveness studies (Raudenbush & Bryk, 2002), and longitudinal data analysis in psychology (Singer & Willett, 2003) and biostatistics (Molenberghs & Verbeke, 2006; Verbeke & Molenberghs, 2000).

In this paper, we rely particularly on the framework proposed by De Boeck and Wilson (2004) for item response models. In their book, De Boeck and Wilson (2004) sharply distinguish between *describing* and *explaining* individual differences. Describing individual differences refers to the possibility of assuming population distributions for certain parameters and estimating some characteristics of these distributions. In such an approach, we merely acknowledge that differences between persons exist and we quantify the variability in the population (through the variances of the population distributions). However, in any scientific enterprize, the ultimate goal is not to simply observe differences, but to attempt to explain why they occur. Individual differences can be explained by relating the person-specific parameters to predictors (see below). In doing so, we consider the variability in the population as to-be-explained, and by including a predictor in the model, we explicitly intend to decrease this unexplained variability.

It is important to emphasize that, while the above discussion was centered on differences between *persons*, an HDM can equally well be applied to populations of items, trials, or indeed any experimental unit (including

[4]There is some ambiguity here about the word "model". In one sense the diffusion model is a process model, and the hierarchical extension is a statistical modeling tool. It is the combination of these two aspects, however, that makes the HDM a powerful framework.

subgroups within populations, or items nested in conditions). Variability across these other experimental units can be captured in exactly the same way as variability across persons. The example applications will make extensive use of this ability of HDMs.

The main difference between the approach of De Boeck and Wilson (2004) and our framework is that De Boeck and Wilson work within a context of item response models: The data they consider are binary (or polytomous) responses of persons to a set of items. These item response models are logistic regression models or extensions and generalizations thereof that relate the responses (or more correctly: the probability of a certain response) to an underlying latent trait (i.e., the individual difference variable). There, the logistic regression model can be considered as the measurement model. In our case, the data are bivariate (choice response and reaction time) and the *measurement level* is the Wiener diffusion model which is considerably more complex.

In the remainder of this section, we will further elaborate and apply the framework of De Boeck and Wilson (2004) to the diffusion model. This will be done by defining several basic building blocks that may be combined with the diffusion model in order to arrive at a hierarchical diffusion model capable of describing and explaining interindividual differences. As it turns out, not only interindividual differences may be tackled in such a way but other sources of variation as well. Before doing so, however, we will define some notation.

4.3.3 Notation

Suppose a person p (with $p = 1, \ldots, P$) is observed in condition i (with $i = 1, \ldots, I$) on trial j (with $j = 1, \ldots, J$) and their choice responses (corresponding to the absorbing boundaries) and response times are recorded,

denoted by the random variables $X_{(pij)}$ and $T_{(pij)}$, respectively (realizations of these random variables are $x_{(pij)}$ and $t_{(pij)}$). We will sometimes write $\boldsymbol{Y}_{(pij)}$ and $\boldsymbol{y}_{(pij)}$ to refer to the random vector $(X_{(pij)}, T_{(pij)})$ and the vector of realizations $(x_{(pij)}, t_{(pij)})$, respectively. It will be said that $\boldsymbol{Y}_{(pij)}$ is distributed according to a Wiener distribution:

$$\boldsymbol{Y}_{(pij)} \sim W\left(\alpha_{(pij)}, \beta_{(pij)}, \tau_{(pij)}, \delta_{(pij)}\right).$$

We use "Wiener distribution" as shorthand for the joint density function of hitting the boundary $X_{(pij)}$ at time $T_{(pij)}$. The distribution is characterized by four basic parameters (explained above in the section of the diffusion model) which here carry a triple index which means that, in principle, they can differ across persons, conditions and trials. In some of the examples, we will add additional indexes to allow more nuanced differences. To avoid confusion with other subscripts, indexes are always put between parentheses.

Finally, it should be noted that we will often "recycle" symbols for new models or new examples, so that a symbol used in one model may be redefined in another model to refer to something else.

4.3.4 Model building blocks

Based on the framework of De Boeck and Wilson (2004), we discern three types of useful model building blocks: levels of random variation, manifest predictors, and latent predictors. These three aspects will be discussed in turn. In order to render the discussion more concrete, we will illustrate the theoretical concepts with the drift rate parameter of the diffusion model. We choose to limit the illustrations to a single parameter for reasons of clarity but a similar story can be told for the other parameters, as will become obvious when we move to the applications later in the paper.

4.3.4.1 Levels of random variation

The data may consist of different levels of hierarchy. We have already implicitly referred to the simplest case when talking about individual differences: Individual differences can only be investigated when a sample of individuals is measured repeatedly. In such a case, the data consist of two levels: At the higher level there are the individuals and at the lower level, the measurements within the persons.

As an example, consider drift rate $\delta_{(pij)}$. Assume that a set of persons are presented with a series of stimuli in a single condition (such that we may drop the index i). The drift rate $\delta_{(pj)}$ can then be written as follows:

$$\delta_{(pj)} = \nu_{(p)} + \varepsilon_{(pj)} \tag{4.1}$$

where $\varepsilon_{(pj)} \sim N\left(0, \eta_\varepsilon^2\right)$ and $\nu_{(p)} \sim N\left(\mu_\nu, \sigma_\nu^2\right)$, with $\varepsilon_{(pj)}$ and $\nu_{(p)}$ independent. Here, the variance η_ε^2 represents trial-to-trial variability in drift rate within a person. This example is akin to the assumption of trial-to-trial variability made by (Ratcliff, 1978). The parameter μ_ν is the population average of individual drift rates and σ_ν^2 is the variance of individual drift rates in the population. The importance of individual differences can be judged by comparing σ_ν^2 to η_ε^2: If σ_ν^2 is much larger than η_ε^2, this means that there are sizeable individual differences, which is not the case if σ_ν^2 is much smaller than η_ε^2. There exist several alternative ways of writing the model in Equation 4.1. For instance, one could include the population average μ_ν directly into the linear decomposition (i.e., $\delta_{(pj)} = \mu_\nu + \sigma_\nu \nu_{(p)} + \eta_\varepsilon \varepsilon_{(pj)}$) and assuming a mean of zero and unit variance for all distributions.

Equation 4.1 can be extended readily to include fixed condition effects as follows:

$$\delta_{(pij)} = \gamma_{(i)} + \nu_{(p)} + \varepsilon_{(pij)} \tag{4.2}$$

where $\gamma_{(i)}$ is a fixed condition effect. Hence, the mean drift rate in condition i for a person p depends on a fixed condition effect $\gamma_{(i)}$ and a random person effect $\nu_{(p)}$. A related model has been proposed earlier by Ratcliff (1985) and Tuerlinckx and De Boeck (2005).

Because individual differences are the main motivation for developing a hierarchical diffusion model, we have thus far restricted the hierarchical structure to trials nested within persons (conditions are viewed as fixed effects). However, there is no reason to stop there if there is a sound reason for more complex forms of levels of random variation. For example, persons may be nested in groups and those groups nested in larger groups. In such a case, there are more than the traditional two levels in the data.

In addition, there is no reason to allow random effects only at the person side. At the condition or item side, it can make sense to allow for condition or random effects. In the types of applications we envision for the HDM, the stimulus material often consists of words or pictures. In psycholinguistics, for example, there has been some controversy over the modeling of word effects. In a seminal paper, Clark (1973) strongly argued that stimulus words should be considered as randomly sampled from a population distribution as well. In such cases, the parameter $\gamma_{(i)}$ in Equation 4.2 can also be assumed to follow a normal distribution with mean μ_γ and variance σ^2_γ. This would yield a *crossed random effects* design (e.g., Gonzalez, Tuerlinckx, & De Boeck, in press; Janssen, Tuerlinckx, Meulders, & De Boeck, 2000; Rouder et al., 2007). Similarly, conditions or items could be nested in categories which are in turn nested in larger categories.

4.3.4.2 Manifest predictors

By identifying and including levels of variation in the analyses, we describe individual differences or, if there are random item effects, differences between stimuli. We call this type of analysis *descriptive* because we are merely observing how the variability in the data is distributed among several sources. However, in a next step we want to *explain* the variability in parameters by using manifest predictors (continuous and/or discrete). More broadly, interindividual, interstimulus or, less intuitively, intertrial variability (represented in random effects and their population variances) might be explained by regressing basic parameters on known *predictors* or *covariates*.

As an example of explaining interindividual variability, assume that the drift rate is person-specific and that there is a person covariate such as age available (with $A_{(p)}$ being the age of person p). We could then adopt the following model for the drift rate:

$$\delta_{(pij)} = \gamma_{(i)} + \beta_0 + \beta_1 A_{(p)} + \nu_{(p)} + \varepsilon_{(pij)} \tag{4.3}$$

where β_0 and β_1 are the regression coefficients of the simple linear regression of $\delta_{(pij)}$ on $A_{(p)}$ and $\nu_{(p)}$ is a person-specific error term with distribution $\nu_{(p)} \sim N\left(0, \sigma_\nu^2\right)$. The other parameters are defined as in Equation 4.2.

Alternatively, we may try to use covariates in order to explain some of the variability between items. For example, differences in recognizability between words may be related to their frequency of use.

In sum, working with manifest predictors in the hierarchical diffusion model means building a regression model for a random effect with known predictors but unknown regression coefficients. Explaining variability in parameters through covariates will be an important theme in several of the examples in this paper.

4.3.4.3 Latent predictors

As shown by De Boeck and Wilson (2004), predictors do not necessarily need to be manifest; they may also be latent. That is, they are unobserved but inferred from the data. For a simple example, take a binary predictor that we will assume to be observed for the moment. Given this predictor, drift rate could be decomposed as follows

$$\delta_{(pij)} = \gamma_{(i)} + \beta_0 + \beta_1 Z_{(p)} + \nu_{(p)} + \varepsilon_{(pij)}, \tag{4.4}$$

where $Z_{(p)}$ is 0 for some persons and 1 for the others. Because $Z_{(p)}$ is a binary predictor, for a person p whose $Z_{(p)} = 0$, drift rate equals $\delta_{(pij)} = \gamma_{(i)} + \beta_0 + \nu_{(p)} + \varepsilon_{(pij)}$, but for a person p whose $Z_{(p)} = 1$, the drift rate becomes $\delta_{(pij)} = \gamma_{(i)} + (\beta_0 + \beta_1) + \nu_{(p)} + \varepsilon_{(pij)}$. Assume now that $Z_{(p)}$ is actually latent instead of observed, and that its distribution is Bernoulli with success probability π (i.e., $P\left(Z_{(p)} = 1\right) = \pi$). In this case we have defined a two-component mixture model for drift rate. The drift rate of a person p on trial j in condition i is then distributed as follows

$$\delta_{(pij)} \sim (1-\pi) N\left(\gamma_{(i)} + \beta_0, \eta_\varepsilon^2 + \sigma_\nu^2\right) + \pi N\left(\gamma_{(i)} + \beta_0 + \beta_1, \eta_\varepsilon^2 + \sigma_\nu^2\right)$$

The variances of both components are equal, but their means differ by a quantity β_1, which is to be estimated from the data.

A mixture model for the diffusion model parameters may be used to detect hidden groups or subpopulations in the data. For instance, one may hypothesize that in a certain experiment, different people use different strategies to respond to the stimuli. In such a case, a mixture model may be of help in finding out whether there are really two groups and how much they differ. Of course, the difference between the subpopulations need not be restricted to one parameter. The groups could, for example, differ in

drift rate and nondecision time as well.

Finally, latent predictors can also be continuous. In the model for drift rate as proposed in Equation 4.2, there is a single person random effect involved: $\nu_{(p)}$. This means that for all trials and in all conditions, the persons may be located on a continuous dimension and that these locations remain the same for all trials and conditions (i.e., that the person component is constant and independent of conditions or items). However, it could be the case that in a given experiment two (or more) dimensions of information processing are required and some conditions rely more on one dimension and other conditions more on the other dimension. Such a model for drift rate can be expressed as:

$$\delta_{(pij)} = \gamma_{(i)} + \lambda_{(i)1}\nu_{(p)1} + \lambda_{(i)2}\nu_{(p)2} + \varepsilon_{(pij)},$$

where $\lambda_{(i)1}$ and $\lambda_{(i)2}$ are the loadings of the underlying dimensions in condition i and $\nu_{(p)1}$ and $\nu_{(p)2}$ are the positions of person p on the two dimensions. Such a model can be called a *factor analysis diffusion model*. However, we will not discuss such models further because they rapidly become very complex and estimating their parameters is computationally very intensive (at least, using current standard approaches for computational Bayesian inference).

4.4 Statistical inference for HDMs

In the practical application of the HDM framework, statistical inference will be performed using Bayesian statistical methods (see e.g., Gelman, Carlin, Stern, & Rubin, 2004). In this section, we provide some background of Bayesian methods that is required for interpreting the results of our analyses, as well as for the application of our software. We believe this

background to be important because, while the philosophy behind Bayesian statistics is fairly straightforward and easy to explain, the computational techniques involved are not. As a result, output from a Bayesian computational analysis must be checked with care before it is interpreted. First, however, we will detail why we have chosen this particular framework for implementing the HDM.

4.4.1 Motivation for the Bayesian statistical framework

Several reasons motivate our choice to use Bayesian inference. The Bayesian framework has many inherent advantages, such as the principled, consistent, and intuitive treatment of uncertainty concerning the parameters of the model. However, there are several advantages specific to the topic of the present paper. Bayesian methods are most suited for flexible implementation of hierarchical models in particular (see also Gelman & Hill, 2007).

The diffusion model in itself, without any hierarchical extension, is already a computationally difficult model (see e.g., Tuerlinckx, 2004; Navarro & Fuss, 2008). These difficulties are exacerbated by even small increases in the hierarchical structure of the model (Ratcliff & Tuerlinckx, 2002; Tuerlinckx, 2004; Vandekerckhove & Tuerlinckx, 2007). Models with more extensive hierarchical structures (as discussed here) are often more interesting, but rapidly become computationally intractable in the classical statistical framework where parameters have to be estimated using maximum likelihood methods. Take for example a crossed random effects model for drift rate (a random effect of person and of item), and assume for simplicity that the other parameters are kept constant across persons and items. When applying such a relatively simple model to a data set of P persons and I items, we are confronted with a likelihood function that contains

an integral of dimension $P + I$. If P and I are both around 100, this means that a 200-dimensional intractable integral has to be approximated by standard numerical integration techniques, which is computationally prohibitive.

4.4.2 Computation in the Bayesian framework

Bayesian methods depend on the computation of the posterior distribution of model parameters. That is, the probability distribution of the parameters, given the data. The posterior distribution can be obtained through Bayes' rule:

$$p(\theta|y) = \frac{p(y|\theta)p(\theta)}{p(y)}$$

where y and θ is a generic notation to refer to all the data and all the parameters, respectively, $p(y|\theta)$ is the likelihood, $p(\theta)$ is the prior distribution on the parameters and $p(y)$ is the marginal probability of the data.

Typically, however, these distributions are mathematically complex and it is nontrivial to compute summary statistics of the (often high-dimensional) posterior $p(\theta|y)$. In many cases it is comparatively much easier to generate randomly drawn samples from this posterior distribution. Summary statistics of a sufficiently large sample can then be used to accurately represent the posterior distribution. A class of general methods for sampling from a complex distribution are *Markov chain Monte Carlo* (MCMC; Robert & Casella, 2003) techniques and in particular the Metropolis-Hastings algorithm, in which samples are drawn from some initial distribution that is not the posterior, and this initial distribution is changed as successive samples are drawn in such a way that after a number of iterations the samples draws are truly samples from the posterior.

Importantly, this means that the first samples are not representative of the posterior distribution. That is, the convergence of the MCMC al-

gorithm to a stationary distribution is not immediate, and may in fact take some time. It is therefore necessary to check whether convergence has occurred before the output of the algorithm may be considered to be draws from the posterior distribution. Our preferred method of assessing convergence is to run a small number of different chains (say, six). After discarding a certain number of iterations, called the *burn-in*, we test whether the remaining draws from the different chains are in fact draws from the same distribution. For this, we use the $\hat{R}$ criterion (Robert & Casella, 2003), a statistic that is similar to the F statistic in analysis of variance. $\hat{R}$ is large if the between-chain variance is larger than within-chain variance, and it approaches unity when the different chains have converged to the same distribution. $\hat{R}$ values lower than 1.1 are typically considered satisfactory. In our applications, we always ran six independent chains and obtained $\hat{R}$ values under 1.05 for all parameters. In the reports of our practical examples, we will always make brief notes regarding the technical results (because we believe this is good practice), even though they indicated good convergence and stable estimates in each of the analyses.

4.4.3 Priors

A final technical matter is the choice of prior distributions which are required for the computation of the posterior distributions. Choice of priors is a somewhat controversial topic in statistics, mainly because they seem to require a somewhat subjective judgment on the part of the researcher. It is also a matter of debate whether it is possible (and desirable) to construct prior distributions that are entirely uninformative. For our purposes, we have struck a simple compromise: we selected theoretically informed limits on the possible range of each parameter, and chose a prior distribution that reflected equiprobability of all values in that range (i.e., a uniform or "flat"

prior). Researchers who have more prior information regarding the distribution of parameters can implement this in our software easily. In practice, however, the influence of even quite informative prior distributions on the posterior distributions is small compared to the weight of the likelihood of the data.

4.4.4 Graphical models

The advanced sampling algorithms described above are implemented in the freely available statistical software package WinBUGS (Lunn, Thomas, Best, & Spiegelhalter, 2000). WinBUGS can also be used very easily to apply a HDM (see Appendix 4.A for more detail). In order to use WinBUGS, however, it is necessary to translate the hierarchical model into a directed acyclical graph or *graphical model*. Graphical models (see Griffiths et al., 2008 and Lee, 2008 for accessible introductions) are a convenient formalism for describing the probabilistic relationships between parameters and data. In a graphical model, variables of interest are represented by nodes in a directed graph, with children depending on their parents. Circular nodes represent continuous variables, square nodes discrete variables, shaded nodes observed variables, and unshaded nodes unobserved variables. In addition, plates enclose parts of a graph to denote independent replications. An example of a simple graphical model is given in Figure 4.2.

As soon as a model has been translated into a graphical model, it can be implemented in WinBUGS and it becomes straightforward to perform full Bayesian inference computationally, using standard MCMC methods to sample from the posterior distribution.

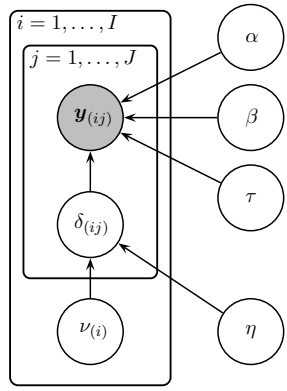


Figure 4.2: *An example graphical model. The shaded node* $\boldsymbol{y}_{(pij)}$ *indicates the (bivariate) data. Nodes* α*,* β*,* τ*, and* $\delta_{(ij)}$ *are parameters of the distribution of* $\boldsymbol{y}_{(pij)}$*. In turn,* $\nu_{(i)}$ *and* η *are parameters of the distribution of* $\delta_{(ij)}$*.*

4.4.5 Evaluating model performance in the Bayesian framework

After the posterior distributions of all parameters have been found, two aspects of model performance can ascertained. To determine *relative model fit* across a series of models, the DIC measure (Gelman et al., 2004) can be computed. This statistic, the *deviance information criterion*, can be considered as a Bayesian alternative to the *Akaike information criterion* (AIC). Like the AIC, the DIC also expresses a balance between the model fit and its complexity. Lower DICs are better.

In order to determine *absolute model fit*, however, we might apply *posterior predictive checks* (PPC; Gelman et al., 2004). The simplest type of PPC involves defining an interesting test statistic $G(\cdot)$ on the data, and computing those statistics for the observed data (i.e., $G(y^{obs})$). Then the same statistic $G(\cdot)$ can be computed on a large number (say, $1,000$) of data sets that are generated from the model, leading to a set $G(y^{rep\ 1}), \ldots, G(y^{rep\ 1000})$. Finally, the position of $G(y^{obs})$ in the distribution of $G(y^{rep\ 1}), \ldots, G(y^{rep\ 1000})$ then indicates the viability of the model

with regard to the data. A more complex type of PPC can be defined as well (such that the test statistic is not pivotal, but also depends on the parameters). However, this type of test statistic requires a (very time-consuming) re-estimation of the model parameters for each replicated data set, which is why we do not apply it.

4.5 Application examples

To illustrate the usefulness of the HDM framework, we now apply it to three data sets with widely different designs, but all three of which seem usefully dealt with using HDMs. In the first application, we will apply a series of HDMs to a "benchmark" data set concerning contrast perception, and apply mainly regression-type analyses, as well as trial-to-trial variability in drift rate, initial bias, and nondecision time. We will also include a very simple hierarchical structure, namely the addition of random variability over conditions. Note that in this example, we will not consider individual differences or complex hierarchical structures. We will use the first application mainly to demonstrate the basic features of the diffusion model, the Bayesian modeling approach, the principles of Bayesian model selection, and the relative ease with which these otherwise involved analyses can be performed.

In the second application, which concerns lexical decision data, we will apply a random effect of lexical items and add a learning curve to the model. We will show that an HDM can easily capture a learning trend in addition to allowing inter-item variability (part 1 of the application). Continuing with the same data set, we will also use information from an item covariate to explain the variability in items (part 2). However, it will turn out that the covariate we selected does not explain the differences well.

In the third and final application, we again use a data set from a psychophysical experiment. This data set is different because it has more participants ($n = 9$), and we will construct an HDM that permits the simultaneous analysis of data from different individuals. Thanks in part to the Bayesian framework, we are able to define a statistic that directly quantifies the effect under consideration, and estimate the distribution of its size in the population.

In each of the examples, we will make a large number of assumptions regarding the structure in the data. We will, sometimes somewhat arbitrarily, select whether certain parameters are allowed to change between experimental units, whether effects are fixed or random, and which parametric forms are taken by population distributions or regression functions. These assumptions will often be debatable, but the central point to be made is that a wide variety of assumptions can be made explicit in the HDM framework with relative ease. For the purposes of illustrating this, it is not of crucial importance exactly which assumptions are made.

