Reinforcement learning algorithms to model learning and decision-making in individuals with depressive disorders

by

Lizelle Niit (NTXLIZ001)

Dissertation presented for the degree of Master of Science

in the Department of Mathematics and Applied Mathematics,

University of Cape Town

December 2021

Supervisor: A/Prof. Jonathan Shock

To my parents

Abstract

Mental illness causes enormous suffering for many people. Current treatments do not reliably alleviate that suffering. Unclear conceptualisations of mental disorders combined with little knowledge about their aetiology are roadblocks to developing better treatments. This dissertation reviews attempts to use reinforcement learning models to improve the way we conceptualise some of the processes happening in the brain in mental illness. The hope is that more clearly defining the problems we are dealing with will eventually have a positive impact on our ability to diagnose and treat them.

I start by giving an overview of the reinforcement learning framework, and detail some of the reinforcement learning models that have been used to understand mental illness better. I explain the statistical techniques used to compare these models and to estimate parameters once a model has been chosen. This leads in to a survey of what researchers have learned about human behaviour using these techniques. I focus particularly on results related to depression. I argue that key parameters like learning rate and reward sensitivity are closely linked to depressive symptoms. Finally, I speculate about the impact that knowledge of this kind may have on the development of better diagnosis and treatment for mental illness in general and depression specifically.

Contents

Abstract ii

Acronyms v

Notation vi

Preliminary notes vii
- Illustrations . . . vii
- Scripts on GitHub . . . vii
- Notes on terminology . . . vii

Acknowledgements viii

1 Introduction 1
- 1.1 Overview . . . 1
- 1.2 Burden of mental illness on society . . . 3
- 1.3 The subjective burden of mental illness . . . 3
- 1.4 Inadequacy of current treatment of depressive disorders . . . 4
- 1.5 Computational psychiatry . . . 6
- 1.6 Goals of this project . . . 7

2 Reinforcement learning in animals 9
- 2.1 What is reinforcement learning? . . . 10
- 2.2 Reinforcement learning vocabulary . . . 11
- 2.3 Interaction between the agent and the environment through actions and rewards . . . 13
- 2.4 Gridworld . . . 14
- 2.5 Model-free vs. model-based learning and decision-making . . . 15
- 2.6 Instrumental (or operant) vs. classical (or Pavlovian) conditioning . . . 16
- 2.7 Rescorla Wagner . . . 17
- 2.8 Deciding what actions to take . . . 19
- 2.9 Variations of Rescorla Wagner . . . 19
- 2.10 Temporal difference (TD) learning . . . 24

2.11 TD learning to predict reward in classical conditioning 27
2.12 Why use RW and not TD? 31
2.13 Why Q-learning sometimes looks like Rescorla Wagner 32

3 Parameter estimation and model selection 34
3.1 Overview of the process 35
3.2 Parameter estimation 38
3.3 Finding priors 40
3.4 Model comparison 45

4 Findings from fitting reinforcement models to behaviour 48
4.1 Introduction to depression and anhedonia 49
4.2 Pavlovian bias and Pavlovian-instrumental transfer 51
4.3 Findings on learning rate and reward/punishment sensitivity 63

5 Discussion: using behavioural findings to improve classification and treatment of mental disorders 70
5.1 A deeper look at what depression is 71
5.2 Current classification of mental disorders 73
5.3 Improving classification with latent variables 75
5.4 A practical suggestion to assist with diagnosis and possibly treatment 76
5.5 The importance of subjectivity 77
5.6 Ethical implications 79

6 Conclusions 81
6.1 Summary 81
6.2 Going back to the research questions 82
6.3 Limitations and suggestions for future research 83
6.4 Final thoughts 84

A Derivation of expectation-maximisation update equations 86
A.1 Getting to the EM starting equation 86
A.2 Deriving the update equations for the mean and variance of the prior 88

B Relationship between the Hessian and covariance matrix of a Gaussian 97

C Eligibility traces 99

Bibliography 100

Acronyms

CS conditioned stimulus.

DALY disability-adjusted life-year.

DSM-5 Diagnositic and Statistical Manual of Mental Disorders, fifth edition.

EM expectation-maximisation.

MASQ Mood and Anxiety Symptom Questionnaire.

MDD major depressive disorder.

PIT Pavlovian-instrumental transfer.

RDoC research domain criteria.

RW Rescorla Wagner.

SSRI selective serotonin reuptake inhibitor.

TD temporal difference.

Notation

Following Sutton and Barto (2020), I distinguish between a general representation of a state/action at time t and a specific state/action.

$Q(s_t, a_t)$	the value of taking action a_t when in state s_t at time t
$V_t(s_t)$	the value of state s_t at time t
s_t	the state at time t (or trial t)
a_t	the action taken at time t (or trial t)
r_{t+1}	the reward/punishment received after taking an action from state s_t
S	a specific state
S'	the specific state following state S
A	a specific action
R	a specific reward
δ	reward prediction error
ρ	reward sensitivity
α	learning rate
b	fixed bias
κ	Pavlovian bias
ξ	a parameter influencing action choices, referred to as 'noise' or 'lapse'
$\mathbf{x}$	a stimulus representation vector with one component for every salient feature of the state
$\mathbf{w}$	a vector containing the weights associated with stimuli, its length equal to the number of components in the stimulus representation vector $\mathbf{x}$

Preliminary notes

Illustrations

Unless stated otherwise, all illustrations in this dissertation are my own.

Scripts on GitHub

Python code for understanding some results of the studies discussed in this dissertation can be found at https://github.com/lizelleniit/lizellemasters.

Notes on terminology

- When I refer to 'animals' in the text, I am including humans.
- 'Go' refers to choosing to act, either by approaching or withdrawing from a stimuli.
- 'Nogo' refers to choosing not to act; that is, choosing neither to approach nor withdraw.
- The valence of a stimulus refers to the degree of attractiveness of that stimulus.

Acknowledgements

For a long time, things weren't looking good for this project. That it eventually got done is a testimony to the extraordinary love and persistent encouragement of my mentors, family and friends.

My supervisor, Jonathan Shock, was unfathomably kind and understanding and supported me through long periods of sub-optimal behaviour on my part. He has a remarkable ability to help me get unstuck and think through problems in a helpful way. He is both a wonderful supervisor and a wonderful human.

My friends and mentors Rose and Richard Grant let me stay with them for extended periods and created a safe, caring space where I could both grow as a person and be productive. Rose, who does educational consulting, worked with me for many hours and transformed the way I thought and felt about my work. The change has been magical. Richard has been an incredibly kind, generous, dependable presence in my life. He is the best outdoor adventure buddy I could ask for, and my adventures with him have helped restore and protect my mental health. He has given me a consistent stream of encouragement that has sustained me, and he bakes the best bread ever.

My friend and research helper Robyn Crowhurst resurrected my enthusiasm with her own at a time when my motivation was extremely low. She also drew my attention to many of the papers I cited in this dissertation.

My friend Daniel Adamiak is caring and consistent and insightful, and has a remarkable knack for making me feel both radically accepted and motivated to improve. He worked with me through difficult maths and statistics and patiently explained concepts as many times as I needed. He did the derivation in Appendix A with me, coming up with several insights I would not have come up with by myself.

I'd like to thank my friend William Grunow for helping with Appendices A and B and for faithfully spreading the goo.

I thank the South African National Research Foundation (NRF) for two years of generous funding, which relieved a huge amount of financial pressure.

Finally, I thank my parents. All my life, they have loved me intensely and given me everything and more than they had to give. They taught me to play, climb things, bake cakes, read books, take things apart, and solve problems. Most importantly, they taught me to care. I love them so much more than I know how to express.

Chapter 1

Introduction

1.1 Overview

Reinforcement learning has become a key tool within the framework of artificial intelligence for modelling the behaviour of agents in the real world, as well as creating artificial agents which can perform certain tasks to superhuman levels (Mnih et al., 2013; Silver et al., 2016). This dissertation reviews applications of reinforcement learning models to better understand and classify mental disorders. It addresses three broad areas of enquiry. Firstly, it addresses the question of how we can go about fitting reinforcement learning models to data. Secondly, it reviews findings from studies which have done this, particularly focusing on the question of what such studies have taught us about differences in reinforcement learning parameters between people with and without depressive disorders. Lastly, it explores how we might use such differences to describe mental health difficulties in alternative ways to current classification systems.

To address these three questions, I have needed to bring together several fields of knowledge: machine learning, Bayesian data analysis, behavioural psychology, and psychiatry. We can perform behavioural psychological experiments to learn more about the patterns in people's behaviour. While it is easy to describe simple aspects of that behaviour (such as the number of correct decisions or reaction times), it is challenging to describe more complex patterns in behaviour, such as the way improvement takes place over time. Reinforcement learning models enable us to quantify these more complex patterns in a succinct way by reducing them to a small number of latent variables. In order to make use of prior information, we use Bayesian data analysis techniques to decide which model and which parameter values fit the data from a given

experiment the best. Analysing behavioural patterns in this way allows us to make inferences about the reinforcement learning processes we think are happening in the brain. We can analyse data from both healthy people and people with mental illness, and we can explore the differences in the parameters we obtain by fitting reinforcement learning models to the data. I claim that specifying such differences among people might form a basis for new ways of defining and classifying the unhealthy processes occurring in mental illness. To set the stage for discussing the above topics, in this introduction I briefly explain what reinforcement learning is, what I mean by depressive disorders and major depressive disorder (MDD), and where this dissertation fits into the broader field of computational psychiatry. Firstly, let me give a very brief introduction to reinforcement learning, although a more detailed overview will be given in chapter 2.

Reinforcement learning involves using patterns of rewards and punishments received in various situations to inform behaviour. A reinforcement learning agent can be either biologically based, such as an animal, or artificially created, such as a computer program. A reinforcement learning agent interacts with its environment by taking actions, and the environment responds with feedback in the form of rewards or punishments. Decisions about how to behave are made based on the values the agent attaches to different states and actions. A reinforcement learning model consists of two components: firstly, a rule for updating values, and, secondly, a rule for choosing actions based on current values. These models have free parameters that are varied to fit the data as well as possible. This dissertation focuses on human agents. Reinforcement learning research on humans has focused both on the behaviour of human agents and the timing and intensity of neural firing. I will focus primarily on behaviour, but touch on neural firing as well.