4.5.1 Example 1: Fixed effects and nonlinear regression

4.5.1.1 Introduction

The first application example involves a data set in a contrast discrimination task that has become something of a benchmark for RT model fitting (Ratcliff & Rouder, 1998, data used with permission). An important reason for this is that these data clearly show the standard RT phenomena for which any model of choice RT should be able to account (Brown & Heathcote, in press). In the experiment, three participants saw ten blocks of trials (after two practice blocks). Each trial consisted of a grid with 75% grey pixels and the remaining 25% either black or white. There were 33 different proportions of black versus white pixels (evenly spaced, so that

the middle level is 50% black, 50% white), and the task was to determine whether this proportion was a draw from a 'bright' or a from 'dark' distribution. Additionally, in half of the blocks, the participants were asked to respond as accurately as possible (accuracy condition; AC), and in the other half to be as fast as possible (speed condition; SC).

The research goal in this study was to study the relationship between stimulus brightness and drift rate. A link was clearly confirmed, and it was found that this link was nonlinear in nature. Here, we will go two steps further. First, we will formalize the nonlinear relation using a cumulative Weibull link function, which is a nonlinear function that is common in the vision literature (see, e.g., Wichmann & Hill, 2001). Then we will investigate the effect of the instruction (AC vs SC) on the relation between stimulus brightness and drift rate—as it could be hypothesized that a task instruction affects the amount of 'noise' in the decision making system, which would show itself in a different shape of the link function. We will focus on a single participant's data.

4.5.1.2 Models

As an introductory example, we apply a simple HDM to these data. However, the features added to the Wiener diffusion are not limited to the trial-to-trial variance used by Ratcliff and Rouder: We also implement a nonlinear regression and allow a difference between the instruction conditions. Specifically, let $C_{(s)} = s/32$ ($s = 0, \ldots, 32$) be a measure of intensity (i.e., brightness) and i ($i = 1$ for AC; $i = 2$ for SC) the instruction condition. Considering only one participant (so that we can drop the index p), we have the following model for the observed response vector $\boldsymbol{Y}_{(sij)}$:

$$\boldsymbol{Y}_{(sij)} \sim W\left(\alpha_{(sij)}, \beta_{(sij)}, \tau_{(sij)}, \delta_{(sij)}\right).$$

We will assume that the parameter α was subject only to a fixed effect of instruction:

$$\alpha_{(sij)} = \alpha_{(i)},$$

whereas β, τ, and δ are subject to random effects of trial:

$$\beta_{(sij)} \sim U\left(\pi^{lo}_{(i)}, \pi^{hi}_{(i)}\right);$$

$$\tau_{(sij)} \sim N\left(\theta, \chi^2\right);$$

$$\delta_{(sij)} \sim N\left(\nu_{(si)}, \eta^2\right).$$

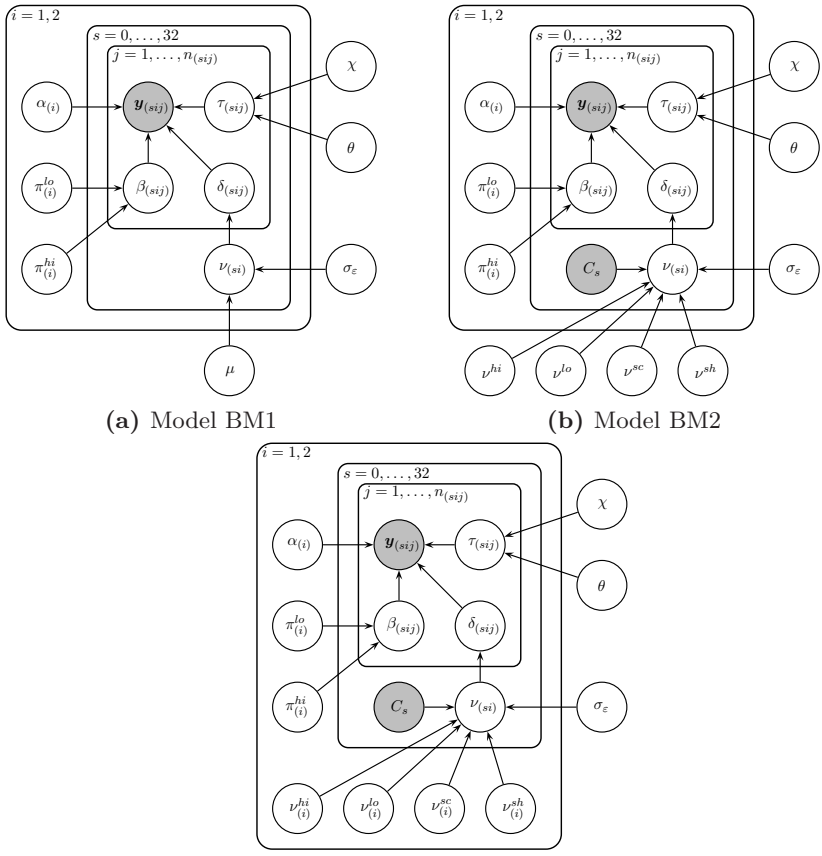
The mean of the trial-to-trial distribution of δ is additionally subject to a random condition effect:

$$\nu_{(si)} \sim N\left(\mu, \sigma^2_{\varepsilon}\right),$$

introducing a key ability of the HDM. Here it becomes most clear why these models are called hierarchical, because 'layers' of randomness are added incrementally (in this case, one at the condition level and one at the trial level). The model with this set of assumptions will be called "Model BM1". Note that model BM1, while acknowledging the possibility of difference between the 66 drift rates, contains no information to quantify the differences between the conditions.

However, we can define multiple competing models. Ratcliff and Rouder's model did not restrict the across-condition drift rate distributions.[5] In contrast, we now define a second model in which we formalize the

[5]Ratcliff and Rouder do mention that they could (in principle) further simplify the model by implementing a regression of mean drift rate as a linear function of the probability that the stimulus was a draw from the 'bright' distribution, i.e., $\nu_{(ps)} = \zeta_{(p)0} + \zeta_{(p)1} P_{(s)}$, with $P_{(s)} = N\left(s|\kappa_1, \phi\right) / \left[N\left(s|\kappa_1, \phi\right) + N\left(s|\kappa_2, \phi\right)\right]$ and $\kappa_1 = 5/8$, $\kappa_2 = 3/8$, and $\phi = 3/16$. However, they did not actually apply this regression.



(a) Model BM1 **(b)** Model BM2

(c) Model BM3

Figure 4.3: *A graphical model representation of each of the models for the first application.*

connection between stimulus intensity and drift rate with a Weibull link. Formally, redefine

$$\nu_{(si)} = \nu^{lo} + \left(\nu^{hi} - \nu^{lo}\right)\left\{1 - \exp\left[-\left(C_{(s)}/\nu^{sc}\right)^{\nu^{sh}}\right]\right\} + \varepsilon_{(si)},$$

with the error term $\varepsilon_{(si)} \sim N\left(0, \sigma_\varepsilon^2\right)$. Note that while σ_ε^2 in BM1 indicated the across-condition variability in $\nu_{(si)}$, here it refers to the *residual*

variability after accounting for the effect of the brightness condition. Importantly, the ability to quantify residual variability after controlling for the effect of the brightness condition allows us to investigate the magnitude of inter-stimulus variability that is not due to an experimental manipulation (but rather due to other manipulations or due to random, uncontrolled differences between stimuli). The second model, now completely specified, will be called "Model BM2".

However, we had originally set out to investigate the effect of the experimental instruction on the drift rates. Let us construct a third model in which we allow a difference in the drift rate distributions as a function of the instruction condition, using the link function

$$\nu_{(si)} = \nu^{lo}_{(i)} + \left(\nu^{hi}_{(i)} - \nu^{lo}_{(i)}\right)\left\{1 - \exp\left[-\left(C_{(s)}/\nu^{sc}_{(i)}\right)^{\nu^{sh}_{(i)}}\right]\right\} + \varepsilon_{(si)},$$

Note that we have added subscripts i to the Weibull's parameters, to indicate their dependence on the instruction condition. This will be Model BM3. The three models are displayed as graphical models in Figure 4.3.

4.5.1.3 Results—technical

We drew samples from the posterior distribution using the software described in Appendix 4.A. We ran six independent chains of $10,000$ iterations, and $5,000$ of these iterations were discarded as burn-in. We computed the $\hat{R}$ statistic for each parameter and confirmed that they were all lower than 1.05, indicating good convergence of the chains. Visual inspection of the chains indicates no issues (see Figure 4.4 for an example).

4.5.1.4 Results—substantive

The means and standard deviations of the marginal posteriors for some of the parameters in each model are given in Table 4.1. Several results are im-

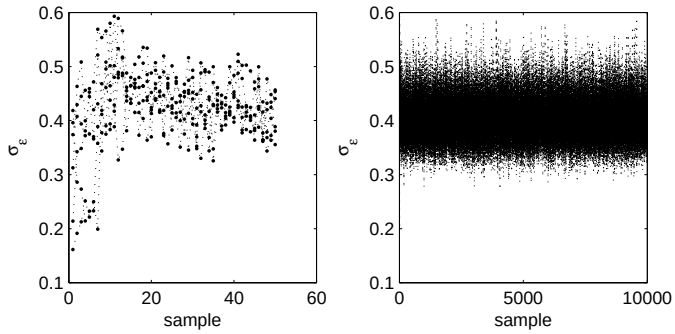


Figure 4.4: *An example of a sample chain. The parameter depicted is σ_ε, the condition-level variability, in BM1. While the chain looks well-mixed in the right panel, the left panel shows that the first few iterations are still distinctly overdispersed. However, since this overdispersion quickly dissipates and is comfortably within the burn-in zone, it does not affect the quality of the sampling.*

mediately obvious. Firstly, the parameters α_{AC} and α_{SC} are very different: the boundary separation in the SC is much lower than in the AC, in all models. This is consistent with the interpretation of that parameter. Secondly, the posterior standard deviations are generally small compared to the posterior means (EAPs), indicating narrow distributions and therefore reliable estimates. Finally, the parameter σ_ε that indicates the amount of unexplained varibility in drift rates strongly differs between models—apparently, the added covariates do explain a fair amount of variance.

We can use the difference in unexplained stimulus variance as a quality measure of the Weibull regression, using a statistic akin to the familiar statistic $R^2 = 1 - \left(\frac{\sigma_{res}}{\sigma_{total}}\right)^2$, where in this case σ_{total} is σ_ε in BM1, and σ_{res} is σ_ε in the model with which we want to compare. Given a series of samples from each of these parameters, we can compute a posterior mean for the proportion of variance that is explained by the addition of the nonlinear regressions.[6] In BM2, the proportion of variance explained

[6]Since we are not dealing with a linear model, and are in fact comparing across

Table 4.1: *Some parameter estimates for the first application. Posterior standard deviations have been multiplied by 100.*

	EAP			(100x) STD		
parameter	BM1	BM2	BM3	BM1	BM2	BM3
α_{AC}	0.2192	0.2314	0.2199	0.4344	0.5465	0.4507
α_{SC}	0.0501	0.0511	0.0502	0.0984	0.0956	0.0990
θ	0.2791	0.2769	0.2789	0.1758	0.1726	0.1779
χ	0.0412	0.0404	0.0410	0.0934	0.0917	0.0937
η	0.1261	0.1425	0.1273	0.7848	0.8903	0.7960
π^{lo}_{AC}	0.3522	0.3431	0.3515	0.9947	0.9402	1.0429
π^{hi}_{AC}	0.5755	0.5832	0.5757	0.8259	0.6975	0.8218
π^{lo}_{SC}	0.4498	0.4492	0.4495	0.9888	1.0170	0.9984
π^{hi}_{SC}	0.4779	0.4771	0.4776	0.9670	0.9835	0.9865
σ_ε	0.4008	0.0732	0.0064	3.8249	1.1022	0.4323

is 96.50%, while in BM3 it is as high as 99.96%.

In Table 4.2, the parameters of the Weibull regression are shown for BM2 and BM3. It is clear from the posterior means and standard deviations that the Weibull regression function is quite different between the two instruction conditions. In particular, the upper and lower asymptotes are more extreme in the SC, and the function is somewhat steeper in that condition as well. In fact, according to the analysis, $P\left(\nu^{sh}_{\text{SC}} > \nu^{sh}_{\text{AC}}\right) \approx .9590$.

To compare the performance of the three models, we computed DIC values for each model and found that BM3 performed best (DIC was -13373.40, -12087.60, and -9642.63, for BM1, BM2, and BM3, respectively).

models with strongly different assumptions, the R^2 statistic used here is not exactly the same as the familiar statistic. However, for the purpose of comparing model fits, we believe it is a succinct summary measure.

Table 4.2: *Parameter estimates of the Weibull regression in the first application. Posterior standard deviations have been multiplied by 100. Note that BM2 does not allow for differences between the accuracy condition (AC) and the speed condition (SC)*

	EAP		(100x) STD	
parameter	BM2	BM3	BM2	BM3
ν^{hi}_{AC}	0.4132	0.3292	2.2774	1.4160
ν^{hi}_{SC}	0.4132	0.5110	2.2774	2.5016
ν^{lo}_{AC}	-0.4296	-0.3516	2.4513	1.4473
ν^{lo}_{SC}	-0.4296	-0.5654	2.4513	2.7277
ν^{sc}_{AC}	0.5258	0.5259	1.0179	0.5080
ν^{sc}_{SC}	0.5258	0.5214	1.0179	0.6037
ν^{sh}_{AC}	5.4092	4.4127	70.6052	24.1439
ν^{sh}_{SC}	5.4092	5.2268	70.6052	42.4271

4.5.1.5 Conclusion

While the model we have applied to these data is quite different from the one used by Ratcliff and Rouder (1998), our conclusions generally echo theirs, with one significant difference: We find an effect of instruction on drift rate. The Weibull link functions are manifestly different between the instruction conditions—evidently the rate of information accumulation (i.e., the amount of noise in the system) is not independent of the participants' motivations. While we do not describe them here, the results were analogous for the other two participants.

In addition to the relative ease with which it was applied (only 30 or so lines of highly redundant WinBUGS code; see Appendix 4.B), the above model contains two properties that are fundamentally novel in the domain. Trial-to-trial variance and constraints on parameters have already been applied (e.g., by Vandekerckhove & Tuerlinckx, 2007), but the application of Bayesian inference and in particular the addition of random effects on the condition (stimulus) level are new. Random effects are an important

modeling construct that has not previously been considered in this context. In the next example, we will focus more closely on the addition of random effects.

4.5.2 Example 2, part 1: Nonlinear regression and random item-domain effects

4.5.2.1 Introduction

The second example application involves data collected within a lexical decision task paradigm, whereby participants are shown a stimulus and have to decide whether it is a word or a non-word (Dutilh, Vandekerckhove, Tuerlinckx, & Wagenmakers, 2008, data used with permission). Each of the four participants was presented with 200 words and 200 word-like (i.e., pronounceable) nonwords in each of 25 blocks, spread over 5 days, for a total of $10,000$ trials per person. Two participants (S1 and S2) were given a speed instruction and two were given an accuracy instruction (A1 and A2). We will again focus on one participant: A2.

Importantly, each stimulus was repeated 25 times for each participant (five times per day). With the repeated measurements in this example, we can try to separate variability due to the item's characteristics from trial-to-trial variability. In order to do so, we will construct an HDM with random item effects.

The original goal of the data collection had been to investigate the effect of practice on task performance with a diffusion model in order to "locate" the practice effect. Accordingly, the modeling should allow for different process parameters on different days.

4.5.2.2 Models

We construct HDMs with random stimulus effects and flexible trial-to-trial variability. This will increase the conceptual complexity of the model somewhat. We will use the following subscripts: d ($d = 1, \ldots, 5$) for days, w ($w = 1$ for nonwords, $w = 2$ for words) for stimulus category, s ($s = 1, \ldots, n_{(w)}$; with $n_{(w=1)} = n_{(w=2)} = 200$) for different stimuli, and j ($j = 1, \ldots, 25$) for repeated presentations of identical stimuli (i.e., trials). Note that stimuli s are nested within categories w. For the present analysis, we have elected to work with a temporal resolution of one day. However, nothing stops us in principle from using smaller temporal resolutions such as one block, or even smaller groupings of trials.

The random item effects are expressed in the drift rate. We will express drift rate as follows:

$$\delta_{(wsjd)} \sim N\left(\nu_{(wsd)}, \eta^2\right),$$

$$\nu_{(wsd)} = \gamma_{(ws)} + \lambda_{(wj)},$$

$$\gamma_{(ws)} \sim N\left(\mu_{\gamma(w)}, \sigma^2_{\gamma(w)}\right),$$

and

$$\lambda_{(wd)} \sim N\left(\mu_{\lambda(w)}, \sigma^2_{\lambda(w)}\right).$$

where the parameters $\lambda_{(wd)}$ and $\gamma_{(ws)}$ express a random effect of day and of item, respectively. Note that we allow both the day and item effect distributions to differ between words and nonwords. As an identifiability constraint, $\mu_{\gamma(w)} = 0$ for all w.[7] The α and β parameters are subject only

[7] We are required to restrict the value of $\mu_{\gamma(w)}$, because it is in a trade-off relationship with the values of the fixed day effect. That is, we could add any number to $\mu_{\gamma(w)}$ and subtract the same number from $\mu_{\lambda(w)}$ and obtain the same model predictions.

to random effects of day:

$$\alpha_{(d)} \sim N\left(\mu_\alpha, \sigma_\alpha^2\right)$$

and

$$\beta_{(d)} \sim N\left(\mu_\beta, \sigma_\beta^2\right).$$

The τ parameters are subject to trial-to-trial variability:

$$\tau_{(wsjd)} \sim N\left(\theta_{(d)}, \chi^2\right),$$

with the mean nondecision time $\theta_{(d)}$ again depending only on a random effect of day:

$$\theta_{(d)} \sim N\left(\mu_\theta, \sigma_\theta^2\right).$$

Trial-to-trial variability in nondecision time is considered to be constant over days. We will call this the *descriptive learning model* (DLM).

The DLM does not address the effect of practice. In order to parameterize this effect, we can apply an *exponential learning curve* as a constraint onto the process parameters. Formally, in the case of the boundary separation parameter,

$$\alpha_{(d)} = \psi_0 + e^{\psi_1 + \psi_2 d} + \varepsilon_{(d)}$$

with $\varepsilon \sim N\left(0, \sigma_\alpha^2\right)$. Equivalently, we could say that the "noiseless" model prediction of $\alpha_{(d)}$ is $\hat{\alpha}_{(d)} = \psi_0 + e^{\psi_1 + \psi_2 d}$, so that $\alpha_{(d)} \sim N\left(\hat{\alpha}_{(d)}, \sigma_\alpha^2\right)$. The story is slightly more complex for the nondecision time τ, where trial-to-trial variance (χ^2) is combined with the exponential learning curve. On the trial level, $\tau_{(wsjd)} \sim N\left(\theta_{(d)}, \chi^2\right)$, whereas on the day level, $\theta_{(d)} \sim N\left(\zeta_0 + e^{\zeta_1 + \zeta_2 d}, \sigma_\theta^2\right)$. The nondecision time distribution ($\theta_{(d)}$) is now expressed as a parametric function of day. Note that this formulation allows us to explicitly compare the parameters as predicted by the learning

curve ($\hat{\theta}_{(d)} = \zeta_0 + e^{\zeta_1 + \zeta_2 d}$) with the parameter that is directly involved in the process model ($\theta_{(d)}$). The amount of discrepancy between these parameters is quantified by the unexplained variance (σ^2_θ). Note also that since the exponential curve has three parameters, and there are only five days, the nonlinear regression will not reduce the complexity of the data by much, and a good fit is likely if the parameters follow the expected qualitative pattern.

Finally, we assume a linear effect of practice on the drift rate component that is day-dependent. Formally, $\lambda_{(wd)} = \varpi_{(w)0} + \varpi_{(w)1} d + \varepsilon_{(wd)}$, with $\varepsilon_{(wd)} \sim N\left(0, \sigma_{\lambda(w)}\right)$. Again, this effect is allowed to differ between words and nonwords.

This model, where trial-to-trial variability is combined with learning functions, will be called the *learning explanatory model* (LEM).

4.5.2.3 Results—technical

We again drew 10,000 samples in each of six chains. We discarded the first 5,000 samples as burn-in and computed the $\hat{R}$ statistic for each parameter. The statistic was always lower than 1.05. Again, no issues with convergence were noticeable in a visual inspection of the sample chains.

4.5.2.4 Results—substantive

Using the DIC criterion, the LEM (DIC: −19426.60) performs much better than the DLM (DIC: −18795.60). This indicates that the decrease in the number of parameters (due to the move from the descriptive to the explanatory model) results in a relatively small change in the lack of fit. Figure 4.5 shows estimates for nondecision time and boundary separation from the two models. It is clear that the nonlinear regression of the LEM captures the trend well. Figure 4.6 shows the posterior distributions for

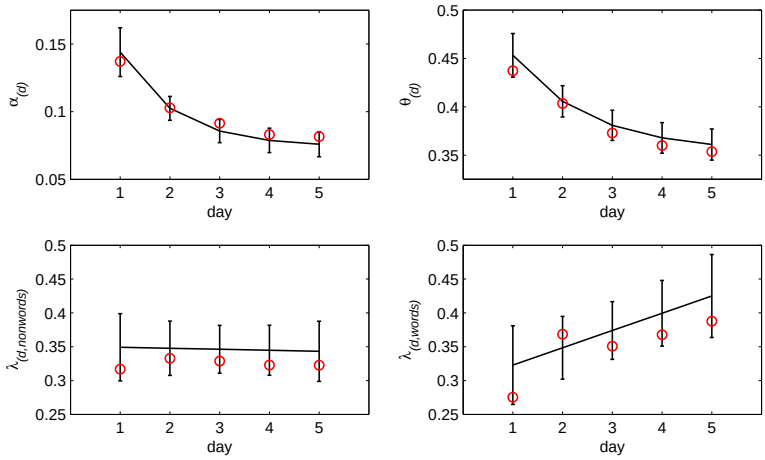


Figure 4.5: *Evolution of diffusion model parameters over days. There is a clearly nonlinear change in nondecision time and boundary separation, and the fit of the LEM (full black line, depicting $\hat{\alpha}_{(d)}$, $\hat{\theta}_{(d)}$, $\hat{\lambda}_{(d,w=1)}$, and $\hat{\lambda}_{(d,w=2)}$) to the freely estimated parameters of the DLM (circles) is very good. The drift rate for nonword recognition remains stable over days, but at word recognition, participants seem to become more adept. The error bars go one posterior standard deviation in both directions.*

ζ_2 and ψ_2. The distributions allow us to derive $P(\zeta_2 > 0)$ and $P(\psi_2 > 0)$, both of which turn out to be numerically zero.

In the case of the parameters that capture the day component of drift rate, the case is not so clear. Figure 4.7 shows the posterior distributions for the $\varpi_{1(w=1)}$ and $\varpi_{1(w=2)}$ parameters. $P\left(\varpi_{1(w=1)} > 0\right)$ turns out to be .5404—in other words, the hypothesis that $\varpi_{1(w=1)} < 0$ (i.e., that there is no positive learning effect in the case of nonword stimuli) cannot be discarded. In contrast, $P\left(\varpi_{1(w=2)} > 0\right) \approx .0835$, providing somewhat more evidence against the hypothesis that $\varpi_{1(w=2)} < 0$ and that learning is occurring in the case of word stimuli.

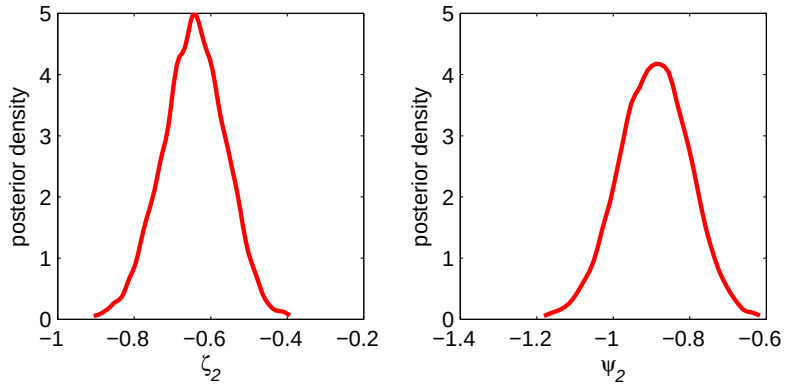


Figure 4.6: *Posterior distributions for the change rate parameters for nondecision time (ζ_2) and boundary separation (ψ_2). Since there is no posterior mass above zero in either case, we can conclude that there is, indeed, an effect of learning on these parameters.*

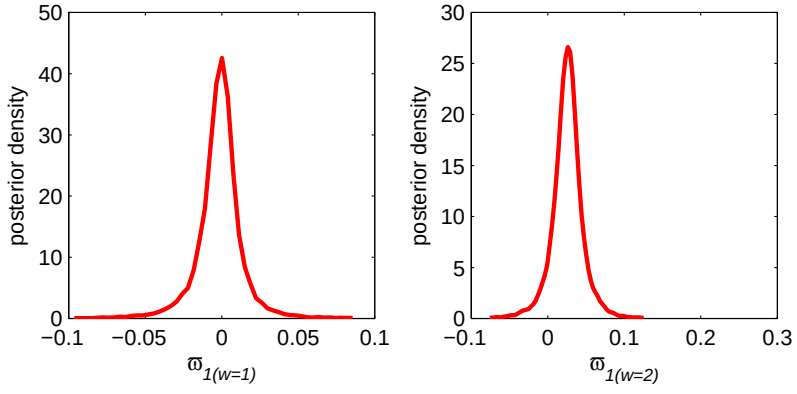


Figure 4.7: *Posterior distributions for the change rate parameters for the two drift rates (nonwords: $w = 1$; words: $w = 2$). In both cases, the distributions clearly straddle the value* 0.

4.5.2.5 Conclusion

The LEM performs relatively well for this data set. Several interesting conclusions can be drawn. Firstly, an exponential learning model captures the changes in boundary separation and nondecision time well. Secondly, there is no evidence for a learning effect on nonwords. Finally, word stimuli bring about more variability in performance than do nonword stimuli. More extensive conclusions can be found in Dutilh et al. (2008).

4.5.3 Example 2, part 2: Extra explanatory covariates

4.5.3.1 Introduction

While the previous models formalize many of the interesting aspects of the theory of practice, and makes use of an external covariate (days) to explain variance in diffusion model parameters, it is purely descriptive on the item side of the data. However, since half of the items used are real Dutch words, several item covariates are available. It is conceivable that variability in item characteristics explains some of the variability in the diffusion model parameters. Particularly, part of the unexplained variance in drift rate (i.e., the magnitude of the random effect of item on drift rate) might be related to an item property. We have obtained one such covariate: the words' usage frequencies. We now construct a new model that extends the LEM above into a *two-way explanatory model* (TEM; because we use covariates to explain both the learning rate and variability on the item side).

4.5.3.2 Model

The extension is mathematically minor. We simply add to the LEM the following assumption about the mean of the distribution that governs the

random effect of item:

$$\mu_{\gamma(w=2,s)} = \rho F_{(s)},$$

where $F_{(s)}$ is the (normalized) frequency of stimulus s. Of course, only words ($w = 2$) have an associated frequency (i.e., $\mu_{\gamma(w=1,s)} = 0$ for all s).

4.5.3.3 Results—technical

As in the previous models, no problems with convergence were apparent. The $\hat{R}$ value for the new parameter ρ was close to 1, and its sample chains showed rapid mixing.

4.5.3.4 Results—substantive

The only new parameter of interest is ρ, whose EAP turns out to be very close to zero. Moreover, since the covariate F was normalized, we can derive that a value of ρ that is less than 0.01 in absolute value is very small and essentially meaningless (because F will typically be less than 3 in absolute value, and the regression contribution to the drift rate $\mu_{\gamma(w=2,s)} = \rho F_{(s)}$ will therefore be less than 0.03 for all items). From the posterior distribution of ρ (shown in Figure 4.8), we can compute that $P(-0.01 < \rho < 0.01) \approx .554$. Hence, the new parameter does not seem to contribute to model performance. It is unsurprising, therefore, that the DIC of the TEM (-19359.00) is worse than that of the LEM.

4.5.3.5 Conclusion

Extending the LEM with a predictor on the item side of the model was, with the predictor we used, not a success. The TEM does not perform better than its simpler version, the LEM. However, the TEM does provide a worked example of how a more complicated, and plausible, hypothesis about the data can be tested within the HDM framework.

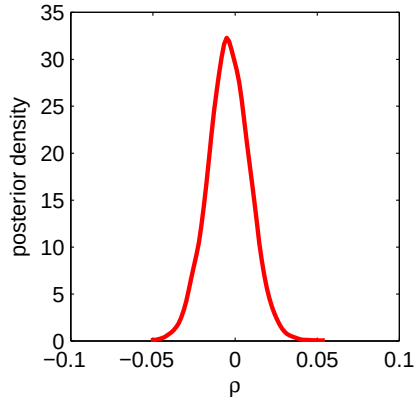


Figure 4.8: *The posterior distribution of the regression slope parameter ρ for model TEM. The parameter does not appear to differ from zero.*

4.5.4 Example 3: ANOVA and random person-domain effects

4.5.4.1 Introduction

In the previous applications, we have focused on single participants (mainly because the data sets contained only three and four participants, respectively). However, one of the most significant advantages of the hierarchical setting is that it allows the simultaneous analysis of many participants' choice response time data. For example, diffusion parameters could be kept constant across items for each participant, but individual participants' parameters would be considered random draws from a population distribution. This would be a most typical hierarchical model; van der Linden (in press) would call this a *population model*. Analyzing data from different participants simultaneously results in greater stability for the statistical inferences. In particular, by allowing "cross-talk" between data from different participants, it becomes possible to fit the model even with relatively few data points per participant.

However, because it remains unreasonable to assume that all parame-

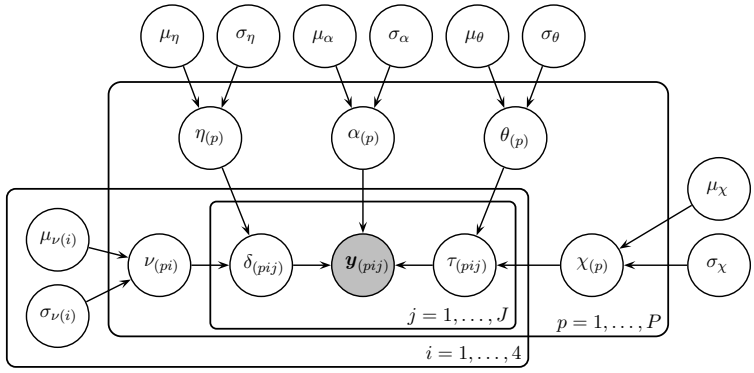


Figure 4.9: *The graphical model representation of the PHM. See text for details.*

ters stay exactly constant across trials, we can combine mixing over trials with mixing over persons. This would yield a *multilevel random effects* design wherein the parameters of individual participants' mixing distributions are themselves draws from a population-level distribution. A graphical representation of this multilevel diffusion model is given in Figure 4.9.

The data set to which we will apply this model is taken from a change detection study (Vandekerckhove, Panis, & Wagemans, 2007, data used with permission). For a detailed description of the research questions, the reader is referred to Vandekerckhove et al. (2007). For the purposes of our demonstration, it suffices to know that the difficulty of a visual detection task was manipulated in a 2-by-2 factorial design, and that there were nine participants. The independent variables of interest will be called Q, for *quality*, and T, for *type*. Because the manipulations are all intended to affect higher-order properties of the stimulus, we expect changes in drift rate, but not in any other variable. The main research question was whether there is an effect of T on detection performance, and whether this effect is independent of Q. It is hence a simple ANOVA-type design, and we are interested in the main effect of T and the T-by-Q interaction. The

factorial design is given in the second and third column of Table 4.3.

4.5.4.2 Model

We will define only one model, which includes a hierarchical structure that provides cross-talk between different participants' data. The assumptions of this population-hierarchical model (PHM) are as follows.

First, the basic parameters follow mixing distributions (over trials) whose parameters depend on participant p and may depend on condition i (in the case of drift rate)

$$\tau_{(pij)} \sim N\left(\theta_{(p)}, \chi_{(p)}\right),$$

$$\delta_{(pij)} \sim N\left(\nu_{(pi)}, \eta_{(p)}\right).$$

Furthermore, we assume that $\beta_{(pij)} = .5$. In words, we will apply an unbiased diffusion process with trial-to-trial variability in nondecision time and in drift rate. The mean of the drift rate distribution depends both on the participant and on the condition.

Second, we treat all interindividual differences as random effects (since we know that participants were a random sample from a larger population):

$$\alpha_{(p)} \sim N\left(\mu_\alpha, \sigma_\alpha^2\right),$$

$$\theta_{(p)} \sim N\left(\mu_\theta, \sigma_\theta^2\right),$$

$$\nu_{(pi)} \sim N\left(\mu_{\nu(i)}, \sigma_{\nu(i)}^2\right).$$

Note that the fixed effect of condition i remains present in the dependence of $\mu_{\nu(i)}$ on i, but now it exists on the population level. It is not necessary to define the factorial structure of the conditions in the experiment at this stage: since the parameters in a linear model that quantify

main effects and interactions are simple linear combinations of the data (i.e., the mean in each condition), we can compute posterior distributions for each conditional mean first, and derive the posterior distributions of the ANOVA parameters later[8] (see the substantive results section).

Finally, although it is not the primary focus of the present analysis, the trial-to-trial variability parameters are also given population distributions: $\chi_{(p)} \sim N\left(\mu_\chi, \sigma_\chi^2\right)$ and $\eta_{(p)} \sim N\left(\mu_\eta, \sigma_\eta^2\right)$.

In all cases, the population distributions are truncated to a reasonable interval (for numerical stability; see Appendix 4.B for the intervals).

4.5.4.3 Results—technical

We ran six chains for $10,000$ iterations each. For no parameters did the $\hat{R}$ value exceed 1.05. Based on visual inspection, mixing was good. See Figure 4.10 for an example.

4.5.4.4 Results—substantive

We are interested in two different aspects of the results. For the experimenter, it is important to know whether a main effect of T and a T-by-Q interaction appear on the mean drift rates $\mu_{\nu(i)}$. From a general-interest perspective, we are additionally interested in the population-level variability of the different parameters.

Summary statistics of the obtained drift rate population distributions (per condition) are given in Table 4.3. It can be seen that the distributions differ strongly between conditions. In order to more precisely investigate our hypotheses, we will transform the drift rate distributions into ANOVA contrast parameters which exactly quantify the effects we are interested in.

[8] Parameters that are not directly estimated themselves but are obtained from transformations and/or combinations of other parameters are sometimes called *derived parameters* or *structural parameters* (Congdon, 2003; Jackman, 2000).

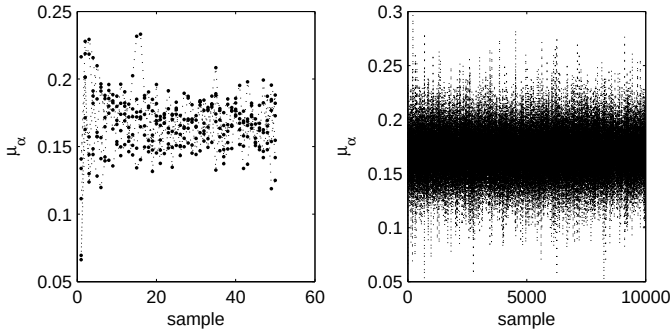


Figure 4.10: *An example sample chain from model PHM. The parameter shown is μ_α, the population mean of the boundary separations. In the left graph, the first* 50 *iterations are depicted, showing that the six chains rapidly converge to the same region. In the right graph, the entire chains are shown, showing that the six chains appear to be sampling from the same stationary distribution.*

Table 4.3: *Posterior distributions of the mean drift rate in the population, per condition. Where T is* 0*, the population distribution of drift rate has much mass around* 0*.*

i	Type (T)	Quality (Q)	$\mu_{\nu(i)}$	$\sigma_{\nu(i)}$
1	1	0	0.0870	0.0357
2	0	0	-0.0347	0.0555
3	1	1	0.2700	0.0558
4	0	1	0.0363	0.0448

First, the main effect of T is given by the contrast $\zeta_T = (\mu_{\nu(1)} + \mu_{\nu(3)}) - (\mu_{\nu(2)} + \mu_{\nu(4)})$, for which the posterior distribution is shown in Figure 4.11. It is clear from that figure that $P(\zeta_T < 0) \approx 0$. There is hence very strong evidence for a main effect of T, averaged over levels of Q. Similarly, in the second panel in Figure 4.11, we can confirm that there is a main effect of Q, since for $\zeta_Q = (\mu_{\nu(1)} + \mu_{\nu(2)}) - (\mu_{\nu(3)} + \mu_{\nu(4)})$, $P(\zeta_Q < 0) \approx .994$.

To investigate the interaction, we compute the interaction contrast $\zeta_I = (\mu_{\nu(1)} - \mu_{\nu(2)}) - (\mu_{\nu(3)} - \mu_{\nu(4)})$. As it turns out, $P(\zeta_I < 0) \approx .886$, providing only marginally convincing evidence for an interaction. A nega-

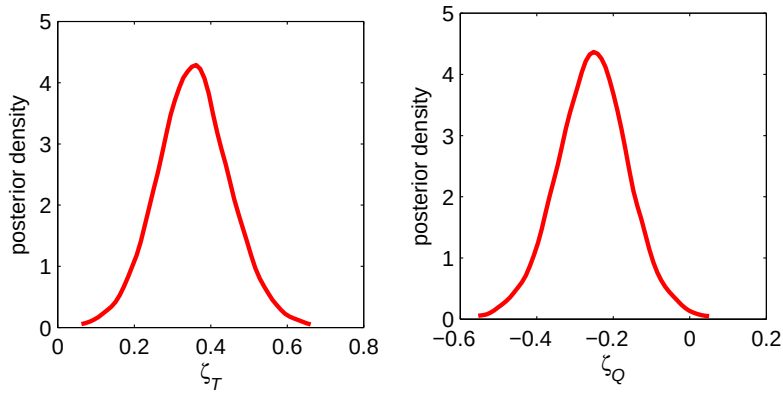


Figure 4.11: *Contrast parameters for the third application. See text for details.*

tive interaction in this context would mean that the joint effect of the T and Q variables is smaller than the sum of their individual main effects.

The population variability in the parameters is directly quantified by their variance parameters. The EAPs of the interperson variabilities of boundary separation, mean nondecision time, intertrial variability of nondecision time, and intertrial variability in drift rate are, respectively, $\sigma_\alpha = 0.0541$, $\sigma_\theta = 0.0663$, $\sigma_\chi = 0.0174$, and $\sigma_\eta = 0.0858$.

Given these estimated population distribution parameters and their remaining uncertainty (i.e., the posterior variance of these parameters), we can now depict the distribution of the model parameters in the population by computing posterior predictive distributions. Take, for illustration, the population distribution of α. Given a single sample $\mu_\alpha^{(s)}$ from the posterior distribution of μ_α, and a single sample $\sigma_\alpha^{(s)}$ from the posterior distribution of σ_α, we can generate a single sample $\alpha^{(s)}$. Repeating this procedure many times yields a vector of α values that are sampled from the population distribution. Thus, a sufficiently high number of samples obtained this way represents the expected population distribution of α. Figure 12 shows

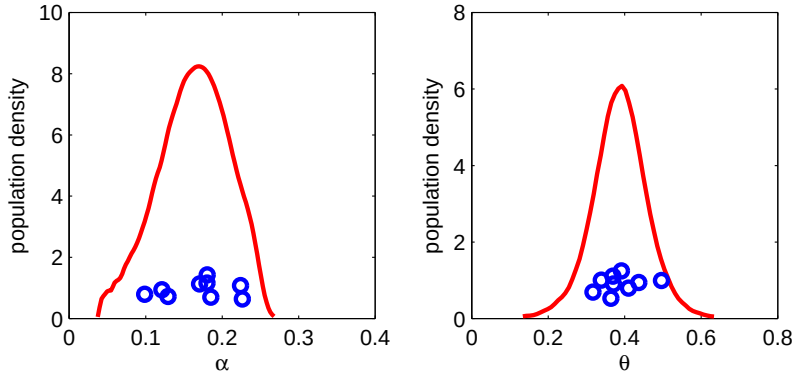


Figure 4.12: *Posterior predictive population distributions of two parameters of the PHM. The population distribution of θ is bell-shaped and narrow. The population distribution of α is wider.*

these predicted population distributions for the α and θ parameters. The parameter estimates for the nine participants in the experiment are shown as circles under the distribution curve. It can be seen from the figure that population variability in α is quite large, while it is comparatively small for θ. Also, while the θ parameters seem to follow a bell-shaped distribution, α parameters are more spread out, and even appear to occur in clusters.

4.5.4.5 Conclusion

In the final application, we applied a population-hierarchical model to choice response time data. We computed ANOVA-style contrasts for the two-by-two factorial design and found two main effects of the independent variables on the drift rate parameter. The population-hierarchical diffusion model is especially noteworthy because it combines information from different participants (and conditions) in a single model, allowing for cross-talk between the data, more reliable parameter estimates, and hypothesis tests at the population level.

4.6 Software implementation and technical details

We believe the hierarchical extension of the Wiener diffusion process has much potential for the field of cognitive science. However, applying this model in practice is difficult and may be prohibitively onerous for many researchers. For this reason, it is important also to publish computer software to aid in the application of the HDM.

Our software implementation (presented in Appendix 4.A) is not a full software package, but rather a plug-in to an existing software package, WinBUGS. The Appendix details how the software and plug-ins need to be installed, and also presents some examples of usage. Note that the software is dependent on Microsoft Windows, and cannot be made to run on other systems. A specialized HDM software package is, at the time of writing, still in the planning stage.

4.7 Discussion

We have introduced a hierarchical extension of the Wiener diffusion model for two-choice response times (hierarchical diffusion model or HDM). With a small set of examples, we have demonstrated the feasibility of the HDM. In strictly formal terms, the HDM is just another nonlinear mixed model (NLMM), but it is special because it uses a bivariate measurement level. This new data analysis approach is characterized by great flexibility compared to existing treatments of choice response time data. Additionally, by using the Wiener diffusion process as the measurement level, the hierarchical analysis can be performed on parameters that have well-defined substantive interpretations.

The substantively interesting process interpretation of the diffusion model parameters is important for several reasons. Firstly, it is particularly

appealing in the context of Borsboom's (2006) view that the fact that measurement models lack substantive foundation is the main reason psychometrics has had a limited impact. That is, while the analysis of choice response time data in a hierarchical framework has already been addressed in the psychometric literature (van der Linden, in press), that approach has been strictly psychometrical and does not have the advantage of an interpretable process model as measurement model.

Secondly, thanks to the substantively interesting process interpretation of the diffusion model parameters, the HDM framework is an instance of *cognitive psychometrics*, a relatively young subdiscipline of psychology. In this subdiscipline, models of cognition are extended to encompass individual differences (i.e., participants are no longer considered as mere replications of one another; Batchelder, 1998) in order to allow for population-level inferences. This strategy has gained some momentum recently, with several articles applying hierarchical models to pool data over participants (e.g., Morey, Pratte, & Rouder, in press; Morey, Rouder, & Speckman, 2008; Hoffman & Rovine, 2007; Wetzels, Vandekerckhove, Tuerlinckx, & Wagenmakers, in press), but the approach is presently far from mainstream.

We have elected to implement the HDM using Bayesian statistical methods. This choice was influenced by many factors, both practical and principled. An important corollary of the Bayesian framework is that results from such an analysis have direct and often intuitive interpretations. In one of the examples, we derived posterior distributions of ANOVA contrasts, from which we could directly draw (probabilistic) inferences regarding the hypotheses at hand.