This dissertation reviews literature examining how reinforcement learning parameters differ among groups of people, in particular people with and without mental illness. To keep the scope of the dissertation manageable, when it comes to mental illness I will focus mostly on what the Diagnositic and Statistical Manual of Mental Disorders, fifth edition (DSM-5) refers to as depressive disorders. The depressive disorders include, among others, major depressive disorder and dysthymia[1] (American Psychiatric Association, 2013, p.155). What these disorders have in common is a depressed mood, which may take the form of sadness, emptiness, hopelessness, or irritability. MDD is the depressive disorder most frequently mentioned in the papers cited in this dissertation. The DSM-5 highlights two key symptoms of major depressive disorder, one of which must be present for a diagnosis. These key symptoms are depressed (sad, empty, hopeless or

[1]Dysthymia is characterised by depression symptoms persisting for at least two years (American Psychiatric Association, 2013, p.168).

irritable) mood and anhedonia (reduced interest or pleasure in most activities). Other symptoms include disturbances in appetite and sleep, inability to concentrate, feelings of worthlessness, and recurrent thoughts of suicide (American Psychiatric Association, 2013, p.160-161). [2]

1.2 Burden of mental illness on society

Mental illness in general and depressive disorders in particular place a huge burden on society. The 2019 Global Burden of Disease study (Institute for Health Metrics and Evaluation, 2021) estimates that mental illness accounts for 4.92% of disability-adjusted life-years (DALYs) (where one DALY represents a year of full health (WHO, 2021)) and 14.6% of years lived with disability. Vigo et al. (2016) estimate that those numbers are in reality much higher, however. They estimate that mental illness accounts for 13.0% of DALY globally (Vigo et al., 2016). Vigo et al. (2016) also estimate that 32.4% of years lived with disability globally are due to mental illness. Walker et al. (2015) estimate that about 14% of global deaths are due to mental illness. The burden mental illness places on society is huge.

According to the 2019 Global Burden of Disease study (Institute for Health Metrics and Evaluation, 2021), depressive disorders in particular account for 1.84% of DALYs and 5.45% years lived with disability globally. This makes depressive disorders the biggest contributor to DALYs out of the categories of mental disorders in the 2019 Global Burden of Disease study (Institute for Health Metrics and Evaluation, 2021). These statistics imply that if our goal is to have as high an impact as possible with the resources we have, depressive disorders are a target worth considering. If we could treat depression better, we could improve life for a significant number of people.

1.3 The subjective burden of mental illness

The statistics above give the objective facts about the extent of the suffering caused by mental illness. I would like to suggest, however, that some notes about the subjective experience of mental illness are also appropriate here, perhaps more so than they would be for illnesses that

[2] Please note that, while I frequently refer to conventional categories like major depressive disorder (MDD), one of the goals of this dissertation is to sketch out ways we might go beyond the current approach to psychiatric diagnosis, so I am questioning the very categories I am using. It may turn out that categories like MDD, depressive disorders and depression are not optimally useful ways to classify the difficulties people face, but for now they are in widespread use.

less directly affect the mind. For example, someone's story about their depression may be more likely to help us understand depression than someone's story about their heart attack can help us understand heart attacks. My argument for this suggestion rests on the fact that mental illness is, by definition, illness of the mind. The mind is an experiencing entity, and its illness directly affects its experiences. Describing the experiences of an unwell mind potentially does more than engender empathy for the person having those experiences: it provides information about the nature of the illness itself. Phenomenological descriptions from sufferers themselves are by their nature subjective and conveyed in emotionally charged, figurative language. I discuss in section 5.5 how we can make use of such descriptions to assist in the scientific study of mental illness. For now, I merely give some examples of the kind of descriptions I have in mind.

One adolescent interviewed by Jackson and Peterson (2003) vividly described the physical aspects of their depressive state:

> I would try to sleep, and there was this physical sensation, this huge surge as if I would have to vomit. It would build up in my stomach and right up in my throat, and it would not be any physical material, just *feeling*. My face would be red and my neck bulging with feeling. And my mind would be rolling over all this very dark material, something I could not control, something so powerful.

Another participant in a study by Jackson (1998) described a state of anhedonia:

> I hit this state of absolute indifference to anything... nothing meant anything to me... I was not feeling anything... I absolutely felt nothing...It's beyond being bored... a complete indifference and lack of empathy to... anything... I didn't feel depressed... I could never have forced myself to cry... I kind of got beyond that...

There are many more such subjective descriptions of mental illness out there. If you are interested in reading more, I can recommend *An Unquiet Mind: A Memoir of Moods and Madness* (Jamison, 1996), *The Noonday Demon: An Atlas of Depression* (Solomon, 2014) and two excellent blog posts by Allie Brosh (Brosh, 2011, 2013).

1.4 Inadequacy of current treatment of depressive disorders

Unfortunately, treating depressive disorders is challenging. Although our current attempts to treat depressive disorders are somewhat effective, there is much room for improvement. Cipriani

et al. (2018) compared the response rates of several antidepressants to placebo and found odds ratios ranging from 2.13 to 1.37.[3] For example, the odds ratio for a response to sertraline, a commonly used antidepressant, was 1.67 (Cipriani et al., 2018). The same study found odds ratios for remission (as opposed to merely a response) ranging from 1.98 to 1.23. For example, the odds ratio for remission in sertraline compared to placebo was 1.52.

On the basis of these numbers, it seems that antidepressants are helpful in effecting remission from depressive disorders, but to put these statistics into perspective we also need to consider the absolute percentages of patients who respond. Cipriani et al. (2018) do not report such percentages, so let us instead consider a meta-analysis of depression treatment in primary care (Arroll et al., 2005).[4] Arroll et al. (2005) found that 56% to 60% of depression patients responded to treatment with a selective serotonin reuptake inhibitor (SSRI), a class of antidepressant. While Arroll et al. (2005) considered the difference between SSRI and placebo to be significant, it is pertinent to note that their findings imply that 40 to 44% of patients do *not* respond to treatment. The situation is slightly better if one attempts a series of treatments: Rush et al. (2009) found that up to two thirds of patients eventually responded to antidepressants when their treatment protocol was followed. However, in most cases patients required two or more treatments to be attempted, which implies a long wait to get better for many patients.

Since there is clearly room for improvement in treating mental illness in general and depressive disorders in particular, researchers have been exploring new tools for understanding such illness. One promising avenue is to study mental illness using methods and concepts from quantitative fields like mathematics, computer science and statistics. Computational psychiatry is a relatively new field of study that exploits such quantitative techniques to better understand mental illness (Friston et al., 2014; Huys, Maia, et al., 2016; Montague et al., 2012; Stephan & Mathys, 2014; X.-J. Wang & Krystal, 2014). Since this dissertation falls under this field, I will give you a brief overview of the field so that you may have a better idea of where this dissertation fits in.

[3] A response is defined by Cipriani et al. (2018) as a 50% reduction on a rating scale for depression, while the odds ratio is the probability of the antidepressant leading to remission divided by the probability of the placebo leading to remission.

[4] Arroll et al. (2005) reports relative risks of improvement, not odds ratios, so it is difficult to compare their results with those of Cipriani et al. (2018). According to A. Cook and Sheikh (2000), relative risk is calculated as the ratio between two probabilities, while an odds ratio is a ratio between two odds, where the odds of improvement would be the number of people who got better divided by the number who did not get better. For example, if one studies 5 people and 1 gets better, the probability of improvement is $1/5$, while the odds of improvement are $1/4$. Arroll et al. (2005) found a relative risk of improvement with SSRI treatment to be 1.37, but it is not clear how this would compare to the odds of improvement.

1.5 Computational psychiatry

Computational psychiatry can be divided into two main areas: data-driven and theory-driven (Huys, Maia, et al., 2016). The data-driven side focuses on classifying data into clusters and making predictions about which data points fit into which categories. Data-driven computational psychiatry is useful when it comes to classification, predicting treatment effectiveness and treatment selection (Huys, Maia, et al., 2016). Theory-driven computational psychiatry is less immediately helpful but just as important (Maia et al., 2017). Theoretical models help us flesh out our conceptual understanding of mental illness. Theoretical models can be subdivided into synthetic models, Bayesian models and algorithmic models (Huys, Maia, et al., 2016).[5] The following paragraph will attempt to distinguish between these three types of models.

Synthetic models begin with biophysical knowledge and make hypotheses about how the different biophysical entities interact (Huys, Maia, et al., 2016). Simulations and mathematics are used to study this interaction; see, for example, Maia and Frank (2011), who apply reinforcement learning models to understanding the role of particular neural structures in psychiatric disorders. Bayesian models, in contrast, start from an analysis of what would be optimal behaviour in a given situation, and attempt to explain observed behaviour with reference to such optimal behaviour.Finally, algorithmic models are simpler and include less biophysical detail than synthetic models, but, like synthetic models, they are grounded in model selection and parameter estimation. They introduce abstract variables into our analysis of behaviour that do not relate immediately to what is going on biophysically.

The reinforcement learning models discussed in this dissertation are algorithmic models. These models, discussed in detail in chapter 2, introduce parameters like reward sensitivity and learning rate. These parameters represent neither biological entities nor directly observable data, but rather serve as intermediate latent variables (Russell & Norvig, 2010, p.816) that expand the conceptual repertoire we use to explain the processes going awry in mental illness. Let us consider how this may be useful. We may have data from a behavioural task that is intended to measure how sensitive participants are to rewards, and we may find that rewards have less impact on one group's behaviour than another. However, it can be difficult to tell without further modelling what the cause of that reduced sensitivity is, and this is where reinforcement learning

[5]The subdivision of theoretical computational psychiatry models into synthetic, Bayesian and algorithmic could be said to map loosely onto Marr's three levels of analysis (Marr, 1982): implementation, computational theory, and representation/algorithm. The divisions given here are specific to computational psychiatry, however, while Marr's levels apply to any information processing device.

models can help us. By fitting a number of reinforcement learning models to the data, each with a different set of parameters, we can determine which model fits the data the best. We might find that the best-fitting model has separate variables for primary reward sensitivity and learning rate. This could be useful knowledge because unhelpful values of these parameters may have different causes and therefore different remedies (Huys et al., 2013).

1.6 Goals of this project

Goals of this project

- Explain how to fit RL models to behavioural data.
- Review studies that have involved fitting RL models to behavioural data.
- Explore the implications of the findings of the above-mentioned studies.

The goals of this dissertation are firstly to explain the above-mentioned model-fitting process; secondly, to survey findings from the literature; and thirdly to explore the implications of those findings for classification of mental illness. By addressing the second goal, surveying the literature, this dissertation aims to find out which reinforcement learning models explain behavioural data the best, and whether there are differences in the values of reinforcement learning parameters between individuals with depressive disorders and healthy individuals. If such differences exist, this raises follow-up questions such as which parameters differ, and whether the parameters in question are bigger or smaller in depressive disorders versus health. As far as the third goal, exploring implications for classification of mental illness, is concerned, we might ask whether values of reinforcement learning parameters inferred from data can play a direct role in diagnosis, whether reinforcement learning models can help reform our concept of mental illness, and what would be a practical way to gather reinforcement learning data from patients. Since this dissertation is a literature review and does not present novel research, presenting and testing hypotheses is not a major focus. However, one prediction for the findings of this review might be that depressed people have an unusually high learning rate for punishment compared to reward; this would be in keeping with the lack of interest and pleasure that depressed individuals experience.

This introduction has given an overview of reinforcement learning and how this field of study

fits into the broader field of computational psychiatry. It has also outlined three goals and raised some questions the rest of this dissertation is going to address. The overview below outlines how the dissertation is organised.

Outline of this dissertation

Chapter 2 gives a broad overview of reinforcement learning and then describes reinforcement learning models that have been used in the behavioural studies cited elsewhere in this dissertation.

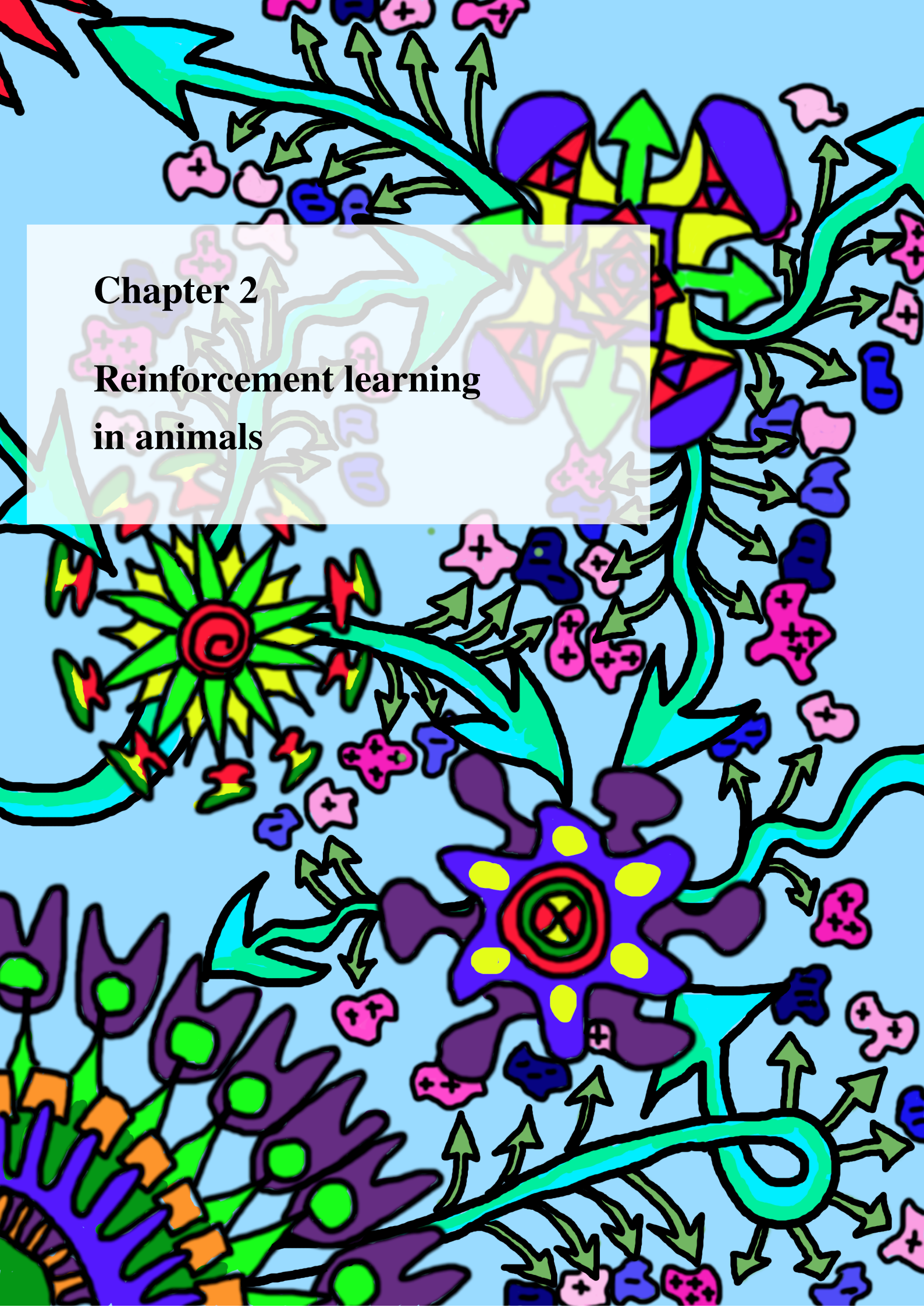
Chapter 3 explores the model fitting techniques that we use to choose among multiple plausible models and to find the parameter values that work best for a given model.

Chapter 4 gives an overview of what researchers have discovered from fitting reinforcement learning models to data from behavioural experiments in humans. It highlights some studies on healthy individuals where researchers compared several different models to identify important parameters for explaining human behaviour. It also discusses results from studies which have examined how these parameters differ between groups of people, particularly healthy people versus people with depressive disorders.

Chapter 5 discusses how the findings from chapter 4 might form the basis for improvements in the ways we classify mental illness. It proposes that it may be feasible to include in our classification systems the values of key reinforcement learning parameters, and suggests that electronic games might be a practical way to gather data for such a diagnostic process.

Chapter 6 concludes the dissertation. It returns to the aims of the dissertation and summarises how the various chapters addressed them. It discusses the limitations of this work and makes suggestions for future research.

We now turn to reinforcement learning models.



Chapter 2

Reinforcement learning in animals

Animals, including humans, learn from rewards and punishments they receive from their environments. Computerised agents can learn that way, too, and it has become apparent that the same algorithms that are used to program such agents can be used to model the learning of animals and humans. This chapter explains a few of these algorithms. Understanding these algorithms provides a doorway into using them to model animal and human behaviour in health and in illness. I kick off the chapter with a description of reinforcement learning and its terminology.

Unless other sources are cited, sections 2.1 to 2.3 of this chapter are based on Sutton and Barto (2020).

2.1 What is reinforcement learning?

There is more than one way we can learn, and some ways work better than others depending on the situation. For example, being explicitly instructed works well for learning that the capital of Brunei is Bandar Seri Begawan, while learning by trial and error is useful for learning to ride a bicycle. This dissertation is concerned with the second type of learning. This kind of learning is known as reinforcement learning, which involves learning by trial and error through repeated interaction with our environment.

We take actions and face the consequences, and this influences our future actions. If the consequences of a particular action are good, we become more likely to repeat that action in that situation. Conversely, if the consequences of an action are bad, we become less likely to repeat it (see Thorndike's Law of Effect (Thorndike, 1911)). When we're learning to ride a bicycle, for example, reinforcement learning is an important component of our learning. At first our movements are somewhat random. When we fall, we learn that whatever we did just before falling probably wasn't helpful, and we become less likely to do it again. As we try again and again, the helpful movements get reinforced and the unhelpful movements get punished, and gradually we learn how to move in such a way that we don't fall over.

Artificial agents can also learn in this way. One can write a computer program that keeps track of the desirability of various actions and updates those desirabilities based on the rewards it receives. Reinforcement learning is a successful branch of machine learning that has developed around this idea. It focuses on creating programs that learn to achieve goals through interaction with environments. Various reinforcement learning algorithms, like TD learning (section 2.10), have been developed. Reinforcement learning has been gaining attention as a way of getting

artificial agents to perform at superhuman level in various tasks like board games (Silver et al., 2016) and old ATARI computer games (Mnih et al., 2013).

Reinforcement learning algorithms from machine learning can be used to model animal (including human) behaviour and neural activity (Sutton & Barto, 1987). The timing and intensity of neural activity was first modelled by Montague et al. (1996) and Schultz et al. (1997). Studies like those by Huys, Cools, et al. (2011) and Huys et al. (2013) have also modelled the choices made by agents. The algorithms provide a framework for explaining large amounts of data and make predictions about what future data might look like. They allow us to model and compare the behaviours of different groups of people in a meaningful way. In this dissertation, I am primarily concerned with the behaviour of mentally ill people compared to healthy people. To build up to that discussion, I need to introduce the reinforcement learning framework and a few algorithms commonly referred to in the relevant literature.

2.2 Reinforcement learning vocabulary

Reinforcement learning scenarios are described using a set of standard vocabulary with specific meanings. The following words will be used in this dissertation:

- Agent: a decision-maker trying to maximise its rewards
- Actions: the choices available to the agent
- Environment: everything outside the agent
- State: the configuration the environment is in at a given time
- Reward: a signal from the environment to the agent.
- Discount factor: a number between 0 and 1 that scales down the value of future rewards
- Policy: a rule for choosing actions

The **agent** in a reinforcement learning problem is a decision-maker that tries to maximise its rewards. In the case of animals, the agent can be the brain or part of the brain. In artificial agents it consists of code that specifies how to respond to various situations.

The agent is in interaction with an **environment**. The environment consists of everything outside the agent (Sutton & Barto, 2020, p.47-48). In a computerised behavioural experiment

involving humans, the environment is in part simulated by code that determines which states and rewards to present to the agent and when to do so. The environment in these experiments can also include physical aspects like being under threat of electric shocks or being given tangible rewards.

A **state** consists of all the information about the environment that is relevant to the agent's future decisions (Sutton & Barto, 2020, p.49). I represent the state the environment is in at a time t as s_t, and when referring to a particular state I do so using a capital S. The state following S is denoted S'. For an agent in a world of only squares (known as a gridworld), the states are the squares of the grid. For a human doing a computerised task as part of one of the behavioural experiments described in chapter 4, states might be pictures on the screen. A state can also consist of a combination of a number of features, in which case it may be useful to represent it using a feature vector $\mathbf{x}$ (Sutton & Barto, 2020, p.204-205), as I do in section 2.11.1.

The agent performs **actions**, chosen according to a **policy**, and these actions impact on the environment to varying degrees. I represent an action at a time t as a_t, and when referring to a particular action I do so using a capital A. The action following A is denoted A'. In some cases, there is a strong correlation between the action the agent takes and the reward or punishment it receives from the environment (for example choosing between two routes home). In other cases, the link is tenuous or non-existent, for example the route you take home has no effect on whether it rains or not (see classical conditioning in section 2.6). In a computerised behavioural experiment with a human subject as the agent, the subject may be asked to provide input using a keyboard or mouse; this input then gets used as the subject's actions.

The environment responds to the agent with rewards and follow-up states. In computer science, even though it is called a reward, it can be either a positive or negative number. In animals, however, aversive events are often treated qualitatively differently from pleasant ones, so then it becomes useful also to introduce the term 'punishment'. I discuss this distinction in chapter 4. Because this dissertation aims to throw light on animal behaviour, I will henceforth refer to a negative reward as a punishment. Rewards and punishments can be either symbolic or concrete. For a computerised agent, feedback comes symbolically, in the form of real numbers. For non-human animal agents, feedback tends to be concrete, for example in the form of food. In a computerised behavioural experiment with humans, feedback can be either purely symbolic (in the form of points or happy/sad faces) or have a concrete component. On the concrete side, rewards may come in the form of food or money, and punishments may come as electric shocks

(Mkrtchian et al., 2017).