In order to facilitate the dissemination of hierarchical models (i.e., cognitive psychometrics) into mainstream cognitive science, we have provided software with which a hierarchical model for two-choice response time data

can be put into practice. While this software has some limitations (in particular, somewhat inefficient sampling), we believe it may be useful for a wide audience.

References

Atkinson, R. C., & Shiffrin, R. M. (1968). The psychology of learning and motivation. In K. W. Spence & J. T. Spence (Eds.), (Vol. 2, pp. 89–105). New York: Academic Press.

Batchelder, W. H. (1998). Multinomial processing tree models and psychological assessment. *Psychological Assessment*, *10*, 331–344.

Batchelder, W. H., & Riefer, D. M. (1999). Theoretical and empirical review of multinomial process tree modeling. *Psychonomic Bulletin & Review*, *6*, 57–86.

Borsboom, D. (2006). The attack of the psychometricians. *Psychometrika*, *71*, 425–440.

Brown, S., & Heathcote, A. (in press). The simplest complete model of choice reaction time: Linear ballistic accumulation. *Cognitive Psychology*.

Brysbaert, M., Van Wijnendaele, I., & De Deyne, S. (2000). Age-of-acquisition effects in semantic processing tasks. *Acta Psychologica*, *104*, 215–226.

Chater, N., Tenenbaum, J. B., & Yuille, A. (2006). Probabilistic models of cognition: Conceptual foundations. *Trends in Cognitive Sciences*, *10*(7), 287–291.

Clark, H. (1973). The language-as-fixed-effect fallacy: A critique of language statistics in psychological research. *Learning and Verbal Behavior*, *12*, 335–359.

Congdon, P. (2003). *Applied Bayesian models*. New York: Wiley.

Cronbach, L. (1957). The two disciplines of scientific psychology. *American Psychologist*, *12*, 671–684.

Cronbach, L. (1975). Beyond the two disciplines of scientific psychology. *American Psychologist*, *30*, 116–127.

De Boeck, P., & Wilson, M. (2004). *Explanatory item response models: A generalized linear and nonlinear approach.* New York: Springer.

Dutilh, G., Vandekerckhove, J., Tuerlinckx, F., & Wagenmakers, E.-J. (2008). *A diffusion model account of practice.* (Manuscript submitted for publication.)

Eastman, K., Stankiewicz, B., & Huk, A. (2007). Optimal weighting of speed and accuracy in a sequential decision–making task. *Journal of Vision*, *7*(9), 428.

Gelman, A., Carlin, J. B., Stern, H. S., & Rubin, D. B. (2004). *Bayesian data analysis (2nd ed.).* Boca Raton, FL: Chapman & Hall/CRC.

Gelman, A., & Hill, J. (2007). *Data analysis using regression and multilevel/hierarchical models.* Cambridge, MA: Cambridge University Press.

Gonzalez, J., Tuerlinckx, F., & De Boeck, P. (in press). Crossed random–effects models for the analysis of complex designs. *Psychological Methods*.

Griffiths, T. L., Kemp, C., & Tenenbaum, J. B. (2008). Bayesian models of cognition. In R. Sun (Ed.), *Cambridge handbook of computational cognitive modeling* (pp. 59–100). Cambridge, MA: Cambridge University Press.

Hoffman, L., & Rovine, M. J. (2007). Multilevel models for the experimental psychologist: foundations and illustrative examples. *Behavior Research Methods*, *39*(1), 101–117.

Jackman, S. (2000). Estimation and inference via Bayesian simulation: An introduction to Markov chain Monte Carlo. *American Journal of*

Political Science, *44*(2), 375–404.

Janssen, R., Tuerlinckx, F., Meulders, M., & De Boeck, P. (2000). A hierarchical IRT model for criterion-referenced measurement. *Journal of Educational and Behavioral Statistics*, *25*, 285–306.

Klauer, K. C., Voss, A., Schmitz, F., & Teige-Mocigemba, S. (2007). Process components of the Implicit Association Test: A diffusion–model analysis. *Journal of Personality and Social Psychology*, *93*, 353–368.

Laming, D. R. J. (1968). *Information theory of choice-reaction times*. London, UK: Academic Press.

Lee, M. D. (2008). Three case studies in the Bayesian analysis of cognitive models. *Psychonomic Bulletin & Review*, *15*, 1–15.

Link, S. W., & Heath, R. A. (1975). A sequential theory of psychological discrimination. *Psychometrika*, *40*, 77–105.

Lunn, D., Thomas, A., Best, N., & Spiegelhalter, D. (2000). WinBUGS — a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing*, *10*, 325–337.

Molenberghs, G., & Verbeke, G. (2006). *Models for discrete longitudinal data*. New York: Springer.

Morey, R. D., Pratte, M. S., & Rouder, J. N. (in press). Problematic effects of aggregation in zROC analysis and a hierarchical modeling solution. *Journal of Mathematical Psychology*.

Morey, R. D., Rouder, J. N., & Speckman, P. L. (2008). A statistical model for discriminating between subliminal and near–liminal performance. *Journal of Mathematical Psychology*, *52*, 21–36.

Navarro, D. J., & Fuss, I. (2008). *Fast and accurate calculations for first–passage times in Wiener diffusion models.* (Manuscript in preparation.)

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*,

85, 59–108.

Ratcliff, R. (1981). A theory of order relations in perceptual matching. *Psychological Review*, *88*, 552–572.

Ratcliff, R. (1985). Theoretical interpretations of the speed and accuracy of positive and negative responses. *Psychological Review*, *92*, 212–225.

Ratcliff, R. (2002). A diffusion model account of response time and accuracy in a brightness discrimination task: Fitting real data and failing to fit fake but plausible data. *Psychonomic Bulletin & Review*, *9*, 278–291.

Ratcliff, R., Gomez, P., & McKoon, G. (2004). Diffusion model account of lexical decision. *Psychological Review*, *111*, 159–182.

Ratcliff, R., & McKoon, G. (1988). A retrieval theory of priming in memory. *Psychological Review*, *95*, 385–408.

Ratcliff, R., & Rouder, J. N. (1998). Modeling response times for two-choice decisions. *Psychological Science*, *9*, 347–356.

Ratcliff, R., & Rouder, J. N. (2000). A diffusion model account of masking in two-choice letter identification. *Journal of Experimental Psychology: Human Perception and Performance*, *26*, 127–140.

Ratcliff, R., & Smith, P. L. (2004). A comparison of sequential sampling models for two-choice reaction time. *Psychological Review*, *111*, 333–367.

Ratcliff, R., Thapar, A., & McKoon, G. (2001). The effects of aging on reaction time in a signal detection task. *Psychology and Aging*, *16*, 323–341.

Ratcliff, R., & Tuerlinckx, F. (2002). Estimating parameters of the diffusion model: Approaches to dealing with contaminant reaction times and parameter variability. *Psychonomic Bulletin & Review*, *9*, 438–481.

Ratcliff, R., Van Zandt, T., & McKoon, G. (1999). Connectionist and

diffusion models of reaction time. *Psychological Review*, *102*, 261–300.

Raudenbush, S. W., & Bryk, A. S. (2002). *Hierarchical linear models: Applications and data analysis methods*. Newbury Park, CA: Sage.

Riefer, D. M., & Batchelder, W. H. (1988). Multinomial modeling and the measurement of cognitive processes. *Psychological Review*, *95*, 318–339.

Riefer, D. M., Knapp, B., Batchelder, W. H., Bamber, D., & Manifold, V. (2002). Cognitive psychometrics: Assessing storage and retrieval deficits in special populations. *Psychological Assessment*, *14*, 184–201.

Rijmen, F., Tuerlinckx, F., De Boeck, P., & Kuppens, P. (2003). A nonlinear mixed model framework for item response theory. *Psychological Methods*, *8*, 185–205.

Robert, C. P., & Casella, G. (2003). *Monte Carlo statistical methods*. New York: Springer.

Rouder, J. N., & Lu, J. (2005). An introduction to Bayesian hierarchical models with an application in the theory of signal detection. *Psychonomic Bulletin & Review*, *12*, 573-604.

Rouder, J. N., Lu, J., Speckman, P. L., Sun, D., & Jiang, Y. (2005). A hierarchical model for estimating response time distributions. *Psychonomic Bulletin & Review*, *12*, 195–223.

Rouder, J. N., Lu, J., Sun, D., Speckman, P. L., Morey, R. D., & Naveh-Benjamin, M. (2007). Signal detection models with random participant and item effects. *Psychometrika*, *72*, 621–642.

Singer, J. D., & Willett, J. (2003). *Applied longitudinal data analysis: Modeling change and event occurrence*. New York: Oxford University Press.

Skrondal, A., & Rabe-Hasketh, S. (2004). *Generalized latent variable*

modeling: Multilevel, longitudinal, and structural equation models. Boca Raton, FL: Chapman and Hall/CRC.

Sternberg, S. (1966). High speed scanning in human memory. *Science*, *153*, 652654.

Strayer, D., & Kramer, A. (1994). Strategies and automaticity: I. *Basic findings and conceptual framework. Journal of Experimental Psychology: Learning, Memory, & Cognition*, *20*, 318–341.

Thapar, A., Ratcliff, R., & McKoon, G. (2003). A diffusion model analysis of the effects of aging on letter discrimination. *Psychology & Aging*, *18*, 415–429.

Tuerlinckx, F. (2004). The efficient computation of the distribution function of the diffusion process. *Behavior Research Methods, Instruments, & Computers*, *36*, 702-716.

Tuerlinckx, F., & De Boeck, P. (2005). Two interpretations of the discrimination parameter. *Psychometrika*, *70*, 629–650.

Tuerlinckx, F., Maris, E., Ratcliff, R., & De Boeck, P. (2001). A comparison of four methods for simulating the diffusion process. *Behavior Research Methods, Instruments, & Computers*, *33*, 443–456.

van der Linden, W. J. (in press). A hierarchical framework for modeling speed and accuracy on test items. *Psychometrika*.

Vandekerckhove, J., Panis, S., & Wagemans, J. (2007). The concavity effect is a compound of local and global effects. *Perception & Psychophysics*, *69*, 1253–1260.

Vandekerckhove, J., & Tuerlinckx, F. (2007). Fitting the Ratcliff diffusion model to experimental data. *Psychonomic Bulletin & Review*, *14*, 1011–1026.

Vandekerckhove, J., Tuerlinckx, F., & Lee, M. D. (2008). A Bayesian approach to diffusion process models of decision-making. In V. Sloutsky, B. Love, & K. McRae (Eds.), *Proceedings of the 30th annual confe-*

rence of the cognitive science society (pp. 1429–1434). Austin, TX: Cognitive Science Society.

Verbeke, G., & Molenberghs, G. (2000). *Linear mixed models for longitudinal data*. New York: Springer.

Voss, A., Rothermund, K., & Voss, J. (2004). Interpreting the parameters of the diffusion model: An empirical validation. *Memory & Cognition*, *32*, 1206–1220.

Wagenmakers, E.-J. (in press). Methodological and empirical developments for the Ratcliff diffusion model of response times and accuracy. *European Journal of Cognitive Psychology*.

Wagenmakers, E.-J., Ratcliff, R., Gomez, P., & McKoon, G. (2007). A diffusion model account of criterion manipulations in the lexical decision task. *Journal of Memory and Language*, *58*, 140–159.

Wetzels, R., Vandekerckhove, J., Tuerlinckx, F., & Wagenmakers, E.-J. (in press). Bayesian parameter estimation in the expectancy valence model of the iowa gambling task. *Journal of Mathematical Psychology*.

Wichmann, F. A., & Hill, N. J. (2001). The psychometric function: I. fitting, sampling and goodness-of-fit. *Perception and Psychophysics*, *63*, 1293-1313.

4.A Software implementation of the HDM

In this Appendix, we present `wiener.odc` and `wienereta.odc`, two pieces of Component Pascal code that can be incorporated into the popular Bayesian computation program WinBUGS (Lunn et al., 2000). With `wiener.odc` and `wienereta.odc` installed, WinBUGS's full range of general-purpose Markov chain Monte Carlo (MCMC) methods can be applied to the Wiener diffusion's two-choice reaction time distribution.

This extension to WinBUGS has been successfully applied to Ratcliff and Rouder's (1998) benchmark data by Vandekerckhove, Tuerlinckx, and Lee (2008).

In what follows, we will first provide instructions on the installation and use of `wiener.odc` and `wienereta.odc`. We will also provide some example code for a basic analysis and an example using a mixed-model Wiener process. We will finish with some warnings regarding (computational) limitations to the code.

Installing the files

The `wiener.odc` and `wienereta.odc` files are released under the GNU General Public License, which can be accessed via `http://www.gnu.org/copyleft/gpl.html`. The files can be obtained from the first author.

Required materials

In order to use these files, you need to download and install three pieces of software, all of which are freely available on the internet. Install them *in the order given*. If you already have BlackBox installed, read the WinBUGS development page for instructions (`http://www.winbugs-development.org.uk/`).

1. *WinBUGS.* This is the basic program you will be using. It can be downloaded from `http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/contents.shtml`. You need to register to get a key, but registration is free. Download and install the most recent version (at the time of writing, version 1.4.3).

2. *WinBUGS Development Interface (WBDev).* To be downloaded via `http://www.winbugs-development.org.uk/`. Unzip the exe-

cutable to your WinBUGS directory. Then open, with WinBUGS, the `wbdev\_01\_09\_04.txt` file that has appeared there and follow the instructions at the top of the file.

3. *BlackBox Component Builder.* This is an integrated development environment for programs written in Component Pascal (as WinBUGS is). It can be freely downloaded from `http://www.oberon.ch/blackbox.html`. This page also has a tutorial on Component Pascal, which may be useful in case you would like to write your own distributions or adapt the `wiener.odc` file. The most recent version of this program is 1.5 at the time of writing. Note that BlackBox Component Builder only runs on Windows platforms.

Download and install these three programs. Install WinBUGS in `/Program Files/WinBUGS` and BlackBox in `/Program Files/BlackBox Component Builder 1.5`. WBDev will have created its own directory `/WinBUGS/WBDev`.

Preparing BlackBox and compiling the ODC files

In your browser, open the WinBUGS directory and select all files (**Ctrl+A**) and copy them (**Ctrl+C**). Then open the BlackBox directory and paste those files there (**Ctrl+V**). Select "Yes to all" if asked about replacing files. Once this is done, you will be able to open BlackBox and run WinBUGS from inside it.

Now copy the files `wiener.odc` and `wienereta.odc` to the `/BlackBox Component Builder 1.5/WBDev/Mod` directory and then use BlackBox to open it. Press **Ctrl+K** to compile the distribution.

Now open the file `/BlackBox Component Builder 1.5/WBDev/Rsrc/Distributions.odc` and add the following lines of text to the end of the file (right above the `END` statement):

```
s ~ "dwiener"(s, s, s, s)I(s, s)
"WBDevWiener.Install"

s ~ "dwiener.eta"(s, s, s, s, s)I(s, s)
"WBDevWienerEta.Install".
```

Restart BlackBox to begin using the new distributions.

Using the distribution

Difference between wiener.ODC and wienereta.ODC

It is a common practice to assume trial-to-trial variability of the diffusion model parameters. While this is in principle easy in a Bayesian context, numerical problems arise in the `wiener.odc` function that make that variability in drift rate becomes difficult to estimate. For this reason, we also provide `wienereta.odc`. This file directly implements a diffusion model with trial-to-trial variability in drift rate (using the logarithm of Eq. 30 in Tuerlinckx, 2004, for the correct PDF for this case) and is numerically more robust for this case. In all other respects, the files are the same.

Formatting two-choice reaction time data

As user–contributed distributions in WinBUGS are necessarily unidimensional, we need to apply a trick to get it to accept the bivariate diffusion PDF. Consider that one dimension of the PDF is binary (response, denoted x_i), and the other is defined only on the positive half–line (reaction time, denoted t_i). It follows that a distribution on the full real line is defined by:

$$y_i = \begin{cases} t_i & \text{if } x_i = 1 \\ -t_i & \text{if } x_i = 0 \end{cases}$$

When using `wiener.odc` or `wienereta.odc`, you will need to code your two–choice reaction time to match the format of y_i. The functions will internally convert the negative response times to positive–valued error responses and treat the distribution as bivariate for the calculation of the likelihood value.

Examples of usage

Basic usage in WinBUGS

In order to implement `wiener.odc`, simply use a line like

```
s ~ dwiener(alpha, zinit, tau, delta)
```

in your WinBUGS code where `zinit` is the starting point in absolute value (i.e., $\alpha\beta$). Below we give a simple example for a data set with nc conditions (labeled $1, \ldots, nc$) and a total of N data points. The responses (properly formatted, in seconds, with positive and negative numbers) are stored in the variable y and condition indicators are in *cond*. This model assumes boundary separation alpha, bias beta, and nondecision time tau to be constant across conditions, but allows drift rate delta to differ (i.e., a fixed effect of condition on drift rate).

```
Fit a simple Wiener diffusion model {
    # Define priors on parameters
    beta  ~ dunif(0.1,0.9)
    tau   ~ dunif(0.05,1.00)
    alpha ~ dunif(0.03,0.25)

    # Insert a 'plate' to define multiple
    # deltas
    for (r in 1:nc)
```

```
    {
        delta[r] ~ dunif(-0.75,0.75)
    }

    # Compute zinit from alpha and beta
    zinit <- alpha*beta

    # Connect the data to the Wiener process
    for (i in 1:N)
    {
        y[i] ~ dwiener(alpha,tau,zinit,
                       delta[cond[i]])
    }
}
```

Advanced usage 1: Mixed model on nondecision time

In many cases, it is desirable to allow a parameter to vary from trial to trial according to a certain distribution (i.e., to use a "mixed model"). In the classical statistical framework, this leads to complicated integrals in the likelihood function. For example, to allow the nondecision time parameter to vary over trials, the likelihood function for conditions $p = 1, \ldots, P$ and items $i = 1, \ldots, I$ becomes complicated as in Equation 4.5, where the diffusion PDF is given in Equation 4.6 and $\mathrm{TN}(\mu, \sigma, L, U)$ indicates the truncated normal distribution with mean μ, standard deviation σ and lower and upper bounds L and U, respectively.

$$L(t_{ip}, x_{ip}|\alpha_p, \zeta_p, \delta_p, \mu_\tau, \sigma_\tau^2, L, U) = \prod_{p=1}^{P}\prod_{i=1}^{I}\int_{-\infty}^{+\infty} \text{Diffusion}(t_{ip}, x_{ip}|\alpha_p, \tau, \zeta_p, \delta_p)\times \text{TN}\left(\tau|\mu_\tau, \sigma_\tau^2, L, U\right) \mathrm{d}\tau \quad (4.5)$$

$$\text{Diffusion}(t, x|\alpha, \tau, \zeta, \delta) = \begin{cases} \frac{\pi s^2}{\alpha^2}\exp\left(-\frac{\zeta\delta}{s^2}\right) \\ \qquad \times \sum_{j=1}^{+\infty} j \sin\left(\frac{\pi j\zeta}{\alpha}\right)\exp\left[\frac{\tau-t}{2}\left(\frac{\delta^2}{s^2}+\frac{\pi^2 j^2 s^2}{\alpha^2}\right)\right] & \text{if } x=0 \\ \frac{\pi s^2}{\alpha^2}\exp\left(-\frac{(\alpha-\zeta)\delta}{s^2}\right) \\ \qquad \times \sum_{j=1}^{+\infty} j \sin\left(\frac{\alpha-\zeta}{\alpha}\pi j\right)\exp\left[\frac{\tau-t}{2}\left(\frac{\delta^2}{s^2}+\frac{\pi^2 j^2 s^2}{\alpha^2}\right)\right] & \text{if } x=1 \end{cases} \quad (4.6)$$

In a classical statistical context, this integration significantly increases the computational cost of the likelihood function, which needs to be evaluated many times in order to numerically find the parameters corresponding to its maximum (see, e.g., Vandekerckhove & Tuerlinckx, 2007). In a Bayesian context, however, this integration can be performed by the MCMC algorithm, and does not pose further computational issues. The following code performs just such an analysis in WinBUGS, where

$$\boldsymbol{Y}_{(ij)} \sim W\left(\alpha_{(i)}, \beta, \tau_{(ij)}, \delta\right)$$

and

$$\tau_{(ij)} \sim TN\left(\theta, \chi^2, 0, 1\right).$$

```
Fit a Wiener diffusion model with mixing over
nondecision time {
    # Define priors on parameters
    delta ~ dunif(-0.9,0.9) # assume only one
                            # drift rate now
    beta ~  dunif(0.01,0.99)

    # but suppose different boundary
    # separations
    for (i in 1:nc)
    {
        alpha[i] ~ dunif(0.03,0.40)
        zinit[i] <- alpha[i]*beta
    }

    # Use a truncated normal distribution for
    # tau, with mean theta and standard
    # deviation chi
    theta ~ dunif(0.05,0.80)
    chi   ~ dgamma(0.001,0.001)

    # Note that, for the parametrization of the
    # normal distribution, WinBUGS uses
    # 1/variance (precision) instead of the
    # standard deviation
    precision <- pow(chi,-2)

    # Connect the data to the Wiener process
    # but add the tau distribution as well
    for (i in 1:N)
    {
        # Use I(X,Y) to truncate below X and
        # above Y
        tau[i] ~ dnorm(theta,precision)I(0,1)
```

```
        y[i]   ~ dwiener(alpha[cond[i]],tau[i],
                         zinit[cond[i]],delta)
    }
}
```

Advanced usage 2: Mixed model on drift rate

In order to apply a mixed model on drift rate, the above method may lead to numerical instability in the computation of the PDF. For this reason, we have also provided `wienereta.odc`, which is optimized for this case. In this implementation, the drift rate δ for each individual trial is assumed to be a draw from a normal distribution with mean ν and standard deviation η. Accordingly, when using the distribution, a fifth input parameter is required. Thus, in order to use `wienereta.odc`, type

```
s ~ dwiener.eta(alpha, zeta, tau, nu, eta)
```