The state is a collection of all the information that is relevant to an agent in its decision-making. In a computerised task involving humans, the state would be a collection of all information given to the subject. This could consist of an image on a screen and/or a sound, or a square in a gridworld (see section 2.4 for more about gridworlds).

Sometimes it makes sense to value rewards that occur further into the future less than immediate rewards; in those cases we scale down the value of future rewards with a discount factor, a number between 0 and 1. In this dissertation, I denote discount factors by the symbol γ. For each additional time step the reward is into the future, an additional factor of the discount factor gets applied. For example, for a reward r two time steps into the future, the amount that reward contributes to the current state will be $\gamma^2 r$.

2.3 Interaction between the agent and the environment through actions and rewards

In reinforcement learning, agents aim to predict and maximise their rewards. They do this by learning from rewards and penalties and modifying their actions in response. A signal that predicts a later concrete reward or punishment can start to serve as a reward or punishment in itself as the agent learns the association. An agent exists in an environment, and that environment can be in various states. The agent takes actions based on the information it has about its state, and the environment responds by giving the agent a scalar reward or punishment and changing into the next state. The new state and reward are in some cases dependent on the agent's action, and in other cases independent of it. This interplay between states, actions and rewards/punishments is at the heart of reinforcement learning.

The agent uses a function called a value function to represent the desirability of a particular state (or state-action pair). In simple cases, where the states are small in number, the value 'function' might be simply a table of values, one for each state. In more complex cases the value function might depend on the state S and a weight vector $\mathbf{w}$ and be written as $\hat{V}(S, \mathbf{w})$. The value function is used by the agent when deciding which action to take in a given situation.

The above process of interaction between the agent and the environment through actions and rewards is studied in human agents using (mostly computerised) behavioural tasks. The states are usually pictures on the screen of an electronic device, or sounds played by the device.

Subjects perform actions using input devices like keyboards or touchscreens. Rewards/penalties are indicated symbolically on the screen using points or happy/sad/neutral faces, and may later be followed by concrete rewards like money. The states, actions and rewards are recorded, and reinforcement learning algorithms can then be fitted to these data. I will describe a number of such tasks throughout this dissertation. Sometimes using such behavioural tasks as examples will not make an explanation optimally clear; in such cases I will explain the reinforcement learning process with reference to a simple environment, known as a gridworld.

2.4 Gridworld

A gridworld is a simple class of examples of reinforcement learning environments that I will use to illustrate the TD learning algorithm in section 2.10. A gridworld consists of a number of squares arranged in a grid. Every square represents a state. An agent starts in one of these squares and is able to make decisions about which adjacent square to move towards next. The decisions are known as actions. Some actions result in numerical rewards or punishments. Eventually the agent reaches a square designated as an endpoint, and the game ends. (Sutton & Barto, 2020)

Figure 2.1 shows a gridworld called BookGrid, the example I will be using. BookGrid is one of the gridworlds implemented in the Berkeley AI reinforcement learning assignment (DeNero & Klein, 2014) and explained by Zhang (2019). The actions available to the agent in this grid are 'up', 'right', 'down' and 'left'. In the Berkeley AI reinforcement learning assignment code, one can set a degree of randomness in the square the agent actually ends up in (as opposed to where it chose to go), but for explanatory purposes I will stick to zero randomness (i.e. the agent goes exactly where one would expect). If the agent's movement takes it into a wall (e.g. it is against the left wall and tries to move left), it will simply stay in the same square. The agent receives zero reward for all moves except ones from the two squares that end the game. For exiting one of these squares, the agent gets feedback of $+1$ and for exiting the other it gets -1.

Gridworld is not usually a puzzle given to human subjects to solve; it is more often a challenge given to artificial agents to demonstrate the performance of a particular algorithm. Because this dissertation focuses on the reinforcement learning process in humans and animals, I now take a diversion into psychology to lay the groundwork for the later discussion of the methods and results of behavioural experiments.

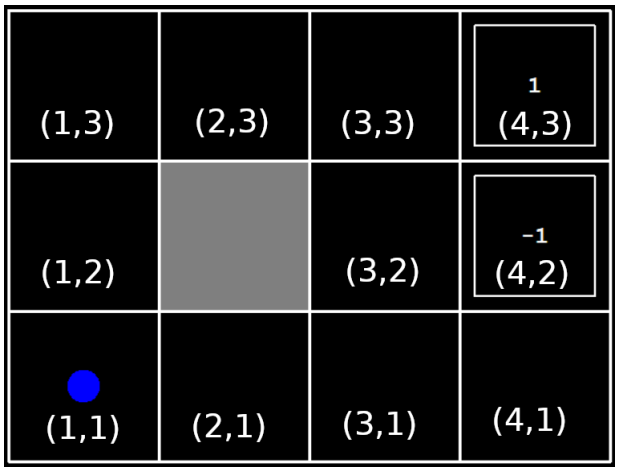


Figure 2.1: **BookGrid.** In this gridworld, the agent starts in the bottom left square. It gets zero reward for all moves except the moves out of the (4,2) and (4,3) squares. Moving out of the (4,3) square to the terminal state gives a reward of 1. Moving out of the (4,2) square to the terminal state gives a reward of -1. Credit goes to DeNero and Klein (2014) for generating the grid on which this image is based.

2.5 Model-free vs. model-based learning and decision-making

Learning and decision-making are closely related concepts. Through the process of learning, animals change the way they make decisions. Learning would have no purpose if it did not result in decisions, while decisions would be random without learning. Animals use two related processes to learn and make decisions: model-free and model-based. In the model-free process, the learning aspect is usually emphasised, so we talk about model-free *learning*. In the model-based process, learning plays a less prominent role than the process of arriving at a decision, so we often speak of model-based *decision-making*.

During model-free learning, we learn by trial and error: when our expected rewards differ from our actual rewards, we update our expected rewards. Gradually, we become more likely to repeat behaviours that have been successful in the past. (Russek et al., 2017). This kind of learning is also known as habitual learning, and allows decisions to be made quickly (Huang et al., 2020). During model-based decision-making, on the other hand, we use an internal model of our environment to predict the consequences of behaving in a particular way (Russek et al., 2017). Model-based decision-making is also known as goal-directed decision-making. Decisions are made deliberately and more slowly than in model-free decision-making (Huang et al., 2020).

Algorithms have been developed to model these two types of learning and decision-making.

The Rescorla Wagner and temporal difference learning algorithms, discussed later in this chapter, are both model-free algorithms. Model-based methods include dynamic programming methods where the agent determines values of states based on a full model of the environment without actually interacting with the environment. In order to keep the scope of this dissertation manageable, I have restricted myself to discussing model-free decision-making because that is what most of the research on decision-making in depression focuses on (see chapter 4). It is worth noting, however, that Daw et al. (2011) found that model-free and model-based decision-making are integrated in the brain and cannot be easily separated. Additionally, impaired model-based decision-making has been associated with psychiatric symptom dimensions like 'compulsive behaviour and intrusive thought' (Gillan et al., 2016). It therefore does not seem far-fetched to imagine that some links between depression and model-based reasoning will eventually emerge.

Another useful distinction made in psychology is between instrumental and classical conditioning.

2.6 Instrumental (or operant) vs. classical (or Pavlovian) conditioning

The explanations in this section are largely drawn from chapter 14 of Sutton and Barto (2020).

In the literature on behavioural experiments in psychology, learning has traditionally been divided into two categories: instrumental (or operant) conditioning and classical (or Pavlovian) conditioning. Since this dissertation focuses on animal learning and behaviour, I will frequently make reference to these two types of learning. When an animal is in a learning situation where their actions have an impact on the rewards and penalties they get from their environment, this is referred to as instrumental conditioning. Responses from the animal that aim to achieve a particular outcome in such a context are referred to as instrumental responses. Classical (Pavlovian) conditioning, on the other hand, occurs when an agent learns about the relationship between a stimulus and an outcome, where the outcome has nothing to do with the agent's actions. In response to a classically conditioned stimulus, the animal will respond by either approaching or withdrawing from stimuli, where the stimuli have been learned to predict rewards or punishments (Huys, Cools, et al., 2011).

With its focus on actions, it is easy to see how instrumental conditioning fits into the reinforcement learning framework: the agent takes actions that influence the rewards and penalties

given to it by the environment. Explaining how classical conditioning fits into the reinforcement learning framework is a bit trickier. Actions are an important part of reinforcement learning, after all; if the learning did not in some way change the agent's behaviour, it would arguably have no point. However, a prerequisite for informed changes in behaviour is *prediction* of the pattern of rewards and penalties likely to occur in future. In situations where the animal's actions have no impact on the rewards and penalties they are likely to receive, we refer to this process of prediction as classical conditioning (Sutton & Barto, 2020, p.342).

Now I move on to descriptions of the algorithms themselves. I discuss these algorithms in the context of animal behaviour. In the case of Rescorla Wagner, discussed in the next section, we will see that the complexity of animal behaviour has made it useful to develop a number of variations of the original learning rule.

2.7 Rescorla Wagner

A simple example of reinforcement learning is the Rescorla Wagner (RW) model. I will discuss it with reference to a simple hypothetical behavioural task of my own design involving mushrooms. In this task, human participants are shown mushrooms one at a time on a screen (from a pool of four; see figure 2.2). They are asked either to eat the mushrooms or refrain from eating them. Two of the mushrooms are good to eat and the other two are not. When a participant eats a good mushroom, there is a 70% probability that the environment responds with a reward and a 30% probability that it responds with a punishment. When a participant eats a bad mushroom, there is a 70% chance of a punishment and a 30% chance of a reward. Refraining from eating a mushroom always results in neutral feedback. Let us now consider the RW model in the context of this game.

Imagine you are playing this mushroom game and you are shown the mushroom with the star pattern. It's the start of the experiment and you have no idea what reward to expect from either eating or not eating the mushroom. You decide to eat the mushroom (because you like eating interesting mushrooms) and you expect a reward of 0. When you eat the mushroom with the star pattern, you get an actual reward of 1.

That means the difference between your actual reward and your expected reward was $1-0=1$. We can call this difference your *prediction error*,

$$\delta_t = r_t - Q_t(S, A)$$

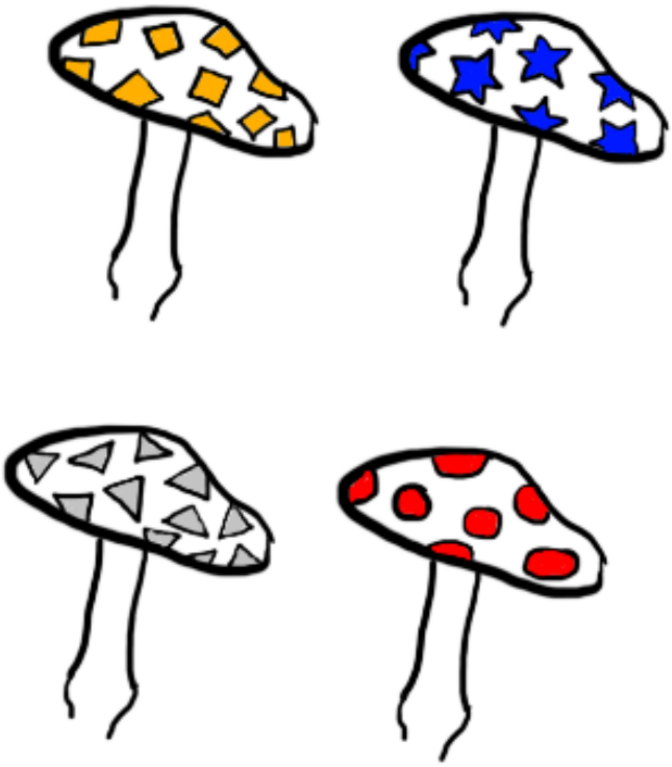


Figure 2.2: An example of four different mushrooms participants get shown one by one in the simplified mushroom task

where t indicates the trial number, r_t is your reward for trial t, and $Q_t(S, A)$ is the reward you expected to get for taking action A from state S. In this example the state S is the mushroom with the star pattern and the action A is the act of eating the mushroom.

Now you may decide to update your expected value for this mushroom and this action for the next trial as follows:

$$Q_{t+1}(S, A) = Q_t(S, A) + \alpha\delta_t \tag{2.1}$$

where α is your learning rate, which determines how much you let your previous prediction error δ affect your new expected value for this mushroom and this action. It is usually somewhere between 0 and 1.

Sometimes it is more useful to consider the overall value of a state instead of the value of

taking a particular action from a state; then the RW update rule is written in the following form:

$$V_{t+1}(S) = V_t(S) + \alpha\delta_t \tag{2.2}$$

Guitart-Masip et al. (2012), for example, find this form useful to model Pavlovian as opposed to instrumental learning.

2.8 Deciding what actions to take

Having just discussed our first rule for updating the values of states or state-action pairs, it is natural to ask what we do with these values once we have them. In an instrumental conditioning situation, a vital part of the reinforcement learning process is deciding on actions. Our model therefore also needs to include a hypothesis about how the agent chooses actions given the values it attaches to those actions. This is what Daw (2011) calls an observation function. Usually this observation function takes the form of a probability of taking a given action given a state; often this probability is a softmax distribution like

$$p(A|S) = \frac{e^{\beta Q(S,A)}}{\sum_i e^{\beta Q(S,A_i)}} \tag{2.3}$$

where β is a parameter called inverse temperature. β specifies the degree of randomness (exploration) in the agent's decisions, where higher β means less randomness in decisions.

In the next section I outline some important variations in both the RW update rule and the observation function (action selection rule) that goes with it. I leave a discussion of actually fitting these models to data for chapter 3.

2.9 Variations of Rescorla Wagner

In search of a model that fits their data well, researchers come up with several models and compare how well they fit (more about this in chapter 3). Thanks to this process, several variations of RW have been developed; see, for example, Guitart-Masip et al. (2012), Huys, Cools, et al. (2011), Huys et al. (2013), Kunisato et al. (2012), and Mkrtchian et al. (2017).

A common variation involves expressing reward prediction error with a reward sensitivity factor ρ multiplying the reward, so that reward prediction error is written as

$$\delta_t = \rho r_t - V_t(S) \tag{2.4}$$

This reward sensitivity factor ρ modulates the effect of rewards on the reward prediction error. This turns out to be useful when modelling human learning because a meaningful way to group people is in terms of their sensitivity to reward. Chapter 4 discusses correlations that have been found between mental illness and this parameter, and how that relates to learning rate α.

Things become a bit more complicated when we consider that animals frequently respond to different degrees to rewards and punishments, such as in Huys, Cools, et al. (2011). Due to this, we sometimes find it helpful to define separate feedback sensitivities and learning rates for rewards and punishments. This will be discussed in chapter 4. An update for the action (Q) value of a state for such a setting might be (Huys, Cools, et al., 2011; Kunisato et al., 2012)

$$Q_{t+1}(S,A) = Q_t(S,A) + \alpha(\rho r_t - Q_t(S,A)) \tag{2.5}$$

where

$$\alpha = \begin{cases} \alpha_{\text{rew}} \text{ in case of reward} \\ \alpha_{\text{pun}} \text{ in case of punishment} \end{cases}$$

and

$$\rho = \begin{cases} \rho_{\text{rew}} \text{ in case of reward} \\ \rho_{\text{pun}} \text{ in case of punishment} \end{cases}$$

Another variation is to introduce a more complex action-value function known as the weight $\mathcal{W}$, so that probabilities are assigned to actions using a function like

$$p(A|S) = \frac{e^{\beta \mathcal{W}(S,A)}}{\sum_i e^{\beta \mathcal{W}(S,A_i)}} \tag{2.6}$$

The weight for an action includes its Q value as well as other factors that might influence action selection. For example, Huys, Cools, et al. (2011) add a fixed bias parameter b which incorporates a general bias towards 'go' actions:

$$\mathcal{W}(S,A) = \begin{cases} Q(S,A) + b & \text{if } A = \text{go} \\ Q(S,A) & \text{otherwise} \end{cases} \tag{2.7}$$

Above, b is zero for 'nogo' actions and, depending on the model, may or may not be allowed to differ for withdrawal 'go' vs. approach 'go' actions.

Guitart-Masip et al. (2012) and Mkrtchian et al. (2017) additionally include a Pavlovian bias parameter $\kappa \geq 0$, which they multiply by the Pavlovian value $V(S)$ of the current state, so that

$$\mathcal{W}(S,A) = \begin{cases} Q(S,A) + b + \kappa V(S) & \text{if } A = \text{go} \\ Q(S,A) & \text{otherwise} \end{cases}$$

The Pavlovian bias parameter κ takes into account that certain behaviours are easier to perform under certain conditions, for example it is easier to take action ('go') as opposed to doing nothing ('nogo') after being shown a rewarding stimulus, and it is easier to inhibit action ('nogo') after being shown an aversive stimulus. Note that, above, $\kappa \geq 0$, and for a stimulus S associated with punishment we have $V(S) < 0$, so $\kappa V(S)$ will be negative and thus reduce the weight for the go action, making it relatively less likely. By similar reasoning, for a stimulus associated with reward, a 'go' action becomes relatively more likely. Depending on the model, κ may have different values depending on whether the agent is approaching reward or withdrawing from punishment.

Huys, Cools, et al. (2011) account for Pavlovian bias differently. They add a term $f(S,A)$ to each weight $\mathcal{W}(S,A)$ from equation 2.7. For all 'nogo' actions, they let $f(S, \text{nogo}) = 0$. For 'go' actions, each of the five Pavlovian stimuli has its own value which is allowed to vary independently of the others. When it comes to action selection, they then use a softmax rule similar to equation 2.6, but where they leave out inverse temperature β and instead of $\mathcal{W}(S,A)$ put $\mathcal{W}(S,A) + f(S,A)$ in the exponent:

$$p(A|S) = \frac{e^{\mathcal{W}(S,A)+f(S,A)}}{\sum_i e^{\mathcal{W}(S,A_i)+f(S,A_i)}}$$

Guitart-Masip et al. (2012) and Mkrtchian et al. (2017) also used a softmax rule similar to equation 2.6 as their observation function, but they left out temperature, and included an extra parameter ξ that Guitart-Masip et al. (2012) referred to as 'noise' and Mkrtchian et al. (2017) referred to as 'lapse', as follows:

$$p(A|S) = \frac{e^{\mathcal{W}(S,A)}}{\sum_i e^{\mathcal{W}(S,A_i)}}(1-\xi) + \frac{\xi}{2}$$

ξ was allowed to vary between 0 and 1. It accounted for the fact that there is a certain degree of randomness in humans' actions.

The last variation of the RW update rule we will consider is specific to a task we have not discussed yet, the probabilistic reward task. It features in a paper which will be important in chapter 4, so I explain it here.

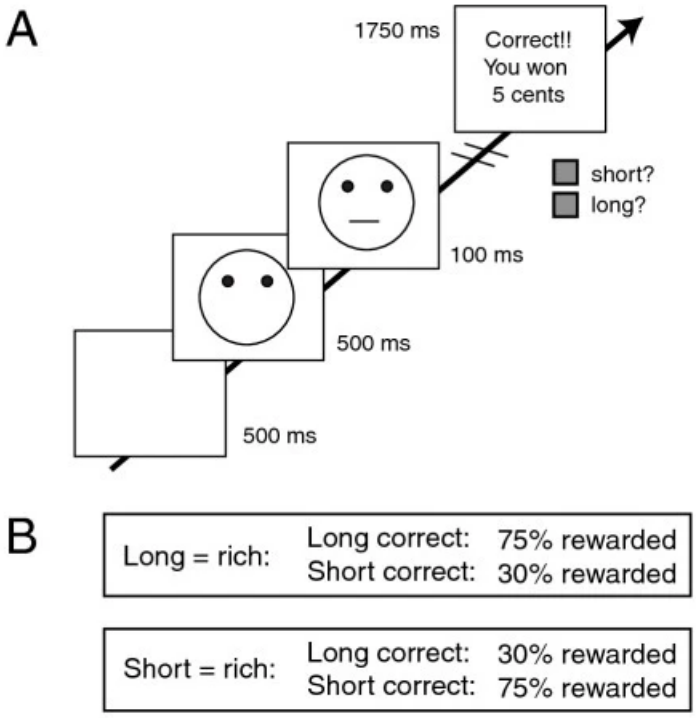


Figure 2.3: This figure depicts the probabilistic reward task and is taken directly from Huys et al. (2013). First, a face without a mouth is displayed on the screen, and then either a long or short mouth is added. The participant is asked to say whether they think the mouth is long or short. If the long mouth is defined as being the rich stimulus, correctly pointing out the long mouth gets rewarded 75% of the time, while correctly pointing out the short mouth only gets rewarded 30% of the time. The opposite reward pattern occurs when the short mouth is the rich stimulus.

2.9.1 RW variation for the probabilistic reward task

In this task, participants get shown a series of cartoon faces on the screen, as shown in figure 2.3 (Huys et al., 2013; Pizzagalli et al., 2008). The faces' mouths have two possible lengths: long and short. Subjects are asked to press buttons to indicate whether they believe the long or the short mouth was being displayed at a given time. Some correct responses get rewarded and others receive no feedback. Incorrect responses never get feedback. One of the lengths gets rewarded more frequently than the other when it is correctly identified; this length is called the 'rich' stimulus. For example, if the long mouth is defined by the experimenters as the rich stimulus, correctly pointing it out gets rewarded 75% of the time, while correctly pointing out the short mouth gets rewarded 30% of the time. Participants with high reward responsiveness

develop a stronger bias towards selecting the rich stimulus, which yields the more frequent rewards when chosen correctly (Vrieze et al., 2013). The uneven reward probabilities mean that, when in doubt, it is frequently more rewarding to say that one saw the 'rich' stimulus instead of choosing at random. This bias goes beyond learning to be more accurate; it technically makes the person's responses *less* accurate.[1]

Huys et al. (2013) gave their subjects the probabilistic reward task. Their model was also based on RW, but departs from it further than the variations I have discussed until now. Due to the design of the task, it was advantageous for subjects to develop a response bias in favour of the rich stimulus.