The parameter η (`eta`) should be restricted (in its prior) to be positive. The following is some example code for using `wienereta.odc`.

```
Fit a Wiener diffusion model {
    # Define priors on parameters
    beta  ~ dunif(0.01,0.99)
    tau   ~ dunif(0.05,0.80)
    alpha ~ dunif(0.03,0.50)
    nu    ~ dunif(-0.6,0.6)
    eta   ~ dunif(0,0.4)

    # Compute zinit from alpha and beta
    zinit <- alpha*beta
```

```
    # Connect the data to the Wiener process
    for (i in 1:N)
    {
        y[i] ~ dwiener.eta(a,ter,z,v,eta)
    }
}
```

Advanced usage 3: Posterior predictive values

The `wiener.odc` and `wienereta.odc` files are equipped with efficient simulators for two–choice reaction time data under a Wiener diffusion model (Tuerlinckx, Maris, Ratcliff, & De Boeck, 2001). This sampler has two uses. Firstly, it can be employed in a simulation study (e.g., for power analysis). Secondly, and more importantly, it can be used to generate posterior predictive values (Gelman et al., 2004). Applying this in WinBUGS is straightforward, and a simple example is given below. WinBUGS can then be made to output these posterior predictive samples to a so-called coda file, which can be read by an external program (e.g., MATLAB or R), which can then compute summary statistics on the posterior predictives and compare the distributions of these samples to the values found in the data.

```
Fit a Wiener diffusion model {
    # Define priors on parameters
    delta ~ dunif(-0.9,0.9)
    beta ~ dunif(0.01,0.99)
    alpha ~ dunif(0.03,0.50)
    zinit <- alpha*beta
```

```
    tau ~ dunif(0.05,0.80)

    # Connect the data to the Wiener process
    for (i in 1:N)
    {
        y[i] ~ dwiener(alpha,tau,zinit,delta)
    }

    # Use the 'cut' function to prevent WinBUGS
    # from including the PPF in the posterior
    alpha.ppf <- cut(alpha)
    tau.ppf <- cut(tau)
    zinit.ppf <- cut(zinit)
    delta.ppf <- cut(delta)

    # Generate the PPF with the Wiener sampler
    for (i in 1:N)
    {
        ppf[i] ~ dwiener(alpha.ppf,tau.ppf,
                         zinit.ppf,delta.ppf)
    }
}
```

4.B WinBUGS code for the example applications

In this Appendix, we provide the exact WinBUGS code we used for the three example applications. There are seven models in total: BM1, BM2, and BM3 for the first application, DLM, DEM, and TEM for the second application, and PHM for the third application.

```
Benchmark data, Model BM1 {
   # Prior for beta
   beta ~ dunif(0.1,0.9)

   # Insert a plate over instruction conditions
   for (i in 1:2)
   {
      # Prior for alphas
      alpha[i] ~ dunif(0.03,0.25)

      # The Wiener distribution code works with
      # the zeta_init parameter and not with
      # beta. So we have to rescale the entire
      # mixing distribution in a somewhat
      # complicated way.

      # The mean of the uniform mixing
      # distribution is alpha*beta
      z[i] <- alpha[i]*beta

      # Now we work on the range of the
      # uniform. The range is constrained
      # by twice the distance to the nearest
      # boundary. So we create a logical node
      # to store this distance (either
      # alpha-zeta_init or zeta_init)
      edges[i,1] <- z[i]
      edges[i,2] <- alpha[i]-z[i]
      szmax[i] <- 2*min(edges[i,1],edges[i,2])

      # Having the maximum of the uniform's
      # range stored, we now create a node
      # with prior U(0,szmax) by rescaling a
```

```
    # U(0,1) prior. Note that the Beta(1,1)
    # prior is the same as a U(0,1) prior.
    sztmp[i] ~ dbeta(1,1)
    sz[i] <- sztmp[i]*szmax[i]
    # sz now has the correct prior

    # To apply the mixing distribution later,
    # we store the lower bound of the
    # trial-to-trial mixing distribution.
    zlo[i] <- z[i]-sz[i]/2

    # Insert a plate over brightness
    # conditions and set a population
    # distribution for the condition-specific
    # drift rates.
    for (s in 1:nc)
    {
       # Mean of the population distribution
       # is determined by a single parameter
       # in this particular model.
       nu.hat[s,i] <- mu
       nu[s,i] ~ dnorm(nu.hat[s,i],prec)
                                  I(-.95,.95)
    }
}

# The mean has a prior
mu ~ dunif(-.7,.7)

# The standard deviation of the population
# distribution is sigma_epsilon but needs to
# be transformed into a precision (1/var).
sigma.epsilon ~ dunif(0.0001,0.6)
prec <- pow(sigma.epsilon,-2)
```

```
# The trial-to-trial mixing of nondecision
# time also needs a mean and a precision.
theta ~ dbeta(1,1)
chi ~ dbeta(1,1)
pt <- pow(chi,-2)

# The trial-to-trial mixing of drift rate is
# coded directly into the distribution file,
# so we don't need to transform the standard
# deviation eta. The mean of the mixing
# distribution is the nu parameter that was
# defined above.
eta ~ dunif(0.0001,.4)

# Now we 'loop' over data points. We have
# covariate vectors 'stim' and 'ins' that
# tell us in which stimulus/instruction
# condition trial j was (ins: 1 for speed,
# 2 for accuracy).
for (j in 1:N)
{
    # Trial-specific nondecision time tau.
    tau[j] ~ dnorm(theta,pt)I(0,2)

    # Trial-specific zeta_init is again a
    # rescaled U(0,1). The dependence on the
    # instruction condition is expressed by
    # using the 'ins' covariate as an index.
    ztemp[j] ~ dbeta(1,1)
    zinit[j] <- zlo[ins[j]]+
                        sz[ins[j]]*ztemp[j]

    # Finally, the data are connected to the
```

```
      # Wiener distribution with the correct
      # parameters.
      t[j] ~ dwiener.eta(alpha[ins[j]],tau[j],
                         zinit[j],nu[stim[j],
                         ins[j]],eta)
   }
}
```

```
Benchmark data, Model BM2 {
   beta ~ dunif(0.1,0.9)

   for (i in 1:2)
   {
      alpha[i] ~ dunif(0.03,0.25)
      z[i] <- alpha[i]*beta
      edges[i,1] <- z[i]
      edges[i,2] <- alpha[i]-z[i]
      szmax[i] <- 2*min(edges[i,1],edges[i,2])
      sztmp[i] ~ dbeta(1,1)
      sz[i] <- sztmp[i]*szmax[i]
      zlo[i] <- z[i]-sz[i]/2

      for (s in 1:nc)
      {
         # Mean of the population distribution
         # is now a nonlinear function of s,
         # the stimulus intensity condition.
         pmf[s,i] <- 1-exp(-pow(s
                           /(33*nu.sc),nu.sh))
         nu.hat[s,i] <- nu.lo + (nu.hi-nu.lo)
                                   * pmf[s,i]
```

```
        nu[s,i] ~ dnorm(nu.hat[s,i],prec)
                                    I(-.95,.95)
      }
   }

   # The Weibull link parameters get priors.
   nu.lo ~ dunif(-0.95,-0.2)
   nu.hi ~ dunif(0.2,0.95)
   nu.sc ~ dunif(0.25,0.75)
   nu.sh ~ dunif(1,20)

   sigma.epsilon ~ dunif(0.0001,0.6)
   prec <- pow(sigma.epsilon,-2)

   theta ~ dbeta(1,1)
   chi ~ dbeta(1,1)
   pt <- pow(chi,-2)

   eta ~ dunif(0.0001,.4)

   for (j in 1:N)
   {
       tau[j] ~ dnorm(theta,pt)I(0,2)
       ztemp[j] ~ dbeta(1,1)
       zinit[j] <- zlo[ins[j]]+sz[ins[j]]
                                    *ztemp[j]
       t[j] ~ dwiener.eta(alpha[ins[j]],tau[j],
                          zinit[j], nu[stim[j],
                          ins[j]],eta)
   }
}
```

```
Benchmark data, Model BM3 {
   beta ~ dunif(0.1,0.9)

   for (i in 1:2)
   {
      alpha[i] ~ dunif(0.03,0.25)
      z[i] <- alpha[i]*beta
      edges[i,1] <- z[i]
      edges[i,2] <- alpha[i]-z[i]
      szmax[i] <- 2*min(edges[i,1],edges[i,2])
      sztmp[i] ~ dbeta(1,1)
      sz[i] <- sztmp[i]*szmax[i]
      zlo[i] <- z[i]-sz[i]/2

      # In this model, the Weibull link
      # parameters are allowed to differ as a
      # function of the instruction condition.
      nu.lo[i] ~ dunif(-0.95,-0.2)
      nu.hi[i] ~ dunif(0.2,0.95)
      nu.sc[i] ~ dunif(0.25,0.75)
      nu.sh[i] ~ dunif(1,20)

      for (s in 1:nc)
      {
         pmf[s,i] <- 1-exp(-pow(s/
                        (33*nu.sc[i]),nu.sh[i]))
         nu.hat[s,i] <- nu.lo[i] +
                        (nu.hi[i]-nu.lo[i])
                                    * pmf[s,i]

         nu[s,i] ~ dnorm(nu.hat[s,i],prec)
                                     I(-.95,.95)
      }
```

```
   }

   sigma.epsilon ~ dunif(0.0001,0.6)
   prec <- pow(sigma.epsilon,-2)

   theta ~ dbeta(1,1)
   chi ~ dbeta(1,1)
   pt <- pow(chi,-2)

   eta ~ dunif(0.0001,.4)

   for (j in 1:N)
   {
       tau[j] ~ dnorm(theta,pt)I(0,2)
       ztemp[j] ~ dbeta(1,1)
       zinit[j] <- zlo[ins[j]]+sz[ins[j]]
                                    *ztemp[j]
       t[j] ~ dwiener.eta(alpha[ins[j]],tau[j],
                          zinit[j],nu[stim[j],
                          ins[j]],eta)
   }
}
```

```
This is the Descriptive Learning Model (DLM) {
   # The means of the item population
   # distributions have to be fixed to achieve
   # model identifiability.
   mu.gamma[1] <- 0
   mu.gamma[2] <- 0

   # A plate for word/nonword
```

```
for (w in 1:2)
{
   # The population standard deviations
   # omega are estimable, however. They
   # need to be converted to precisions.
   sigma.gamma[w] ~ dunif(0.0001,.45)
   tau.gamma[w] <- pow(sigma.gamma[w],-2)

   # A plate for the individual stimuli.
   for (s in 1:nstim)
   {
      gamma[s,w] ~ dnorm(mu.gamma[w],
                         tau.gamma[w])
                               I(-.7,.7)
   }

   mu.lambda[w] ~ dunif(-.6,.6)
   sigma.lambda[w] ~ dunif(0.001,0.3)
   tau.lambda[w] <- pow(sigma.lambda[w],-2)

   # A plate for the fixed effect of day.
   # Both the drift rate component lambda
   # and the trial-to-trial standard
   # deviation eta are affected.
   for (d in 1:nd)
   {
      lambda[d,w] ~ dnorm(mu.lambda[w],
                          tau.lambda[w])
                                  I(-.7,.7)
   }
}

# The other parameters also have an effect
# of day.
```

```
# Random effect for alpha, beta and theta:
mu.alpha ~ dunif(0.03,0.20)
sigma.alpha ~ dunif(0.0001,0.15)
tau.alpha <- pow(sigma.alpha,-2)

mu.theta ~ dunif(0,1)
sigma.theta ~ dunif(0.0001,0.2)
tau.theta <- pow(sigma.theta,-2)

mu.beta ~ dunif(0.35,0.65)
sigma.beta ~ dunif(0.0001,0.2)
tau.beta <- pow(sigma.beta,-2)

# Finally, trial-to-trial variability
chi ~ dunif(0.001,0.2)
chi.inv <- pow(chi,-2)
eta ~ dunif(0.001,0.3)

for (d in 1:nd)
{
   alpha[d] ~ dnorm(mu.alpha,tau.alpha)
                                I(0.01,0.25)
   beta[d]  ~ dnorm(mu.beta,tau.beta)
                                I(0.25,0.75)
   theta[d] ~ dnorm(mu.theta,tau.theta)I(0,)
   zeta.init[d] <- beta[d]*alpha[d]

   # Each stimulus drift rate is the sum of
   # a day component and an item component.
   # Other combinations than the additive
   # are possible, and very easy to
   # implement.
   for (s in 1:nstim)
   {
```

```
         nu[d,s,1] <- lambda[d,1]+gamma[s,1]
         nu[d,s,2] <- lambda[d,2]+gamma[s,2]
      }
   }

   for (i in 1:N)
   {
      tau[i] ~ dnorm(theta[day[i]],chi.inv)
                                          I(0,)
      t[i] ~ dwiener.eta(alpha[day[i]],
            tau[i],zeta.init[day[i]],
            nu[day[i],stim[i],wnw[i]],
            eta)
   }
}
```

```
This is the Learning Explanatory Model (LEM) {
   mu.gamma[1] <- 0
   mu.gamma[2] <- 0 # the addition to phi[2]
                    # is made below

   # One new parameter: the regression weight
   # of the covariate.
   rho ~ dunif(-0.1,0.1)

   varpi1[1] ~ dnorm(0,100)
   varpi1[2] ~ dnorm(0,100)
   varpi0[1] ~ dnorm(.3,10)
   varpi0[2] ~ dnorm(-.3,10)

   psi0 ~ dnorm(0.08,1000)
```

```
psi1 ~ dnorm(-4,10)
psi2 ~ dnorm(-1,100)

zeta0 ~ dnorm(.33,100)
zeta1 ~ dnorm(-3.53,10)
zeta2 ~ dnorm(-.68,100)

sigma.alpha ~ dunif(0.0001,0.15)
tau.alpha <- pow(sigma.alpha,-2)
sigma.theta ~ dunif(0.0001,0.2)
tau.theta <- pow(sigma.theta,-2)

for (w in 1:2)
{
   sigma.gamma[w] ~ dunif(0.0001,.45)
   tau.gamma[w] <- pow(sigma.gamma[w],-2)

   for (s in 1:nstim)
   {
      gamma[s,w] ~ dnorm(mu.gamma[w],
                         tau.gamma[w])
                                I(-.7,.7)
   }

   sigma.lambda[w] ~ dunif(0,0.2)
   tau.lambda[w] <- pow(sigma.lambda[w],-2)

   for (d in 1:nd)
   {
      lambda.hat[d,w] <- varpi0[w]
                          + varpi1[w]*(d-3)
      lambda[d,w] ~ dnorm(lambda.hat[d,w],
                          tau.lambda[w])
                                  I(-.7,.7)
```

```
   }
}

chi ~ dunif(0.001,0.2)
chi.inv <- pow(chi,-2)

eta ~ dunif(0.001,0.3)

mu.beta ~ dunif(0.35,0.65)
sigma.beta ~ dunif(0.0001,0.2)
tau.beta <- pow(sigma.beta,-2)

for (d in 1:nd)
{
   alpha.hat[d] <- psi0 + exp(psi1
                              + psi2*(d-3))
   alpha[d] ~ dnorm(alpha.hat[d],tau.alpha)
                               I(0.01,0.25)

   beta[d]  ~ dnorm(mu.beta,tau.beta)
                               I(0.25,0.75)

   zeta.init[d] <- beta[d]*alpha[d]

   theta.hat[d] <- zeta0 + exp(zeta1
                             + zeta2*(d-3))
   theta[d] ~ dnorm(theta.hat[d],
                    tau.theta)I(0,)

   for (s in 1:nstim)
   {
      nu[d,s,1] <- lambda[d,1]+gamma[s,1]
      # Here we add a frequency weight to nu
      nu[d,s,2] <- lambda[d,2]
```

```
                              +gamma[s,2]+rho*0
      }
   }

   for (i in 1:N)
   {
      tau[i] ~ dnorm(theta[day[i]],chi.inv)
                                          I(0,)
      t[i] ~ dwiener.eta(alpha[day[i]],tau[i],
                         zeta.init[day[i]],
                         nu[day[i],stim[i],
                         wnw[i]],eta)
   }
}
```

```
This is the Two-way Explanatory Model (TEM) {
   mu.gamma[1] <- 0
   mu.gamma[2] <- 0 # the addition to phi[2] is
                    # made below

   # One new parameter: the regression weight
   # of the covariate.
   rho ~ dunif(-.1,.1)

   varpi1[1] ~ dnorm(0,100)
   varpi1[2] ~ dnorm(0,100)
   varpi0[1] ~ dnorm(.3,10)
   varpi0[2] ~ dnorm(-.3,10)

   psi0 ~ dnorm(0.08,1000)
   psi1 ~ dnorm(-4,10)
```

```
psi2 ~ dnorm(-1,100)

zeta0 ~ dnorm(.33,100)
zeta1 ~ dnorm(-3.53,10)
zeta2 ~ dnorm(-.68,100)

sigma.alpha ~ dunif(0.0001,0.15)
tau.alpha <- pow(sigma.alpha,-2)
sigma.theta ~ dunif(0.0001,0.2)
tau.theta <- pow(sigma.theta,-2)

for (w in 1:2)
{
   sigma.gamma[w] ~ dunif(0.0001,.45)
   tau.gamma[w] <- pow(sigma.gamma[w],-2)

   for (s in 1:nstim)
   {
      gamma[s,w] ~ dnorm(mu.gamma[w],
                         tau.gamma[w])
                                I(-.7,.7)
   }

   sigma.lambda[w] ~ dunif(0,0.2)
   tau.lambda[w] <- pow(sigma.lambda[w],-2)

   for (d in 1:nd)
   {
      lambda.hat[d,w] <- varpi0[w]
                          + varpi1[w]*(d-3)
      lambda[d,w] ~ dnorm(lambda.hat[d,w],
                          tau.lambda[w])
                                  I(-.7,.7)
   }
```

```
}

chi ~ dunif(0.001,0.2)
chi.inv <- pow(chi,-2)

eta ~ dunif(0.001,0.3)

mu.beta ~ dunif(0.35,0.65)
sigma.beta ~ dunif(0.0001,0.2)
tau.beta <- pow(sigma.beta,-2)

for (d in 1:nd)
{
   alpha.hat[d] <- psi0 + exp(psi1
                              + psi2*(d-3))
   alpha[d] ~ dnorm(alpha.hat[d],tau.alpha)
                               I(0.01,0.25)

   beta[d]  ~ dnorm(mu.beta,tau.beta)
                               I(0.25,0.75)

   zeta.init[d] <- beta[d]*alpha[d]

   theta.hat[d] <- zeta0 + exp(zeta1
                             + zeta2*(d-3))
   theta[d] ~ dnorm(theta.hat[d],tau.theta)
                                     I(0,)

   for (s in 1:nstim)
   {
      nu[d,s,1] <- lambda[d,1]+gamma[s,1]
      # Here we add a frequency weight to nu
      nu[d,s,2] <- lambda[d,2]+gamma[s,2]
                               +rho*freq[s]
```

```
      }
   }

   for (i in 1:N)
   {
      tau[i] ~ dnorm(theta[day[i]],chi.inv)
                                          I(0,)
      t[i] ~ dwiener.eta(alpha[day[i]],
            tau[i],
            zeta.init[day[i]],
            nu[day[i],stim[i],wnw[i]],
            eta)
   }
}
```

```
This is the Population Hierarchical Model (PHM) {
    mu.alpha ~ dunif(0.02,0.30)
    sigma.alpha ~ dunif(0.0001,0.15)
    prec.alpha <- pow(sigma.alpha,-2)

    beta <- 0.5

    mu.theta ~ dunif(0.02,0.70)
    sigma.theta ~ dunif(0.0001,0.15)
    prec.theta <- pow(sigma.theta,-2)

    mu.chi ~ dunif(0.0001,0.15)
    sigma.chi ~ dunif(0.0001,0.10)
    prec.chi <- pow(sigma.chi,-2)

    mu.eta ~ dunif(0.001,0.45)
```

```
    sigma.eta ~ dunif(0.0001,0.15)
    prec.eta <- pow(sigma.eta,-2)

    for(i in 1:nc)
    {
        mu.nu[i] ~ dunif(-.5,.6)
        sigma.nu[i] ~ dunif(0,0.6)
        prec.nu[i] <- pow(sigma.nu[i],-2)
    }

    for(p in 1:np)
    {
        alpha[p] ~ dnorm(mu.alpha,prec.alpha)
                                    I(0.01,0.25)
        zeta_init[p] <- alpha[p]*beta
        theta[p] ~ dnorm(mu.theta,prec.theta)
                                    I(0.01,0.80)
        chi[p] ~ dnorm(mu.chi,prec.chi)
                                I(0.00001,0.49)
        prec.tau[p] <- pow(chi[p],-2)
        eta[p] ~ dnorm(mu.eta,prec.eta)
                                I(0.00001,0.49)
        for(i in 1:nc)
        {
            nu[p,i] ~ dnorm(mu.nu[i],
                            prec.nu[i])
                                    I(-.7,.7)
        }
    }

    for(j in 1:N)
    {
        tau[j] ~ dnorm(theta[pnum[j]],
                       prec.tau[pnum[j]])I(0,)
```

```
        t[j] ~ dwiener.eta(alpha[pnum[j]],
                    tau[j],
                    zeta_init[pnum[j]],
                    nu[pnum[j],cond[j]],
                    eta[pnum[j]])
    }
}
```

CRE

CHAPTER 5

A crossed random effects diffusion model for speeded semantic categorization decisions

Abstract

Choice reaction times (RTs) are often used as a proxy measure of typicality in semantic categorization studies. However, other item properties have been linked to choice RTs as well. We apply a tailored process model of choice RT to a speeded semantic categorization task in order to deconfound different sources of variability in RT. Our model is based on a diffusion model of choice RT, extended to include crossed random effects (of items and participants). This model retains the interesting process interpretation of the diffusion model's parameters, but it can be applied to choice RTs even in the case where there are few or no repeated measurements of each participant-item combination. Different aspects of the response process are then linked to different types of item properties. A typicality measure turns

out to predict the rate of information uptake, while a lexicographic measure predicts the stimulus encoding time. Accessibility measures cannot reliably predict any component of the decision process.