In Huys et al.'s (2013) base model, actions are chosen using a softmax function similar to equation 2.6, except that there is no β parameter and there are only two actions, a_t and the action not taken, $\bar{a}_t$:

$$p(a_t|s_t) = \frac{e^{\mathcal{W}(s_t,a_t)}}{e^{\mathcal{W}(s_t,a_t)} + e^{\mathcal{W}(s_t,\bar{a}_t)}}$$

The weights are calculated using

$$\mathcal{W}_t(s_t, a_t) = \eta\mathcal{I}(s_t, a_t) + \zeta Q_t(s_t, a_t) + (1 - \zeta)Q_t(s_t, \bar{a}_t).$$

The term $\mathcal{I}(s_t, a_t)$ is 1 when a subject takes the action that matches the true current state, for example correctly reporting that the long mouth had been displayed. η is a free parameter that determines to what degree a subject makes the instructed choices (i.e. choices that match the image displayed on the screen). ζ is a probability that represents how certain a subject is, on average, that they had truly seen the stimulus they thought they'd seen. Q then gets updated like in equation 2.1.

I refer again to this model and the other variations of RW discussed above in chapter 4 when I discuss what studies have found from fitting data to these models. For now, I move on to temporal

[1] Assume that the rich stimulus gets rewarded with probability 75% and the other stimulus with a probability of 30%. All non-zero rewards equal 1. Now assume you are 50% sure you saw the rich stimulus. Then the expectation value of saying you saw the rich stimulus is $\mathrm{E} = \sum_r rp(r) = 0 \times p(0) + 1 \times p(1) = p(1) = p(\mathrm{rich})p(r = 1|\mathrm{rich}) = (0.5)(0.75) = 0.375$ and that for saying you saw the other stimulus is $(0.5)(0.3) = 0.15$. Clearly in this case it is better to say you saw the rich stimulus. This remains true even if you are slightly less than 50% sure you saw the rich stimulus; for example being 30% sure leads to expectation values of 0.225 and 0.21 for saying you saw the rich and non-rich stimulus respectively. At what point does this stop being true? Let your certainty of seeing the rich stimulus be c. Then if we solve the equation $0.75c = 0.3(1 - c)$ we get approximately $c = 0.286$. Therefore the break-even point, where it does not matter which stimulus you say you saw, is when you are approximately 28.6% certain you saw the rich stimulus. Of course, human brains are unlikely to represent degree of belief with such a high numerical precision.

difference learning models, which depart more significantly from the plain RW model than the models I just discussed. These models have not been extensively used in behavioural modelling (I speculate about reasons for this in section 2.12), but they have been used to predict the timing of reinforcement learning signals in the brain, which I believe makes them worth mentioning in a dissertation on reinforcement learning algorithms and animal learning. Additionally, a version of temporal difference learning called $SARSA(\lambda)$ has been used to model learning in two-step tasks (Daw et al., 2011; Kool et al., 2016), and having a basic understanding of temporal difference learning is a prerequisite to understanding that algorithm. Modelling tasks with two or more steps is beyond the scope of this dissertation, but doing so will be necessary to implement the suggestions I make in chapter 5 about using game-based tasks to learn about the ways people learn and make decisions.

2.10 Temporal difference (TD) learning

Please note that, in the following sections, the subscript t indicates a time step within a trial (episode) and not the number of the trial like in the case of RW. Also, in keeping with the notation in Sutton and Barto (2020), I will now use r_{t+1} instead of r_t to denote the reward received after entering state s_t.

The advantage of the RW update rule is that it is simple, but it falls short in that it does not distinguish between time steps within trials. This means it cannot make predictions about what will happen if the timing of stimuli within a trial gets varied (Sutton & Barto, 1987). Another, perhaps clearer, way to understand where RW falls short is to imagine how a RW agent would perform in a gridworld (see section 2.4). The agent would keep track of the value of each state or state-action pair, and on each move it would calculate the prediction error as the difference between the actual feedback received and the value it expected. Imagine a RW agent starting with all its state values equal to zero. If it exited state (4,3) in BookGrid (figure 2.1) on its first trial, its prediction error would be 1 and its value of state (4,3) would change by the product of the learning rate and prediction error. However, this would be of little help to the agent because each episode starts back in state (1,1). There would be no way for the information about what happened when leaving state (4,3) to 'propagate' back to state (1,1) and guide the agent's behaviour there. Temporal difference learning addresses this problem.

We can deal with the above-mentioned problem by calculating our agent's prediction error in a new way. First I will simply give the formula, and then I will explain it with reference to

BookGrid (figure 2.1).

We can write the prediction error as follows:

$$\delta_t = r_{t+1} + \gamma V(s_{t+1}) - V(s_t)$$

We can motivate this formula as follows: In TD learning, unlike in RW, our agent's value function V represents the expected value of *all* our agent's future discounted rewards, not just the reward it expects to get from the current state. When our agent takes an action and experiences a reward, it has to figure out the difference between the experienced reward and what it expected. However, its expected value was for all future rewards, not just the next reward. So it needs to add the actual reward to the expected values of the remaining steps, and find the difference between that sum and the old expected total reward. Another way to think about it is as taking the experienced reward and subtracting the difference between the old expected value of all the states from t and the expected value of all the states from $t + 1$.

Let us consider how this new prediction error helps the information about what happens late in a trial 'propagate' back to early in the trial. Imagine that we are an agent in the gridworld BookGrid (figure 2.1). Recall that in BookGrid we start in state (1,1). If we exit state (4,3), the episode ends and we get a reward of $+1$. If we exit state (4,2), the episode ends and we get a penalty of -1. The other states all have a reward of 0. Let us set the discount factor to 1 and the learning rate α to 1.

We initially move around at random because the expected values for all the state-action pairs are 0. Now imagine we are in state (3,3) and after our next action we end up in state (4,3). We exit state (4,3) and get a reward of 1. This leads to a prediction error: we had no reason to expect a reward, and now we have received one. The value we previously attached to state (4,3) was 0, so if we were using a RW update rule our prediction error would be $\delta_t = r_t - V_t(s_t) = 1 - 0 = 1$. I explained above why this wouldn't help us. Because we are now using temporal difference (TD) and considering what might happen in later time steps of this trial, our calculation of the prediction error should also include the expected value of the state we went to after exiting state (4,3). In this case, however, that state is what is referred to as the 'terminal state' because the trial has ended, and the value of the terminal state is by convention always zero (Sutton & Barto, 2020), so coincidentally our prediction error ends up being the same as it would have been for RW. Soon, however, the usefulness of this new prediction error calculation will become clear.

The next episode starts and we move around a bit. Imagine we end up moving from state (3,3) to state (4,3). This is where RW and TD differ. The RW prediction error would have been $r_t - V_t(s_t) = 0 - 0 = 0$. The TD prediction error, on the other hand, is

$$\delta_t = r_{t+1} + \gamma V(s_{t+1}) - V(s_t) = 0 + 1 - 0 = 1$$

This means the new value of state (3,3) is 1, which means the agent now has access to important information about where to move one step earlier. On the next episode, if the agent moves from (2,3) to (3,3), the value of (2,3) will become 1, and as the episodes continue this value will continue to propagate to earlier and earlier states, until it is providing guidance to the agent from its first move.

Below is the above process formalised in pseudocode. This pseudocode is for looking one step ahead and applies to instrumental learning, where the agent's actions matter:

Algorithm 1: Pseudocode for using one-step-ahead temporal difference learning to learn the values of states. Actions are taken according to policy π. S' is used to denote the state subsequent to state S, just as A' denotes the action subsequent to action A. Adapted from Sutton and Barto (2020)

```
Initialise V(S)=0 ∀ S;
Input policy π;
Input learning rate α;
for each episode do
    Initialise S;
    while S is not terminal do
        A ← action given by policy π for S;
        Take action A;
        R, S' ← the output from taking action A;
        V(S') ← V(S) + α(R + γV(S') − V(S));
        S ← S';
    end
end
```

2.11 TD learning to predict reward in classical conditioning

While the above algorithm is for TD learning in the context of instrumental conditioning, the TD update rule is also useful for keeping track of state values in the context of classical conditioning. It was used for this purpose by Kumar et al. (2008), Montague et al. (1996), and Schultz et al. (1997). In these papers, the TD algorithm predicts the firing of neurons in the brain in response to a stimulus that predicts reward. I will not delve into neuroscience, but rather just show how the algorithm enables an agent to associate a previously unrelated stimulus to a reward. Making such associations requires some additional formalism, which I introduce in the next section.

2.11.1 Value functions, weights and stimulus representation vectors

In classical (Pavlovian) conditioning experiments, stimuli get presented to the agent. Sometimes such a stimulus is not rewarding or punishing by itself, but becomes associated with a rewarding or punishing outcome by preceding that outcome sufficiently many times. Such a stimulus is called a conditioned stimulus (CS).

When we are modelling findings from classical (Pavlovian) conditioning experiments, not only the stimuli presented to the animal matter; their timing also matters and can dramatically influence the results (Dayan & Abbott, 2001, chapter 9). In these cases, there is no simple way to define states like in the mushroom experiment from section 2.7. Instead, we replace tables of values with value functions, which depend on a vector of weights and a stimulus representation vector. I denote a value function by $\hat{V}$ in this dissertation.

2.11.2 Representing a stimulus and its timing

The environment gets represented using a stimulus representation vector $\boldsymbol{x}$. $\boldsymbol{x}$ contains all the relevant information about the environment, including which stimulus is presented when. In the experiments I will be discussing, an element of $\boldsymbol{x}$ is 0 if a stimulus is absent and 1 if it is present. Contrary to what one might expect, $\boldsymbol{x}$ is not a vector with one element for every time step; it is in fact two-dimensional, with the second dimension having as many elements as there are time steps following the CS. This allows us to model the agent keeping track of the number of time steps that have elapsed since the CS was seen. This works as follows.

When we model learning processes where timing matters, for example learning that a light predicts reward after some delay, our agent needs some way to 'remember' the stimulus (say, a

light being switched on for one time step) in order to link it with the reward that comes later. If our prediction error only looks one time step into the future, and the reward comes, say, three steps later than the light is switched on, our agent needs some other way to connect the light and the reward. One way to do this, instead of having a single vector representing the light, is to have several vectors representing the 'memory' of the light. So if the light gets presented at time $t = 2$, and I am considering 5 time steps, then I will have stimulus vectors $\langle 0, 0, 1, 0, 0\rangle$, $\langle 0, 0, 0, 1, 0\rangle$ and $\langle 0, 0, 0, 0, 1\rangle$.[2]

When doing simulations, we can conveniently arrange the stimulus vectors into a matrix where the rows represent the vector number i and the columns represent the time steps t:

$$x = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.8}$$

If the stimulus is presented at time t, then $x_{i,t+i} = 1\ \forall i$ and all other elements of x are 0.