Keywords: semantic categorization; response times; cognitive psychometrics; hierarchical models; diffusion model

5.1 Introduction

In speeded semantic categorization tasks, participants are required to verify whether a lexical item is a true member of some semantic category, and to do so as fast and as accurately as possible. Such tasks have for decades been a primary tool in the study of semantic memory. It is commonly believed that the difference in the time it takes a participant to determine that *apple*[1] is a member of the category ***fruit*** and the time it takes them to determine the same of *lime* may reveal important aspects of the representation of the category ***fruit*** (McCloskey & Glucksberg, 1979; Smith, Shoben, & Rips, 1974).

Historically, various views on the organization of semantic memory have succeeded one another. The types of variables that have been considered as determinants of categorization time differences offer some insight into these different views. In the original work by Landauer and Freedman (1968) and by Collins and Quillian (1970), two factors were considered important determinants of categorization time: the frequency with which lexical items appear in written discourse, and the size of the categories to which these items (supposedly) belong. In later work, researchers turned to associative accounts of the time needed to verify or discard category

[1]Throughout, we will typeset lexical entries in *italics* and categories in ***boldface italics***.

membership. For example, Wilkins (1971) argued that the number of times an item has been associated with the category in the past is an important determinant of the item's categorization time, while Loftus (1973) also made the argument for the importance of the reverse association. The number of times a category has been associated with an item should allow one to predict how long a participant will take to establish the set inclusion relationship between the item and the category. However, it wasn't until Wilkins' production frequency or instance dominance predictor and Loftus' category dominance predictor were complemented by measures of category representativeness that the speeded semantic categorization task achieved its prominence (Larochelle & Pineau, 1994). Up till now the task remains best known for demonstrating that items that are representative or typical of a category are more quickly endorsed than category members that are not (Rips, Shoben, & Smith, 1973; Rosch, 1973). Since the work by Glass and Meany (1978) and by McCloskey (1980) it is now also generally recognized that whenever measures of typicality have an effect, measures of familiarity are likely to be of influence as well.

As a result of these developments, researchers who nowadays are interested in studying speeded semantic categorization decisions have no choice but to include a vast number of covariates to account for categorization time differences. This is especially true in light of the multiple methodological variations the task affords (i.e., presentation order of item and category, choice of negative instances, etc.) and that prevent any single contributor to categorization time variability from emerging (Casey, 1992). The multitude of covariates that have an impact upon semantic categorization time has evoked quite different attitudes towards the task. Some choose to rally against it (e.g., Kintsch, 1980), arguing that the varying findings indicate that existing accounts of the task are underspecified and lack the ability to reveal anything meaningful about the structure of se-

mantic memory. Others see it as an opportunity to investigate the coherence and interaction among the theoretical constructs thought to underlie the various covariates. They have introduced methodological variants of the task and employed multiple regression techniques to disentangle the contributions of the covariates to the resulting categorization time differences (Casey, 1992; Chumbley, 1986; Hampton, 1997; Larochelle & Pineau, 1994; Larochelle, Richard, & Soulires, 2000).

The approach taken in the present paper is informed by both these attitudes towards the speeded semantic categorization task—we believe that current methods may be too weak, and that an in-depth investigation should account for different covariates and their interplay. In the next section, we will argue that clearly specified cognitive process models are interesting tools for extracting information from data that are known to result from processes with multiple sources of variability.

5.1.1 Process models and cognitive psychometrics

The statistical methods we apply in the present improve upon the classical methods in two distinct ways. Firstly, we will apply a process model that is inspired from cognitive psychology. Using a process model allows us to express the data with a concise set of parameters that have interesting psychological interpretations. Secondly, we will apply a hierarchical model in order to allow for differences between persons and between items. That item differences should not be ignored was argued very strongly by Clark (1973) and by Coleman (1964), and the detrimental effects of averaging over persons have been demonstrated by, among others, Estes (1956, 2002) and Heathcote, Brown, and Mewhort (2000). Viewing the model as a whole, each data point in the set will be conceptualized as a single realization of a specific response process, whose parameters are (at each trial)

a unique combination of person-specific and item-specific parameters.

As discussed in the previous section, several item covariates have been shown to covary with semantic categorization RT to some extent. The standard methods for demonstrating these relationships have typically involved general linear models (GLMs). That is, they have focused predominantly on the mean RT (often after log-transformation). Others have focused on accuracy scores, or performed person-specific regressions (and then averaged the results). However, there have been repeated calls for extracting more of the information that is available from RTs (e.g., Heathcote, Popiel, & Mewhort, 1991). An alternative for this standard type of analysis is to focus rather on the response process that governs the participants' behavior (or their interaction with the items). Process modeling is very similar to usual statistical modeling in that a set of assumptions are made about regularities that are presumed to be present in the data, a set of parameters is defined that together give rise to a certain range of predicted distributions of data, and then from the empirical distributions the parameters are estimated using these predictions as a template.

For the general linear model, the assumptions are well-known: (1) the criterion is in reality a linear combination of the predictors, (2) any deviation from this pattern is noise, which follows a normal distribution with mean zero, and (3) the variance of the noise distribution is constant and independent of the predictors. These assumptions might seem quite strict, but they provide mathematical convenience and are familiar—it is quite easy to estimate the parameters of this model with readily available ('off-the-shelf') methods. In process modeling, the genesis of a model works from a different direction: assumptions about the process are made first (based on theoretical insights and prior knowledge about the world) and mathematical convenience is considered only after that. Of course, convenience decisions still come into play, but typically the plausibility of the

process and the interpretability of its parameters are paramount. In the artificial category literature, process models already abound (e.g., Lamberts, 2000; Nosofsky & Palmeri, 1997), but in the domain of natural language categories they are largely unexplored.

A major advantage of this approach is that it occasionally allows us to specify different, possibly independent, components of a process that together generate the response behavior in an experimental task. In the specific case of a choice response task (like the semantic categorization task) it makes sense to assume that there is more than one factor at work in the response process at any given trial. In the model that we will use (cf. infra), separate parameters are included for a person's ability in the task (i.e., their propensity to give a correct response, irrespective of the item properties), but also of their carefulness and the speed with which they are able to execute a motor response—all parameters that can reasonably be taken to influence the eventual RT. Additionally, there are separate parameters for the degree to which an item evokes a correct[2] response, and how long it takes to encode it before a semantic decision is made. This allows for an analysis with a level of detail that is not possible with conventional methods like the general linear model.

When applying such a cognitive process model, we will explicitly allow for individual differences (both between participants and between items) by embedding the model in a *hierarchical* structure. We will in other words assume that while individuals (or items) are not identical in their cognitive process parameters, they are all members of some superordinate population. In this way, hierarchical models are a compromise between assuming

[2]It should be noted here that the accuracy of a categorization response can be somewhat subjective. For example, is a *tomato* a ***vegetable*** or a ***fruit***? Is a *raft* a ***vehicle***? For the purposes of the present paper "true category membership" was determined a-priori by the experimenters, but was kept uniform across the different data sets (see section Data sets for more details).

that all participants are interchangeable (Batchelder, 2007) and can hence be averaged over (possibly leading to averaging artifacts; Estes, 1956, 2002; Heathcote et al., 2000), and assuming that they share no commonality at all. This hierarchical structure is a second (but equally important) way in which our method improves upon the traditional approach. As an additional feature of hierarchical models, we will be able to (attempt to) explain part of the observed variance in parameters, through the use of covariate information (De Boeck & Wilson, 2004). Using a process model in this fashion is sometimes called *cognitive psychometrics* (Batchelder, 2007; Batchelder & Riefer, 1999).

5.1.2 Paper outline

The outline of the rest of the paper is as follows. In the next section, we will briefly describe one data set (due to De Deyne, 2008) that contains speeded semantic categorization data. Then we describe the so-called *Leuven data set* (De Deyne et al., 2008) which contains many possible covariates of the categorization time differences observed by De Deyne (2008). Then we will analyze this joint data set using the classical approach (multiple linear regression), but the results will turn out to be inconsistent and unclear. In the section after that, we will describe the *hierarchical diffusion models* (HDM) statistical framework (Vandekerckhove, Tuerlinckx, & Lee, 2008b) which we believe is optimally suited for the analysis of this coupled data set. Then we perform this analysis and discuss the results. We conclude with a discussion of the application of process models for the purpose of disentangling different sources of variability in choice RTs and implications for semantic categorization studies.

5.2 Data sets

5.2.1 Speeded semantic categorization data

The semantic categorization data are due to De Deyne (2008). The participants were eight male and thirty-six female students of the University of Leuven, who were paid the equivalent of $10 per hour for their participation.

Each of these participants provided speeded semantic categorizations decisions for each of eight categories (***birds***, ***fish***, ***insects***, ***mammals***, ***musical instruments***, ***reptiles***, ***tools***, and ***vehicles***). All items that were listed as exemplars of these categories in De Deyne et al. (2008) served as targets in the experiment. As was noted before, an exemplar generation task that was described in Ruts et al. (2004) informed the construction of these lists. This resulted in the inclusion of some items that could not be considered true category members (e.g., *dolphin* as an exemplar of ***fish***). De Deyne (2008) decided not to retain these items as targets for his semantic categorization experiment. In addition, he excluded all items that were composed of more than one word (e.g., *adjustable spanner*). For each category the resulting targets were complemented by an equal number of distractors. For the natural kind categories (***birds***, ***fish***, ***insects***, ***mammals***, and ***reptiles***) related items from the domain of animals constituted the distractors (e.g., *platypus*, *lobster*, *amoeba*, *seahorse*, and *octopus* for the respective categories). For the artefact categories (***musical instruments***, ***tools***, and ***vehicles***) related artefacts served the part (e.g., *microphone*, *camera*, and *container* for the respective categories).

All participants provided categorization decisions for all items. Instructions stressed both speed and accuracy. Following a recommendation by Hampton (1997), De Deyne (2008) opted for a blocked presentation order of categories. At the onset of a block, participants were informed

about the category that would have to be referenced by displaying the category label for 3500ms on the screen. Those targets and distractors that were assigned to that particular category were then presented one by one in a randomized order. Each trial consisted of the presentation of a mask (500ms), a fixation point (500ms), a blank (500ms), and the stimulus word. The stimulus word was presented for a maximum of 1800ms or until the participant responded by pressing one of two buttons on a response-box. A blank screen (800ms) separated consecutive trials. Presenting the items one at a time, blocked per category, should remove the random variance in RT that would occur if a new category label had to be read on each trial.

Participants were familiarized with the procedure through the completion of a practice block. They then completed the experimental blocks in a randomized order.

5.2.2 The Leuven data

In our introduction to the semantic categorization task we already provided a brief overview of the variables that have been found to have an impact upon participants' performance. They are of a diverse nature, including measures that pertain to semantic categories' internal structure (e.g., Typicality), the availability of the categories' items (e.g., Word Frequency and Familiarity) and the co-occurrence of category label and category items in the categories' learning history (e.g., Category Dominance and Production Frequency). In order to disentangle the contributions of these variables to task performance it is crucial that they are collected within a homogeneous population, since cultural or regional differences are known to affect the pattern of intercorrelations (Hampton & Gardiner, 1983; Larochelle & Pineau, 1994). The Leuven data (De Deyne et al., 2008) meet this condition, as all norms were collected within a few years' time with University of Leu-

ven students. They are therefore extremely well suited for the endeavor at hand: the semantic categorization data collection by De Deyne (2008) took place in the very same student population that provided the normative data and all target category members were selected from among the Leuven norms (cf. supra). Hence, the available data allow an investigation of the differences that arise among true category members in speeded semantic categorization.

Following Hampton (1997) we focused on five covariates to account for these differences: Typicality, Familiarity, Word Frequency, Production Frequency, and Word Length. All five variables are included in the Leuven data and below we will briefly remind the reader how they were collected. Although we agree with Hampton that these variables are generally of interest in the speeded semantic categorization literature, the choice for this set of covariates does not constitute a strong theoretical commitment of the authors. Had the Leuven data included a measure of category dominance, for instance, then we would have included it in our analyses. Nor should the absence of variables like imageability or age of acquisition in our analyses be considered as a stance against their role in semantic processing. Our choice for the named five variables merely reflects the aspiration of demonstrating an approach that we believe to be valuable using a set of theoretically justified variables.

5.2.2.1 Typicality (T)

The representativeness of a category's items can be assessed in a variety of ways. One of them requires participants to indicate on a Likert-type scale how typical each category item is of the category. Students who provided typicality ratings for the Leuven data, indicated on a scale ranging from 1 to 20 how typical they found each category member to be (De Deyne et

al., 2008).

5.2.2.2 Familiarity (F)

Familiarity was assessed in a similar way. Participating students stepped through a list of category items, indicating on a five point Likert-type scale how familiar they were with each of the items. A rating of 1 indicated that they had never seen, heard, or used the word before. A score of 2 indicated that they had seen, heard, or used the word just once or twice. A score of 3 indicated that they had sometimes seen, heard, or used the word. A score of 4 indicated that they had seen, heard, or used the word often. A score of 5, finally, indicated that they had seen, heard, or used the word very often.

5.2.2.3 Word frequency (W)

A measure of item availability that is not based on participants' judgements, but on the frequency with which the item appears in written discourse, can also be obtained from the Leuven data. The reported word frequencies in De Deyne et al. (2008) are the logarithmically transformed lemma counts taken from the Dutch CELEX lexical database (Baayen, Piepenbrock, & van Rijn, 1993).

5.2.2.4 Production frequency (P)

For each of the category members, the measure of production frequency that is distributed with the Leuven data tallies how many out of a total of 120 student participants generated the member in response to the category label. For the purposes of all following analyses, the production frequencies were incremented by one and logarithmically transformed.

5.2.2.5 Word length (L)

The variable word length finally, contains the number of characters in each category member. The effect of this lexicographic variable is usually of minor theoretical importance in accounts of semantic categorization and therefore regularly overlooked (imprudently, as we will show).

5.2.2.6 Covariate preprocessing

Each of the covariates was transformed to a standardized scale with mean 0 and standard deviation 0.1. The distractor items (i.e., items that were not true members of the target category) were included in the analysis after the standardization (i.e., their covariate scores were not used to compute the standard deviation of the covariate). For most covariates, we had no information regarding these distractors, and they were assigned a value of 0 accordingly. Only for the variable Word Length (which is of course easy to obtain) were the distractors given a value.

5.3 Regression analysis

We subjected the joint data set to a standard analysis: multiple linear regression. For the present analysis, we removed all error responses and all responses that were faster than 250ms or slower than 1800ms (which was the experimental cut-off). Using the logarithm of RT as the criterion variable, and Typicality T, Length L, Familiarity F, Word Frequency W, Production Frequency P, and category membership C as predictors, the following regression model is obtained:

$$\begin{aligned} \mu_{(pi)} &= \beta_0 + \beta_1 T_{(i)} + \beta_2 L_{(i)} + \beta_3 F_{(i)} + \beta_4 W_{(i)} \\ &\quad + \beta_5 P_{(i)} + \beta_6 C_{(i)} \\ \log(RT_{(pi)}) &\sim N(\mu_{(pi)}, \sigma^2). \end{aligned}$$

In this model, $RT_{(pi)}$ is the RT of person p ($p = 1, \ldots, 45$) to item i ($i, \ldots, I$), $\mu_{(pi)}$ is the predicted value of $\log(RT_{(pi)})$, and σ^2 is the unexplained variance. Note again that since we only have covariate information for target items, all covariates except Word Length L and Category Membership C take the value 0 for all distractor items.

The regression results are summarized in Table 5.1. We have immediately performed inference on these results, and omitted all regression weights that turned out to be not statistically significant. In this way, the table concisely portrays the conclusions that would usually be drawn from the data with respect to sign and significance. Unfortunately, even a cursory inspection of Table 5.1 shows that the picture is inconsistent at best. While the Typicality measure is traditionally found to have a negative effect on RT (i.e., higher Typicality leads to shorter RTs; Rips et al., 1973; Rosch, 1973), in the present data set this effect only surfaces in three out of eight categories. In two categories RT increased with Typicality, and in the remaining three categories, no effect can be discerned. The Length measure is the only one with effects that are consistent across categories, but it only significantly increased RT in three categories. For Familiarity, Word Frequency, and Production Frequency, the regression weights can take either sign, depending on categories.

Retaining error responses or not removing fast and slow responses affected the pattern of significance, but in no case did a consistent pattern arise. Clearly, the classical analysis yields disappointing results. In the next section, we introduce a process model for choice RT with which we will reanalyze the present data.

Table 5.1: *Classical linear regression. The signs of the regression weights whose p-value was less then* 0.05 *are displayed, others are replaced by a dot.*

Category	T	L	F	W	P
birds	·	·	·	·	−
fish	·	·	+	+	·
insects	+	+	−	·	·
mammals	−	·	·	·	+
musical instruments	−	+	+	−	+
reptiles	+	+	·	+	·
tools	·	·	·	·	+
vehicles	−	·	+	·	·

5.4 Hierarchical diffusion models

In the domain of choice RTs, models based on the Wiener diffusion process have garnered significant attention (Ratcliff, 1978). The Wiener diffusion model is one of the broad class of accumulator models where, in this case, a single evidence counter evolves over continuous time until it hits one of two absorbing boundaries. The time to absorption is then related to the RT and which boundary was hit indicates the response given. The model is considered particularly interesting because the parameters that drive the process (explained below) have intuitive interpretations relating to the sequential accumulation of information.

The increasing popularity of the diffusion model for choice RTs is likely due in part to the interesting interpretations of its parameters on the one hand, and the model's ability to naturally account for many empirically observed phenomena on the other (for an excellent review of recent advances with the diffusion model, see Wagenmakers, in press). It is unfortunate, therefore, that the possibilities for application of the diffusion model have thus far been somewhat limited. For example, as noted by

Wagenmakers (in press), fitting the diffusion model to empirical data requires a large number of observations. Importantly, with the methods currently in practice (Ratcliff & Tuerlinckx, 2002; Vandekerckhove & Tuerlinckx, 2007, 2008; Voss & Voss, 2007) it has typically been necessary to have an appreciable number of data points in each cell of the experimental design. That is, some independent replications *under invariant conditions* are required in order to obtain parameter estimates.

As a result, applications of the diffusion model have largely been limited to the analysis of "long" data sets (i.e., a typical psychophysical design where there are few participants, and many repeated trials for each participant and in each condition). A little-explored alternative would be to apply it to "wide" data sets with many participants and few repeated measurements (like the present semantic categorization data set; Hampton, 1997, recommends against repeating items in such a paradigm). Such analyses are more challenging for several reasons. For example, if all participants are analyzed independently of one another, the available number of data points on which the estimates would be based would be very low. On the other hand, it would be unreasonable to keep many parameters constant across individuals, making it impossible to pool the data together (i.e., to allow sharing of information between data from different participants). Other methods of pooling data across participants (or, for that matter, items), such as quantile averaging (or *vincentizing*; Ratcliff, 1979; Rouder & Speckman, 2004), come with preconditions that may not be met by the diffusion model (i.e., same location-scale family; Thomas & Ross, 1980), they do not permit an investigation of individual differences (in which we might be interested), and they cannot be applied in the case where individual differences are expected on both the person and the item side (i.e., if both persons and items are random draws from their respective populations, and there are no repetitions of person-by-item combination

trials, then there are no distributions left to combine). Taking the statistically principled route of treating participants as random samples from a population (random effects approach to individual differences) typically leads to models that rapidly become quite complex mathematically.

Ratcliff (1978), Laming (1968), and Link and Heath (1975) have already approached part of this problem with a random-effects strategy, by allowing *trial-to-trial variability* in model parameters. Effectively, it is assumed that some parameters change over time in that they are, at each point in time, a random sample from some higher-order distribution. Parameters of this superordinate distribution are then estimated in lieu of the trial-specific parameters themselves. In practice, the variability in a parameter is implemented by multiplying the model's likelihood function with the likelihood function of the trial-to-trial variability and then integrating over the parameter(s) that is (are) allowed to vary (see Ratcliff & Tuerlinckx, 2002; Tuerlinckx, 2004). However, this method is computationally laborious (involving multidimensional integrations that have to be approximated by sums) and somewhat inflexible (in the sense that the likelihood function has to be adapted in such a way that makes it difficult to write a universal but efficient algorithm).

Recently, Vandekerckhove et al. (2008b) have provided a solution to this problem by applying the flexibility of Bayesian hierarchical modeling (see, e.g. Rouder & Lu, 2005; Rouder, Lu, Speckman, Sun, & Jiang, 2005; Rouder et al., 2007 for some examples of Bayesian hierarchical modeling) to the Wiener diffusion process. This statistical framework (HDM) can easily cope with many simultaneous random effects, and software for its implementation is freely available (Vandekerckhove et al., 2008b). A diffusion model with crossed random effects can be applied to a data set where there are no repeated observations in the item-by-participant cells of the experimental design. Such a design would be inaccessible to typical process

model approaches, but it is important in order to account for the random sampling scheme that is normally used for lexical items in the semantic categorization paradigm (Clark, 1973; Coleman, 1964) as well as participants. This crossed random effects diffusion model is especially suited for this case, because other methods that are typically used for dealing with uncontrolled variability (e.g., vincentizing) cannot cope with the crossed random effects design without repeated observations of each cell of the design. Additionally, even if it were possible to have repeated observations of the same person/item combinations (this is not recommended in the semantic categorization context according to Hampton, 1997, but it might be possible in other cases), then the vincentization procedure would only allow us to account for the variability. It does not permit an easy quantification of the variability, nor would it allow us to attempt to explain the variability through external covariates (De Boeck & Wilson, 2004).

5.4.1 Diffusion models

At the basis of the Wiener diffusion model is a random walk process in continuous time and with a continuous state space. The random walk has two boundaries at values α ("upper") and 0 ("lower") and its step-size over a discrete time period t is a randomly drawn value from $N\left(\delta t, \sigma^2 t\right)$ (Feller, 1970). By convention, $\sigma = 0.1$. After a number of steps, the process will hit one of its boundaries (see Fig. 5.1). If δ, called the *drift rate* (or *drift* for short), is high in absolute value, then the number of steps will be small. The boundary that was hit is then linked to the response given, and the first passage time (i.e., the number of steps taken to reach the boundary) relates to the RT. By convention, a hit at the upper boundary (α) is linked to correct responses and a hit at the lower boundary (0) is

an error.[3] Of prime interest in the modeling of choice reaction times are the proportions with which the absorbing boundaries are hit, as well as the predicted first passage time distribution at either boundary. The two parameters of this model (sometimes also called the *drift diffusion model*) have straightforward interpretations. Boundary separation α relates to the amount of information that is required to make a decision—that is, it indicates the caution level of the decision system (in this case the participant). We will therefore usually let α be different for different persons (but identical across experimental blocks, because we do not expect people to adapt their caution level in the middle of an experimental block). The second parameter, drift rate δ, is the speed of information accumulation. We can easily suppose this to depend both on the participant (who may be more or less able to rapidly process information) and on the item (which may be relatively rich or poor in information content)[4].