2.11.3 Weight vector w

We introduce a vector $\mathbf{w}$. This is called a weight vector, and it represents the degree to which each input stimulus influences the value function $\hat{V}$. The same state may have different values if two different weight vectors get fed into the value function $\hat{V}$. The weight vector is the same length as $\boldsymbol{x}(t)$. We consider only the case where $\hat{V}(\boldsymbol{x}, \boldsymbol{w}) = \sum_i x_i(t) w_i$. The weights start at zero and get updated at the end of every trial.

2.11.4 Finding $\hat{V}$

At each time step, the value $\hat{V}$ gets updated using the following formula:

$$\hat{V}(t) = \sum_i w_i x_{i,t} \tag{2.9}$$

Directly afterwards, δ gets updated as follows:

$$\delta(t) = r(t) + \gamma \hat{V}(t+1) - \hat{V}(t) \tag{2.10}$$

[2]On the surface it seems to make sense to have a vector $\langle 0, 1, 0, 0, 0\rangle$, too, since the light is actually on at $t = 2$, but it turns out that this leads to the prediction error occurring at $t = 1$.

At end of each trial the weights get updated using

$$w_i \longleftarrow w_i + \alpha \sum_t x_{i,t}\delta(t). \tag{2.11}$$

Algorithm 2 summarises this process.

Algorithm 2: Algorithm used to obtain prediction errors and values in figure 2.4

Initialise x as an $n_{\text{timesteps}} \times n_{\text{trials}}$ matrix ;
for *each trial* **do**
  for *each time step in current trial* **do**
    $\hat{V}(t) = \sum_i w_i x_i(t)$;
    $\delta(t) = r(t) + \gamma\hat{V}(t+1) - \hat{V}(t)$;
  end
  for i *in* $\boldsymbol{w}$ **do**
    $w_i \longleftarrow w_i + \alpha \sum_t x_i(t)\delta(t)$;
  end
end

Applying this algorithm to the five-step example described above gives the output in figure 2.4 for trials 1, 3 and 6.

Let us consider a larger example. We now run the simulation for 120 time steps and 100 trials. A conditioned stimulus (CS) gets presented at $t = 40$ and a reward arrives at $t = 54$. On trial 30, the usual reward is withheld. We can see in figures 2.5 and 2.6 that, by trial 30, there is a positive prediction error that precedes the reward in time. This is thanks to the design of the algorithm, which creates a sort of domino effect: a prediction error at a given time on one trial results in a prediction error one time step earlier on the next trial. This simulates the way a living agent would remember what happened when on previous trials and would adjust its expectations accordingly.

On trial 30, when the reward is suddenly withheld, this causes the prediction error δ to be sharply negative. A reward r was expected, but a reward of 0 was received, so the prediction error is $-r$. On the next trial, however, the reward is given at the usual time, and the agent continues the process of learning that the light predicts reward. By trial 100, the prediction error due to the light being switched on is the same size of the original response to the reward. In a sense, the light itself is now serving as a reward.

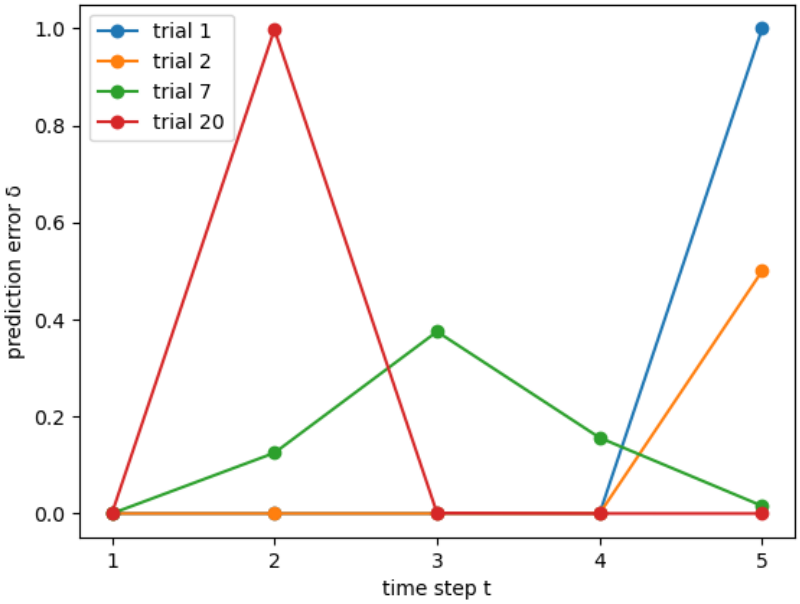


Figure 2.4: **How the plot of prediction error vs. time changes as learning takes place from trial to trial.** A reward is given at $t = 5$ and a light goes on at $t = 2$. We can see from the positive prediction error δ that on trial 0 our agent is surprised by the reward at $t = 5$. On the next trial, the agent has learned to some degree to expect the reward, so its surprise is less. As the trials continue, the agent starts to associate the reward with its memory of the light going on. By trial 7, the agent's prediction error is non-zero *before* the reward is received; this non-zero value keeps propagating backwards in time until on trial 20 a prediction error the same size as the original response to the reward occurs in response to the light being switched on. It is as though the light itself is now serving as a reward. This diagram was generated using learning rate $\alpha = 0.5$ and discounting factor $\gamma = 1$. See script td_pavlovian_small_example.py

Our previous algorithm used the prediction error δ at time t to update only the component of the weight vector $\mathbf{w}$ that matches the component of the stimulus vector $\mathbf{x}$ that was active at time t. Sometimes we want to let the prediction error at time t also influence components of the weight that match components of the stimulus vector that were active in the past. To do this, eligibility traces are helpful. We will not focus on them here, but there is an explanation of them in Appendix C.

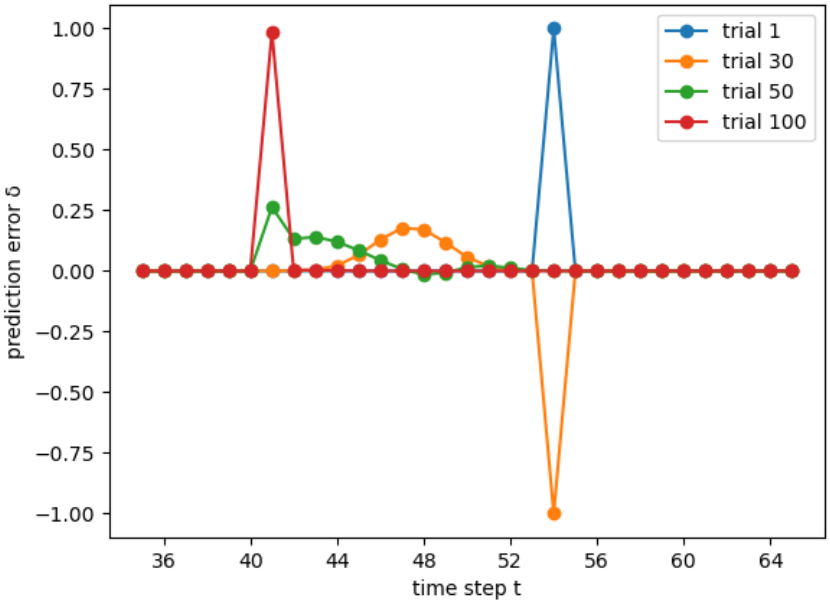


Figure 2.5: **How prediction error vs. time changes from trial to trial when reward is withheld on trial 30.** There are 120 time steps repeating over 100 trials. At $t = 40$ a light gets switched on. In all trials except the thirtieth, a reward arrives at $t = 54$. In this example, learning rate α is 0.3 and discount factor $\gamma = 1$.

2.12 Why use RW and not TD?

RW and variations of it are the update rules I have come across most frequently in my readings on fitting reinforcement learning models to behavioural data. I suspect that this is because RW is the simplest model we have; perhaps many behavioural tasks are simple enough that more complex models like temporal difference learning (section 2.10) don't make sense. It doesn't really make sense to use TD models for single-step tasks because the selling point for TD models is exactly the fact that they distinguish between time steps within trials. Now you might think, "What if I simply define a trial as 'showing 5 mushroom pictures in a row' - can't TD be helpful then?" I don't think it would, because there is no causal relationship between the successive pictures; the picture shown at time $t-1$ and the reward associated with it don't tell us anything about the picture and reward at time t.

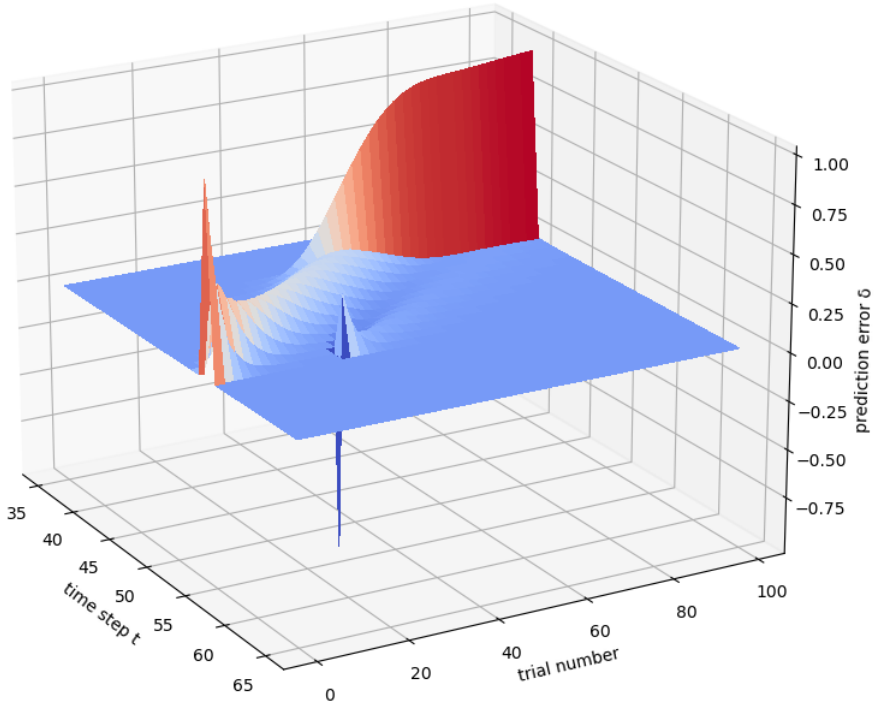


Figure 2.6: **How prediction error vs. time changes from trial to trial when reward is withheld on trial 30 (3D version).** See caption of figure 2.5.