Typically two more parameters are introduced to the unbiased drift diffusion model. Firstly, a *bias* parameter to indicate that the starting point of the process may be closer to 0 or to α. This parameter is $\beta \in [0, 1]$, so that the starting value of the process is exactly $\alpha\beta$. Secondly, a shift parameter τ is added to represent RT components that are not part of the decision time (e.g., encoding the stimulus and executing the motor response). The nondecision time is assumed to be stochastically

[3]This convention can be adapted. We could for example say that the upper boundary indicates a category affirmation response and the lower is connected to a category negation. However, the interpretation of the parameters would then change: a high drift rate would no longer evoke a correct response, but rather the category affirmation response, whether correct or not.

[4]In principle, it would also be possible that some participants have a better affinity with some items, resulting in a person-by-item interaction. Such an interaction could be most interesting when, for example, comparing groups of participants with different levels of experience with a certain semantic category (e.g., comparing ichthyologists with laymen in their categorization performance of ***fish***). However, including a person-specific interaction would require more than one replication of each person-by-item combination, which the present data set does not offer.

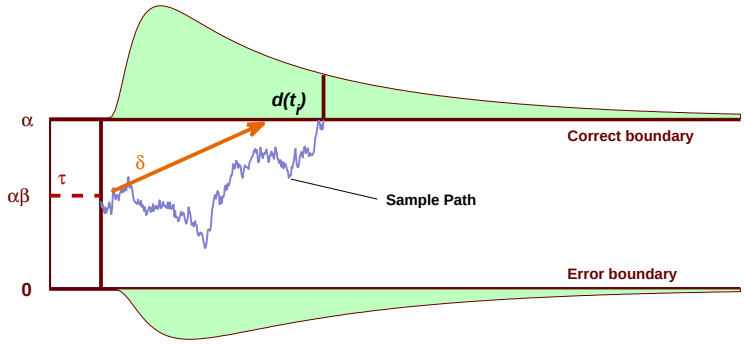


Figure 5.1: *A graphical illustration of the Wiener diffusion model.*

independent from the decision time. The joint probability density of the RT and accuracy (i.e., the Wiener likelihood function, or its probability density function, PDF) is given in Tuerlinckx (2004), and we denote it with $W_{X,T}(x,t|\alpha,\tau,\beta,\delta)$, where the random variables X and T represent the response given and the response latency, respectively. Instances of X and T will be denoted as x and t.

5.4.2 Hierarchical extension

In a hierarchical diffusion model (HDM; Vandekerckhove et al., 2008b), the four parameters that drive the response process are considered random draws from some partly specified distribution (Rouder et al., 2005) that may be subject to many different influences. For example, it may be assumed that the drift rate $\delta_{(i)}$ of the response process at trial i is a random draw from a normal distribution with mean ν and standard deviation η:

$$\delta_{(i)} \sim N\left(\nu, \eta^2\right).$$

The parameters of this distribution can in turn be considered random draws from some higher-order distribution, or they may be seen as some fixed function of other parameters or of data. The multitude of combinations that are possible make the HDM framework an exceedingly flexible method for the analysis of two-choice RT data.

5.4.3 Bayesian implementation

Obtaining parameter estimates in such a flexible context would be quite challenging in general. Finding the maximum-likelihood parameter estimates for a random-effects diffusion model would require repeated computations of a multidimensional integral over the (already nontrivial) Wiener distribution. However, the inclusion of randomly varying parameters and integrating over their distributions is the basic modus operandi in Bayesian statistics. Hence, casting the HDM in a Bayesian statistical framework (bulding upon Vandekerckhove, Tuerlinckx, & Lee, 2008a) allows us to apply the model easily. In the next section, we will specify a specially-tailored HDM, which we will then apply to the Leuven data set.

5.5 Analyzing the Leuven data

5.5.1 Model assumptions

Formally speaking, a statistical model is little more than a set of assumptions regarding structure that is present in the data. We discern five different types of assumptions in the present model, which we describe in turn. We will apply this model to each category separately.

5.5.1.1 The measurement model

At the most basic level, our assumption is that each data point is generated by a diffusion process whose parameters may differ between persons and/or items (i.e., words). We have chosen to allow boundary separation α to differ between persons p, while nondecision time τ and drift rate δ may be different for each item-by-person combination pi. Since we do not want to assume that participants have an a-priori bias for the correct or erroneous responses[5], we fix the bias β to 0.5 for the remainder of this presentation. Formally, the measurement model is written as follows:

$$(t_{(pi)}, x_{(pi)}) \quad \sim \quad W(\alpha_{(p)}, \beta, \tau_{(pi)}, \delta_{(pi)}).$$

This gives the expected distribution of data point $(t_{(pi)}, x_{(pi)})$ (for person p on item i) given all the relevant parameters. W is the Wiener PDF. Note that, as mentioned in an earlier section, we do not let boundary separation α depend on items, so it does not receive an index i.

5.5.1.2 Trial-to-trial variability

Parameters τ and δ are assumed to vary both between persons and between items (and hence from trial to trial). For this random variability, we assume a normal distribution, which is the most common assumption in hierarchical modeling (e.g., De Boeck & Wilson, 2004) and we see no reason to deviate from it here.[6] The normal also serves as a useful first approximation. In both cases, we allow the mean of the trial-to-trial distri-

[5] We can safely assume this, since there were 50% targets and 50% distractors in each block of the experiment.

[6] In principle, one could object that τ cannot follow a normal distribution, as it can never be negative, but in practice the mean (θ) of this distribution has always turned out to be very large compared to its standard deviation (ϕ), so that the mass of this distribution below zero can be safely ignored.

bution to depend on both persons and items. The dependence on persons is simply to allow for interindividual differences (which we believe exist), but the dependence on items is crucial in order to explain interitem differences with the Leuven covariates. Formally:

$$\begin{aligned} \delta_{(pi)} &\sim N(\nu_{(pi)}, \eta^2_{(p)}) \\ \tau_{(pi)} &\sim N(\theta_{(pi)}, \phi^2_{(p)}). \end{aligned}$$

It can be seen that we also allow for the possibility of different trial-to-trial variances between persons.

5.5.1.3 Independent item and person contributions

As explained in the previous section, we want items and persons to have independent effects on two different aspects of the decision process. For the drift rate $\delta_{(pi)}$, we call the item and participant contributions $\lambda_{(i)}$ and $\gamma_{(p)}$, respectively. For the nondecision time $\tau_{(pi)}$ we call them $\psi_{(i)}$ and $\chi_{(p)}$. In both cases, we assume these to be independent and additive (this is a typical construction in psychometrics; for example the Rasch model uses the same assumption; De Boeck & Wilson, 2004):

$$\begin{aligned} \nu_{(pi)} &= \gamma_{(p)} + \lambda_{(i)} \\ \theta_{(pi)} &= \chi_{(p)} + \psi_{(i)}. \end{aligned}$$

5.5.1.4 Population distributions

Since both items and participants were random samples from a larger population, a random effects design is appropriate. Those parameters that have a population distribution are thus assigned population-level parameters. Two distributions over the item population must be defined: that of the item contribution to the drift rate (i.e., $\lambda_{(i)}$) and of the item contri-

bution to the nondecision time (i.e., $\psi_{(i)}$). These components again get normal population distributions:

$$\begin{aligned} \lambda_{(i)} &\sim N(\mu_{\lambda(i)}, \sigma^2_{\lambda(i)}) \\ \psi_{(i)} &\sim N(\mu_{\psi(i)}, \sigma^2_{\psi}). \end{aligned}$$

Note that, since we expect the drift rates for targets and items to be quite different, we also allow their population variances to be different.

For reasons of model identifiability, the mean of one of the random components must be constrained, so we set the mean of $\gamma_{(p)}$ and $\chi_{(p)}$ to 0:

$$\begin{aligned} \gamma_{(p)} &\sim N(0, \sigma^2_{\gamma}) \\ \chi_{(p)} &\sim N(0, \sigma^2_{\chi}). \end{aligned}$$

Finally, we define a population distribution for the boundary separation α:

$$\alpha_{(p)} \sim N(\mu_{\alpha}, \sigma^2_{\alpha}).$$

5.5.1.5 Regression structure

We make most of the previous assumptions in order to account for the possibility of variation between persons or items. Until now, however, the model is strictly descriptive (i.e., it does not include any external covariates that might be employed to explain the variability that is observed). A final set of assumptions pertains to the relationship between the diffusion model parameters and the Leuven data set. Following Hampton (1997), we include five covariates: Typicality (T), Word Length (L), Familiarity (F), Word Frequency (W), and Production Frequency (P). All of these covariates were standardized to have a mean of 0 and a standard deviation of 0.1. As in the regression analysis we showed before, we also add the item's category membership as a predictor (i.e., $C_{(i)} = 1$ if the item was a

target, $C_{(i)} = 0$ if it was a distractor). We call the regression weights ζ for the drift rate and ρ for the nondecision time:

$$\begin{aligned}\mu_{\lambda(i)} &= \zeta_0 + \zeta_1 T_{(i)} + \zeta_2 L_{(i)} + \zeta_3 F_{(i)} \\ &\quad + \zeta_4 W_{(i)} + \zeta_5 P_{(i)} + \zeta_6 C_{(i)} \\ \mu_{\psi(i)} &= \rho_0 + \rho_1 T_{(i)} + \rho_2 L_{(i)} + \rho_3 F_{(i)} \\ &\quad + \rho_4 W_{(i)} + \rho_5 P_{(i)} + \rho_6 C_{(i)}.\end{aligned}$$

In principle, we could do the same for the person contributions to the drift rate or nondecision time, or for the caution parameter α. For example, intelligence might predict the drift rate component (see e.g., Ratcliff, Schmiedek, & McKoon, 2008) or neuroticism might be connected to the caution parameter. Unfortunately, the present data sets do not include person covariates.

5.5.2 Results

The model we have presented is an instance of a hierarchical diffusion model. Software to implement such a model was made available by Vandekerckhove et al. (2008b, "wienereta.odc"). Using this software, we obtained posterior distributions for each of the parameters in the model.[7]

The posterior distributions for ζ, per semantic category, are displayed in Fig. 5.2, and those for ρ are in Fig. 5.3. These posterior inference plots may be read as follows. In each of the subplots, the five horizontal lines represent the posterior distributions of the regression weights of the five Leuven covariates. The lines indicate the Bayesian credibility interval (BCI): the region around the mean that contains 95% of the mass of the parameter's posterior distribution. The diamonds indicate the posterior

[7] We followed the recommendations made by Vandekerckhove et al. (2008b) to check for convergence issues and found that there were none (all convergence statistics $\hat{R} < 1.05$, all chains show proper mixing).

Table 5.2: *Regression weights in the HDM. The signs of the regression weights whose 95% credibility intervals do not contain 0 are displayed, others are replaced by a dot.*

	ζ					ρ				
	T	L	F	W	P	T	L	F	W	P
birds	+	−	·	−	·	·	·	·	·	·
fish	·	·	−	·	+	·	·	·	·	·
insects	+	·	·	·	·	·	+	·	·	·
mammals	+	+	·	·	·	·	+	·	·	·
musical instruments	·	·	·	·	·	·	·	·	·	·
reptiles	+	−	·	·	·	·	+	·	·	·
tools	+	·	·	·	·	·	+	·	·	·
vehicles	+	·	·	+	−	·	+	·	−	·

means. The vertical line is the value 0. In these figures, two patterns emerge quite clearly: the effect of Typicality (T) on drift rate is always positive, and most of those BCIs do not include 0. Similarly, in Fig. 5.3, Word Length (L) generally has a positive effect on nondecision time.

To compare these results to the ones obtained from the standard analysis in Table 5.1, we constructed a similar table for these two sets of regression weights. We display the sign of a regression weight if its 95% credibility interval does not contain 0 (i.e., with 95% probability the parameter is not 0). In contrast with the classical analysis, results here are predominantly consistent—for the drift rate regression, only Typicality consistently shows up as a good predictor. For the nondecision time, Word Length has a consistent influence. In both cases, the sign of the regression weight is as expected.[8]

Fig. 5.4 shows the relationship between the Typicality score of an item

[8]The present analysis is based on a multiple regression. In one alternative attempt, we restricted ourselves to univariate regressions (i.e., including one covariate at a time), and obtained comparable results.

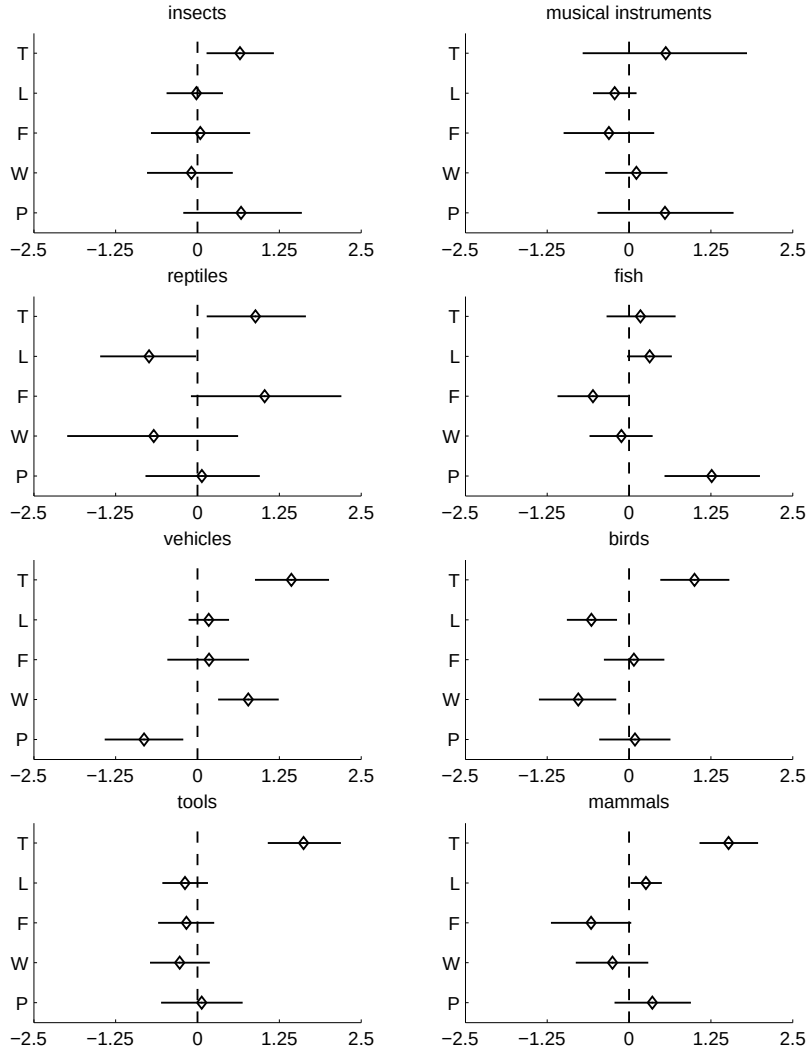


Figure 5.2: *Posterior inference plots for the regression weights ζ (the regression weights for the λs, the item contributions to the drift rates). See text for details.*

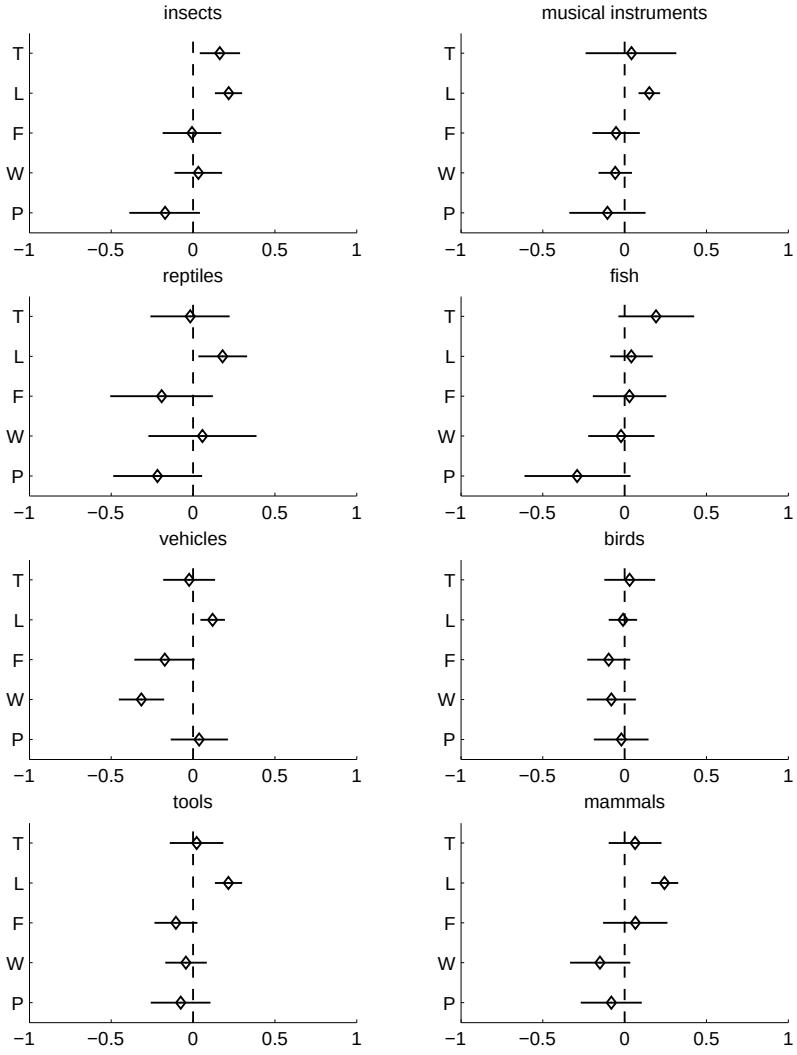


Figure 5.3: *Posterior inference plots for the regression weights ρ (the regression weights for the ψs, the item contributions to the nondecision times). See text for details.*

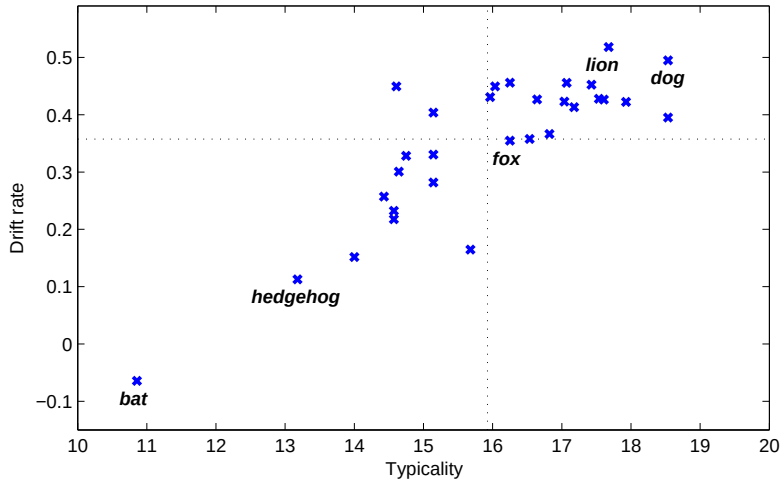


Figure 5.4: *An example regression result. Drift rate on average increases with increasing Typicality. Item bat is a clear outlier on both dimensions. The dotted lines indicate the mean Typicality and mean drift rate. To avoid confusion: the values on the vertical axis are the* total *drift rates assuming an average person with* $\gamma_{(p)} = 0$.

and its contribution to the drift rate (depicted for an average participant; i.e. $\gamma_{(p)} = 0$) in the category ***mammals***. A somewhat linear relationship is evident[9], and we have labeled some of the items on the graph. Item *bat* has the lowest Typicality rating, and also the lowest drift rate. Items *dog* and *lion* reside on the opposite side of the spectrum.

While some of these effects are very easy to interpret, others are less intuitive. In Fig. 5.5, we display the effect that drift rate has on the raw data. We selected three items from the range of Typicality ratings (from the ***mammals*** category) and display the expected distribution of their (correct) RTs and their expected accuracy scores.

[9]The linear relationship is clearly not perfect, and perhaps even better prediction could have been achieved with a non-linear regression, but we do not explore that avenue here.

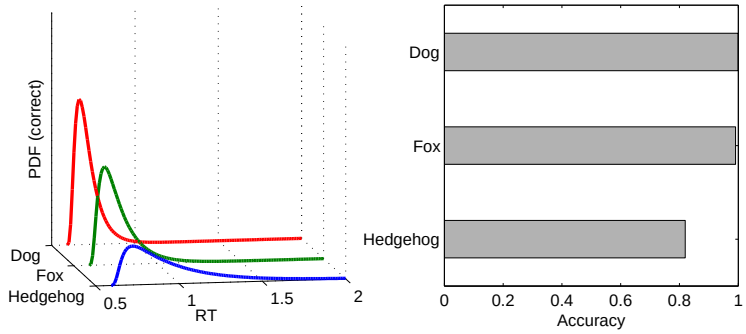


Figure 5.5: *The effect of higher Typicality (and hence higher drift rates) on the raw data. RT distributions become more compact and less skewed (smaller mean, smaller variance) as Typicality (drift rate) increases. Accuracy increases with higher Typicality. The PDFs are marginal PDFs (i.e., conditional upon a correct answer) and have been normalized so that they integrate to* 1*).*

The interindividual variability is also notable. In particular, the person-specific α parameter that represents a person's caution shows much variation. Fig. 5.6 shows the effect of different boundary separations (keeping all other factors constant). We selected three participants from the population (corresponding to the 10^{th}, 50^{th}, and 90^{th} percentiles) and plotted their expected raw RT distributions and accuracies (for an average item). The range of α values in the population has a small but noticeable effect on both the RT distribution and the accuracy scores.

The effect of Word Length is, from a research methods point of view, perhaps the most important to keep in mind (we will elaborate on why we believe this is so in the Discussion section below). Fig. 5.7 shows the relationship between Word Length and nondecision time for the category ***tools*** (here, too, the effect might be better captured by a non-linear regression). The nondecision times associated with particular items range from 500ms to 630ms—the interquartile range is more than 60ms. A graphical presentation for this effect (like the ones in Figs. 5.5 and 5.6) would show

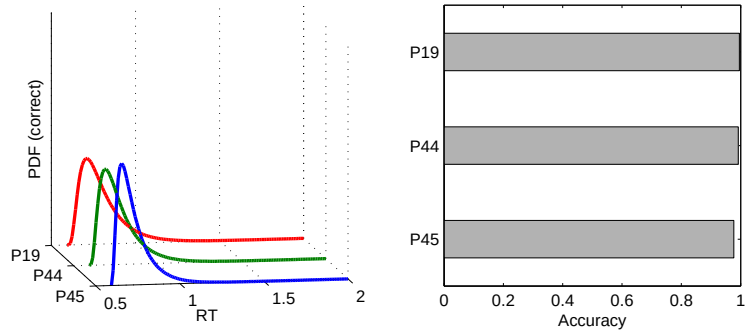


Figure 5.6: *The effect of higher boundary separation on the raw data. P19 has a high α, P44 has a median value, and P45 has a low α. RT distributions become more skewed with increasing α, but accuracy increases. The PDFs are marginal PDFs (i.e., conditional upon a correct answer) and have been normalized so that they integrate to* 1*).*

identical accuracy scores and identically shaped RT distributions, but shifted to the right for items with higher values of $\psi_{(i)}$. For interpretation, we can compute that, on average, adding one letter to a word shifts the RT by 7–12ms, depending on the category.[10]

Finally, we can compare differences between participants with differences between items. Table 5.3 shows population standard deviations from the HDM (the values shown are the means of the posterior distributions of the parameters). Comparing the drift rate's variability due to persons (σ_γ) with its variability due to item differences ($\sigma_{\lambda(1)}$ for targets[11], $\sigma_{\lambda(2)}$ for distractors), we can see that, with the exception of the category ***fish***, the item variance is always much larger than the person variance. The reverse is true for the nondecision time: residual item variance there is much smaller than the variance in the person population.

[10] This is in line with results from Hampton (1997).

[11] Note that this is the *residual* item variance, after correcting for all the covariates.

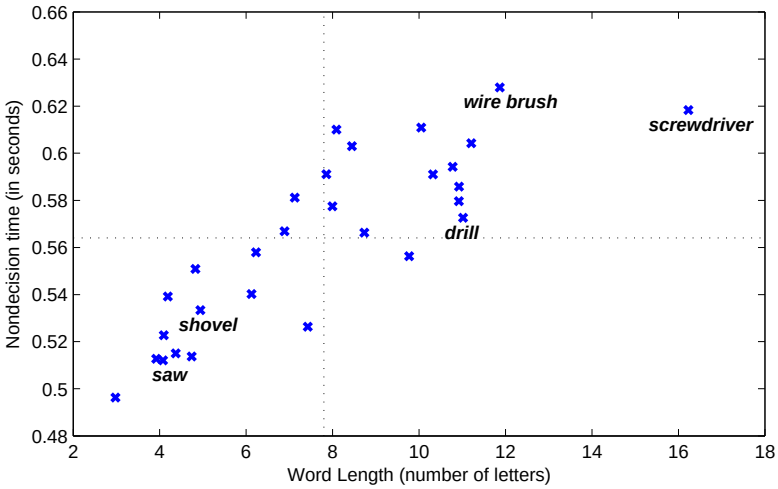


Figure 5.7: *Nondecision time in the category* ***tools*** *on average increases with increasing Word Length. The dotted lines indicate the mean Word Length and mean nondecision time. The values on the vertical axis are the total nondecision times (assuming an average person with* $\chi_{(p)} = 0$*). Word Length has been jittered to avoid overlapping symbols. The original Dutch versions of the labeled items were (from left to right): zaag, schop, boormachine, staalborstel, and schroevendraaier.*

5.6 Discussion

The theoretical advantages of using a process model on the one hand and a hierarchical model on the other (together leading to a cognitive psychometric model) were extensively described in the introduction. However, the demonstration in the present article also shows the practical applicability of this method.

We believe that, as a methodological advance, the HDM framework (Vandekerckhove et al., 2008b) can contribute not only to semantic categorization studies, but to a more general class of paradigms. If speeded binary choice RTs are collected, and if it is likely that there are interindividual (or interitem) differences, then the HDM framework might prove

Table 5.3: *Population variability parameters in the HDM. We can compare the person variabilities with the item variabilities (see text for details). All values are standard deviations.*

	boundary separation	nondecision time		drift rate		
	σ_α	σ_χ	σ_ψ	σ_γ	$\sigma_{\lambda(1)}$	$\sigma_{\lambda(2)}$
birds	0.018	0.067	0.023	0.025	0.077	0.115
fish	0.013	0.084	0.035	0.122	0.057	0.176
insects	0.022	0.064	0.024	0.018	0.113	0.229
mammals	0.024	0.067	0.023	0.019	0.069	0.085
musical instruments	0.021	0.058	0.020	0.093	0.091	0.157
reptiles	0.024	0.091	0.029	0.022	0.106	0.247
tools	0.015	0.081	0.020	0.021	0.078	0.148
vehicles	0.020	0.056	0.026	0.025	0.093	0.164

useful.

In the introduction, we have also referred to Estes' (1956, 2000) view on individual differences and how averaging over participants (or items) may lead to averaging artifacts. Hierarchical modeling deals with this issue in a practical and efficient way. In the domain of choice RTs, a different type of artifacts may occur if variability in the various facets of the response process are ignored. In the particular case of the HDM, variability in the nondecision process time (i.e., encoding and processing time) can easily be confused for variability in the decision process time. Indeed, past analyses of semantic categorization data have found effects of word length on RT, but the present analysis strongly suggests this to be an artifact—word length does not predict the information uptake rate, but rather the encoding time of the process. However, accounting for this variability in nondecision time is important to achieve proper parameter estimates.

5.6.1 Implications for semantic categorization studies: item properties

The model of speeded semantic categorization we have introduced is very explicit about the various stages involved in making a category membership decision towards a visually presented verbal stimulus. We believe that we have convincingly shown that elaboration of the aspects involved in arriving upon that decision is a useful practice. By attributing the effects of typicality and word length to different aspects of the response process, we have moved beyond the common practice of regressing these covariates upon the observed RTs. The very nature of the latter approach confines it to the mere establishment of the relative effect of both covariates upon RT. The former approach allows the effect to be attributed to specific components of the RT.

The critical reader might raise the objection that we have not been explicit enough in our account of the categorization behavior, and might point out that for those among us who are interested in understanding semantic cognition the question "what governs semantic categorization time differences" has shifted towards "what governs information uptake differences".

In response to this objection we readily admit that, indeed, we have been less than explicit about the representation upon which the accumulator process acts. We have not committed ourselves, for instance, to featural representations of the kind Smith et al. (1974) or McCloskey and Glucksberg (1979) have argued for. Nor did we attempt to attempt to link the accumulation process to the semantic markers that were proposed by Glass and Holyoak (1974). Although the terminology we have used throughout this manuscript (e.g., information uptake, accumulation of evidence) might tempt the reader into thinking that the diffusion model is more in favor of a successive comparison of exemplar and category features than of an or-

dered search through semantic markers, we do not necessarily believe this to be the case. Any representational format that allows for a stochastic accumulation of evidence for or against the endorsement of an item as a category member is in principle compatible with the diffusion model we propose. This does not imply that the methodology we have proposed in this manuscript can not be brought to bear upon the representational issue. In much the same way as we have explored the relative contributions of different covariates to the degree of information uptake, one could evaluate the predictions of rivalling representations, providing that they are explicit enough to warrant quantification. One might consider using the Leuven data again for such endeavors as they include plenty of information on the intension and extension of semantic categories.

For now, however, we dare suggest that casting speeded semantic categorization decisions in terms of a diffusion model constitutes sufficient explicitation. As we have pointed out in the Introduction, much of the efforts during the last three decades have been aimed at disentangling the various constructs that are likely to influence semantic categorization. As it is along the lines of these constructs that theories of semantic behavior are likely to develop, tools that shed light on the varying manners in which they exert their influence are valuable. At the very least, we hope to have shown that the diffusion model is able to accomplish this.

In the near future we hope the model will allow us to study the effects of variables that are present in the Leuven data, but were not incorporated in the current analyses for reasons of brevity. The questions of whether age of acquisition exerts an effect in semantic categorization, and how that effect might come about, for instance, deserve some attention as they have generated considerable debate (Brysbaert, Van Wijnendaele, & De Deyne, 2000; De Deyne & Storms, 2007; Morrison, Ellis, & Quinlan, 1992; Morrison & Gibbons, 2006). We also hope to study the impact category do-

minance has on the categorization performance participants display. This will require the collection of additional data as the Leuven norms do not include a direct measure of the association strength between an item and its superordinate(s). (See De Deyne & Storms, 2008 for a discussion of the differences between the direct or constrained measures of category dominance that are mostly used in the semantic categorization literature and the unconstrained measure that can be found in the Leuven norms). These and other investigations will undoubtedly benefit from experimental manipulations that are supposed to influence the effect a particular covariate has on the distribution of one of the model's parameters, but not on that of others (Hampton, 1997).

5.6.2 Implications for semantic categorization studies: person properties

In the Results section we already indicated that our analyses demonstrated evidence of interindividual variability in semantic categorization behavior. Namely, the person-specific α that represents a person's caution showed considerable variation with accompanying effects on the degree to which true category exemplars were endorsed as such (see Fig. 5.6 for a demonstration). These differences between persons reflect (more or less) imprudent task strategies resulting in (more or fewer) erroneous decisions. It has been demonstrated a number of times (e.g., Hampton, 1998, 2007; McCloskey & Glucksberg, 1978) that people disagree considerably about the items they consider to be true members of a semantic category. The degree to which people disagree is likely to be reflected in the variation of the α parameter.

As De Deyne (2008) did not record any information on the students participating in the semantic categorization task but their age and gender, our assumptions regarding the person side of the diffusion model have

remained strictly descriptive (i.e., no external covariates that might be employed to explain the interindividual variability that was observed were available). Looking at recent applications of the speeded semantic categorization task, in which the decisions of individuals with autism were compared with those of matched controls (Gastgeb, Strauss, & Minshew, 2006) or the differences in categorization behavior by Broca's and Wernicke's aphasic individuals were investigated (Kiran & Thompson, 2003), it seems that the approach argued for in this manuscript can also prove to be valuable when applied to person properties instead of item properties. One can image proposing a diffusion model of speeded semantic categorization in which person variables are regressed upon the model's parameters or a model whose parameter distributions are allowed to differ from one group to another. Along these lines we have begun to compare the categorization behavior of individuals displaying many schizotypal traits to that of individuals who display few schizotypal traits. The difference in the degree to which individuals in the general population display these traits is thought to accompany their willingness to endorse weak semantic associates as true category members (Kiang & Kutas, 2005, 2006). Hence, we would expect that in the diffusion model analysis participants scoring high on schizotypy would demonstrate a greater bias β towards the target than participants who obtained a low score.

References

Baayen, R. H., Piepenbrock, R., & van Rijn, H. (1993). *The CELEX lexical database [CD-ROM].* (Philadelphia: University of Pennsylvania, Linguistic Data Consortium.)

Batchelder, W. H. (2007, October). *Cognitive psychometrics: Combi-*

ning two psychological traditions. CSCA Lecture, Amsterdam, The Netherlands.

Batchelder, W. H., & Riefer, D. M. (1999). Theoretical and empirical review of multinomial process tree modeling. *Psychonomic Bulletin & Review*, *6*, 57–86.

Brysbaert, M., Van Wijnendaele, I., & De Deyne, S. (2000). Age-of-acquisition effects in semantic processing tasks. *Acta Psychologica*, *104*, 215–226.

Casey, P. J. (1992). A reexamination of the roles of typicality and category dominance in verifying category membership. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *18*, 823–834.

Chumbley, J. I. (1986). The roles of typicality, instance dominance, and category dominance in verifying category membership. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *12*, 257–267.

Clark, H. (1973). The language-as-fixed-effect fallacy: A critique of language statistics in psychological research. *Learning and Verbal Behavior*, *12*, 335–359.

Coleman, E. B. (1964). Generalizing to a language population. *Psychological Reports*, *14*, 219–222.

Collins, A. M., & Quillian, M. R. (1970). Does category size affect categorization time? *Journal of Verbal Learning and Verbal Behavior*, *8*, 240–247.

De Boeck, P., & Wilson, M. (2004). *Explanatory item response models: A generalized linear and nonlinear approach.* New York: Springer.

De Deyne, S. (2008). *Proximity in semantic vector space.* Unpublished doctoral dissertation, University of Leuven.

De Deyne, S., & Storms, G. (2007). Age-of-acquisition differences in young and older adults affect latencies in lexical decision and semantic ca-

tegorization. *Acta Psychologica*, *124*, 274–295.

De Deyne, S., & Storms, G. (2008). Word associations: Norms for 1,424 Dutch words in a continuous task. *Behavior Research Methods*, *40*, 198–205.

De Deyne, S., Verheyen, S., Ameel, E., Vanpaemel, W., Dry, M., Voorspoels, W., et al. (2008). Exemplar by feature applicability matrices and other Dutch normative data for semantic concepts. *Behavior Research Methods*, *40*, 1030-1048.

Estes, W. K. (1956). The problem of inference from curves based on group data. *Psychological Bulletin*, *53*, 134–140.

Estes, W. K. (2002). Traps in the route to models of memory and decision. *Psychonomic Bulletin & Review*, *9*, 3–25.

Feller, W. (1970). *An introduction to probability theory and its applications: Vol. I.* New York: John Wiley & Sons.

Gastgeb, H. Z., Strauss, M. S., & Minshew, N. J. (2006). Do individuals with autism process categories differently? The effect of typicality and development. *Child Development*, *77*, 1717–1729.

Glass, A. L., & Holyoak, K. J. (1974). Alternative conceptions of semantic memory. *Cognition*, *3*, 313–339.

Glass, A. L., & Meany, P. J. (1978). Evidence for two kinds of low-typical instances in a categorization task. *Memory & Cognition*, *6*, 622–628.

Hampton, J. A. (1997). Associative and similarity-based processes in categorization decisions. *Memory & Cognition*, *25*, 625–640.

Hampton, J. A. (1998). Similarity-based categorization and fuzziness of natural categories. *Cognition*, *65*, 137–165.

Hampton, J. A. (2007). Typicality, graded membership, and vagueness. *Cognitive Science*, *31*, 355–384.

Hampton, J. A., & Gardiner, M. M. (1983). Measures of internal category structure: A correlational analysis of normative data. *British Journal*

of Psychology, *74*, 491–516.

Heathcote, A., Brown, S., & Mewhort, D. J. K. (2000). The power law repealed: The case for an exponential law of practice. *Psychonomic Bulletin & Review*, *7*, 185–207.

Heathcote, A., Popiel, S. J., & Mewhort, D. J. K. (1991). Analysis of response time distributions: An example using the Stroop task. *Psychological Bulletin*, *109*, 340–347.

Kiang, M., & Kutas, M. (2005). Association of schizotypy with semantic processing differences: An event-related brain potential study. *Schizophrenia Research*, *77*, 329–342.

Kiang, M., & Kutas, M. (2006). Abnormal typicality of responses on a category fluency task in schizotypy. *Psychiatry Research*, *145*, 119–126.

Kintsch, W. (1980). Semantic memory: A tutorial. In R. S. Nickerson (Ed.), *Attention and performance VIII* (pp. 595–620). Hillsdale, NJ: Erlbaum.

Kiran, S., & Thompson, C. K. (2003). Effect of typicality on online category verification of animate category exemplars in aphasia. *Brain and Language*, *85*, 441–450.

Lamberts, K. (2000). Information-accumulation theory of speeded categorization. *Psychological Review*, *107*, 227–260.

Laming, D. R. J. (1968). *Information theory of choice-reaction times*. London, UK: Academic Press.

Landauer, T. K., & Freedman, J. L. (1968). Information retrieval from long-term memory: Category size and recognition time. *Journal of Verbal Learning and Verbal Behavior*, *7*, 291–295.

Larochelle, S., & Pineau, H. (1994). Determinants of response times in the semantic verification task. *Journal of Memory and Language*, *33*, 796–823.

Larochelle, S., Richard, S., & Soulires, I. (2000). What some effects might not be : The time to verify membership in "well-defined" categories. *The Quarterly Journal of Experimental Psychology*, *53A*, 929–961.

Link, S. W., & Heath, R. A. (1975). A sequential theory of psychological discrimination. *Psychometrika*, *40*, 77–105.

Loftus, E. F. (1973). Category dominance, instance dominance, and categorization time. *Journal of Experimental Psychology*, *97*, 70–74.

McCloskey, M. E. (1980). The stimulus familiarity problem in semantic memory research. *Journal of Verbal Learning and Verbal Behavior*, *19*, 485–502.

McCloskey, M. E., & Glucksberg, S. (1978). Natural categories: Well defined or fuzzy sets? *Memory & Cognition*, *6*, 462–472.

McCloskey, M. E., & Glucksberg, S. (1979). Decision processes in verifying category membership statements: Implications for models of semantic memory. *Cognitive Psychology*, *11*, 1–37.

Morrison, C. M., Ellis, A. W., & Quinlan, P. (1992). Age of acquisition, not word frequency affects object naming, not object recognition. *Memory & Cognition*, *20*, 705–714.

Morrison, C. M., & Gibbons, Z. C. (2006). Lexical determinants of semantic processing speed. *Visual Cognition*, *13*, 949–967.

Nosofsky, R. M., & Palmeri, T. J. (1997). An exemplar-based random walk model of speeded classification. *Psychological Review*, *104*, 266–300.

Ratcliff, R. (1978). A theory of memory retrieval. *Psychological Review*, *85*, 59–108.

Ratcliff, R. (1979). Group reaction time distributions and an analysis of distribution statistics. *Psychological Bulletin*, *86*, 446–461.

Ratcliff, R., Schmiedek, F., & McKoon, G. (2008). A diffusion model explanation of the worst performance rule for reaction time and IQ. *Intelligence*, *36*, 10–17.

Ratcliff, R., & Tuerlinckx, F. (2002). Estimating parameters of the diffusion model: Approaches to dealing with contaminant reaction times and parameter variability. *Psychonomic Bulletin & Review*, *9*, 438–481.

Rips, L. J., Shoben, E. J., & Smith, E. E. (1973). Semantic distance and the verification of semantic relations. *Journal of Verbal Learning and Verbal Behavior*, *12*, 1–20.

Rosch, E. (1973). On the internal structure of perceptual and semantic categories. In T. E. Moore (Ed.), *Cognitive development and the acquisition of language* (pp. 111–144). New York: Academic Press.

Rouder, J. N., & Lu, J. (2005). An introduction to Bayesian hierarchical models with an application in the theory of signal detection. *Psychonomic Bulletin & Review*, *12*, 573-604.

Rouder, J. N., Lu, J., Speckman, P. L., Sun, D., & Jiang, Y. (2005). A hierarchical model for estimating response time distributions. *Psychonomic Bulletin & Review*, *12*, 195-223.

Rouder, J. N., Lu, J., Sun, D., Speckman, P., Morey, R., & Naveh-Benjamin, M. (2007). Signal detection models with random participant and item effects. *Psychometrika*, *72*(4), 621–642.

Rouder, J. N., & Speckman, P. L. (2004). An evaluation of the Vincentizing method of forming group-level response time distributions. *Psychonomic Bulletin & Review*, *11*, 419-427.

Ruts, W., De Deyne, S., Ameel, E., Vanpaemel, W., Verbeemen, S., & Storms, G. (2004). Dutch norm data for 13 semantic categories and 338 exemplars. *Behavior Research Methods, Instruments, & Computers*, *36*, 506–515.

Smith, E. E., Shoben, E. J., & Rips, L. J. (1974). Structure and process in semantic memory: A featural model for semantic decisions. *Psychological Review*, *81*, 214–241.

Thomas, E. A. C., & Ross, B. H. (1980). On appropriate procedures for combining probability distributions within the same family. *Journal of Mathematical Psychology*, *21*, 136–152.

Tuerlinckx, F. (2004). The efficient computation of the distribution function of the diffusion process. *Behavior Research Methods, Instruments, & Computers*, *36*, 702-716.

Vandekerckhove, J., & Tuerlinckx, F. (2007). Fitting the Ratcliff diffusion model to experimental data. *Psychonomic Bulletin & Review*, *14*, 1011–1026.

Vandekerckhove, J., & Tuerlinckx, F. (2008). Diffusion model analysis with MATLAB: A DMAT primer. *Behavior Research Methods*, *40*, 61–72.

Vandekerckhove, J., Tuerlinckx, F., & Lee, M. D. (2008a). A Bayesian approach to diffusion process models of decision-making. In V. Sloutsky, B. Love, & K. McRae (Eds.), *Proceedings of the 30th annual conference of the cognitive science society* (pp. 1429–1434). Austin, TX: Cognitive Science Society.

Vandekerckhove, J., Tuerlinckx, F., & Lee, M. D. (2008b). *Hierarchical diffusion models for two-choice response times.* (Manuscript submitted for publication)

Voss, A., & Voss, J. (2007). Fast-dm: A free program for efficient diffusion model analysis. *Behavior Research Methods*, *39*, 767–782.

Wagenmakers, E.-J. (in press). Methodological and empirical developments for the Ratcliff diffusion model of response times and accuracy. *European Journal of Cognitive Psychology*.

Wilkins, A. J. (1971). Conjoint frequency, category size, and categorization time. *Journal of Verbal Learning and Verbal Behavior*, *10*, 382–385.