2.13 Why Q-learning sometimes looks like Rescorla Wagner

This is the Q-learning update rule given in Sutton and Barto (2020):

$$Q_{t+1}(s_t, a_t) = Q_t(s_t, a_t) + \alpha(R_t + \gamma \text{max}_a Q_t(s_{t+1}, a) - Q_t(s_t, a_t)) \tag{2.12}$$

But frequently the following update rule gets referred to as Q-learning (Gershman, 2016; Guitart-Masip et al., 2012):

$$Q_{t+1}(s_t, a_t) = Q_t(s_t, a_t) + \alpha(R_t - Q_t(s_t, a_t)) \tag{2.13}$$

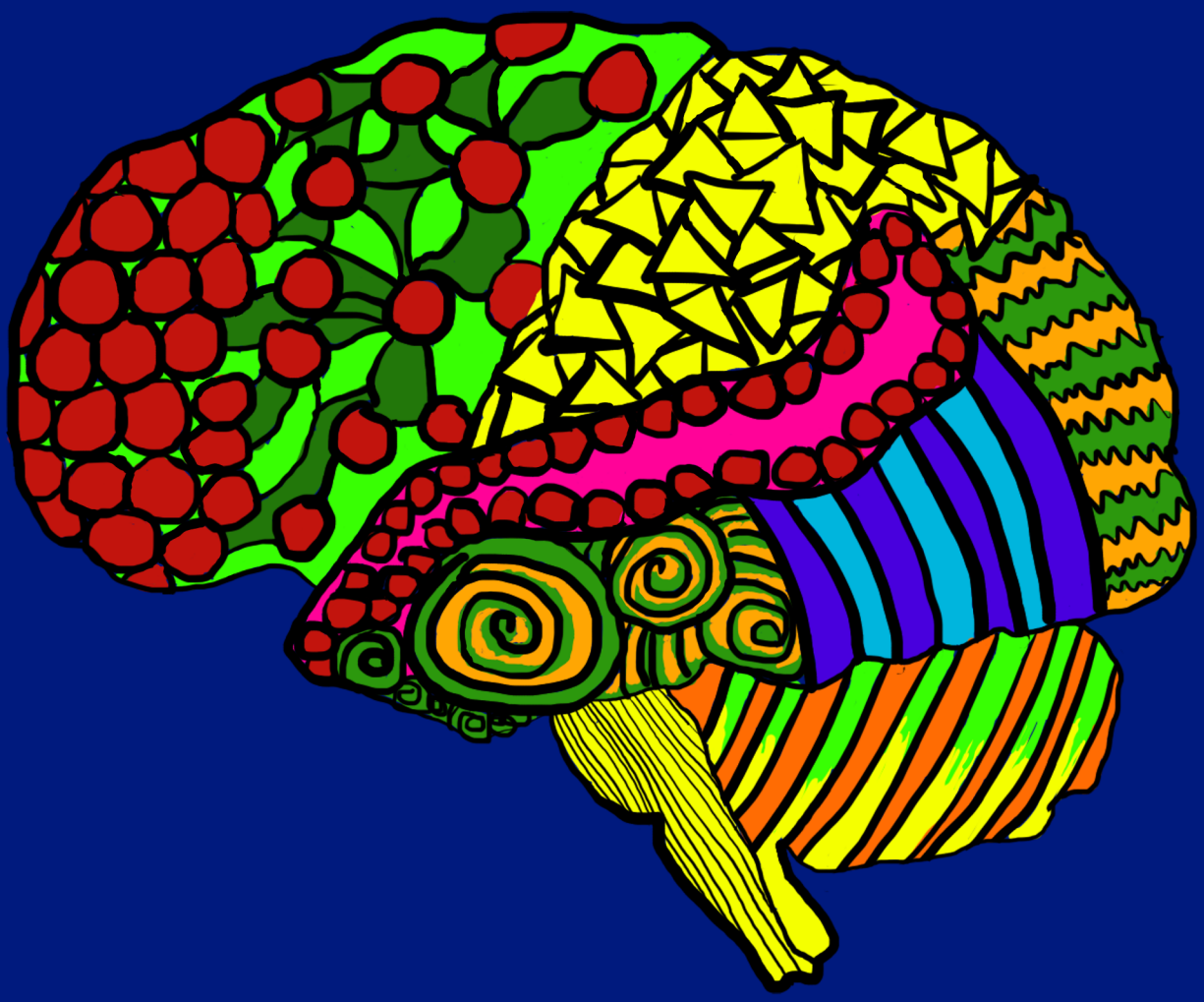
This might cause confusion because it looks just like the Rescorla Wagner update rule and not like the Q-learning rule above. But consider that the trials in Gershman (2016) and Guitart-Masip et al. (2012) have only one time step each. That means that state s_{t+1} refers to the terminal

state, which by convention always has a value of zero. So actually equation 2.12 reduces to exactly equation 2.13 when $s_{t+1} = 0$.

Now that we have surveyed some reinforcement learning algorithms, it is time to examine how they can be used to model data from behavioural experiments.

Chapter 3

Parameter estimation and model selection



The previous chapter discussed a number of reinforcement learning models and explored some ways an agent might select actions. Since we are interested in how these models apply to real data, I discuss next how one might go about choosing a model and how one can determine which parameter values make a given model fit the data best. I begin by giving an overview of this process.

3.1 Overview of the process

Imagine giving a friend a simple computer game to play. The game we will use in this example is the mushroom game from section 2.7, which is available at https://github.com/lizelleniit/lizellemasters/blob/master/mushroom_game/collect_mushroom_game_data.py. Your friend is shown a mushroom on the screen (this is the state), and they have to decide whether to eat it. If they choose to eat it, they press a key; if they decide not to eat it, they do nothing. They then receive a reward, punishment or neutral feedback from a distribution, where the distribution is determined by their choice. This state, action, reward/punishment sequence is repeated several times. Figure 3.1 shows a possible sequence of events in the game.

Next, imagine you are interested in predicting the way your friend will play this game in future. Maybe you want to scramble the pictures so that different reward distributions are attached to different pictures and predict how fast your friend will learn this time. A way to do this is to fit a reinforcement learning model to the data from their previous play of the game. Let us discuss exactly what this involves.

Your reinforcement learning model needs to have two parts: a learning model (also known as an update rule) and an observation function. I have already outlined some examples of these in chapter 2, but I will make the two parts explicit here. Figure 3.2 illustrates how these parts fit into the reinforcement learning process. The first part, the learning model (Daw, 2011), will specify how your friend might be keeping track of the values of the various mushrooms and the choices available to them. The second part, the observation function (Daw, 2011), specifies how the values (updated by the learning model) might be used to make choices. The observation function and the learning model are the ingredients you need to predict what choices your friend might make in future. Note, however, that even if you choose exactly the same learning model and observation function your friend is using, you will probably not be able to predict with complete accuracy what choices your friend will make. This is because there is very likely a probabilistic

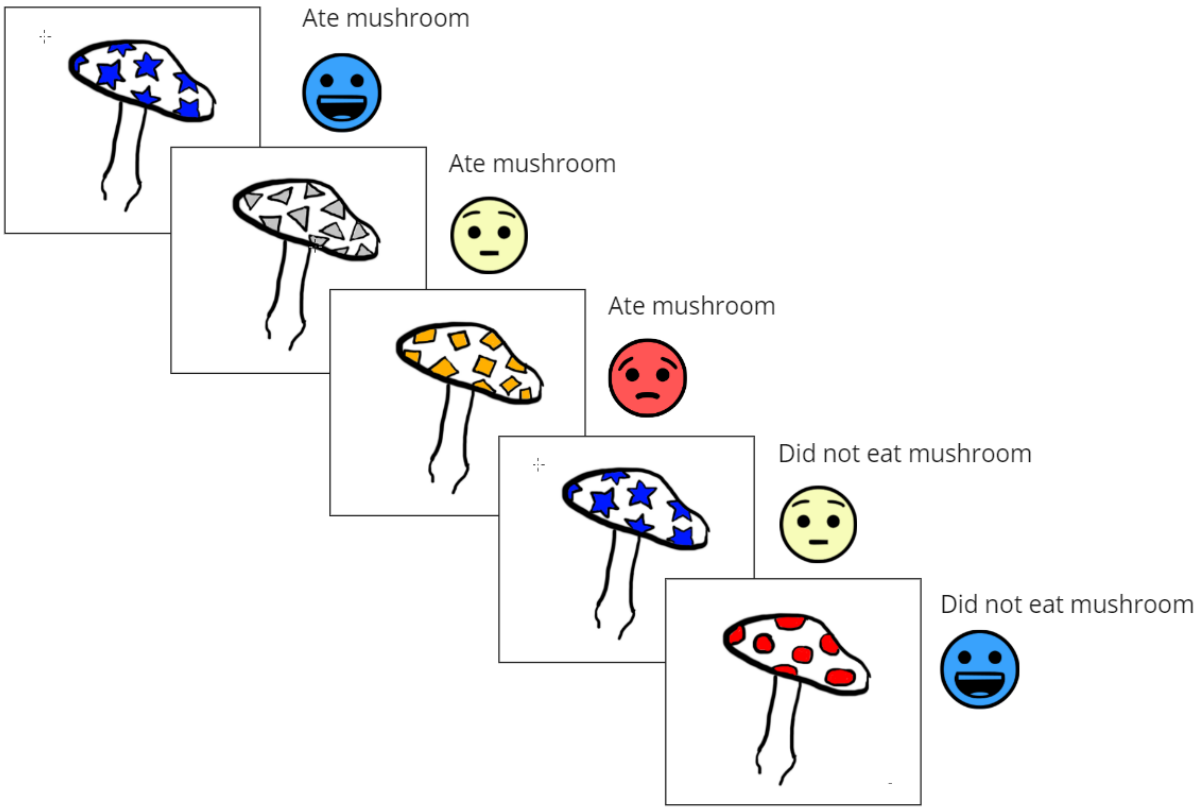


Figure 3.1: This figure shows a possible sequence of events in the mushroom game. The feedback is partly probabilistic. When eating a 'bad' mushroom, there is a 0.7 probability of a loss and a 0.3 probability of no reward. When eating a 'good' mushroom, there is a 0.7 probability of a reward and a 0.3 probability of no reward. Refraining from eating a mushroom results in neutral feedback.

component to your friend's learning model, your friend's observation function, and/or the way the environment responds.

There are two distinct processes involved in modelling data: choosing a model (model selection), and finding parameters for a model we have already selected (parameter estimation). You would need to attend to both of them in order to predict your friend's future behaviour. Before we dig into those, however, we need to discuss a sticky concept that is necessary for carrying out both those processes: the likelihood function.

The likelihood function describes the probability that we would observe a certain data set as a function of the parameters in our model. In other words, it helps us answer the question of how likely it is to see the data we are seeing given various parameter values. We need to be careful not to equate this with the probability that particular parameter values are correct given

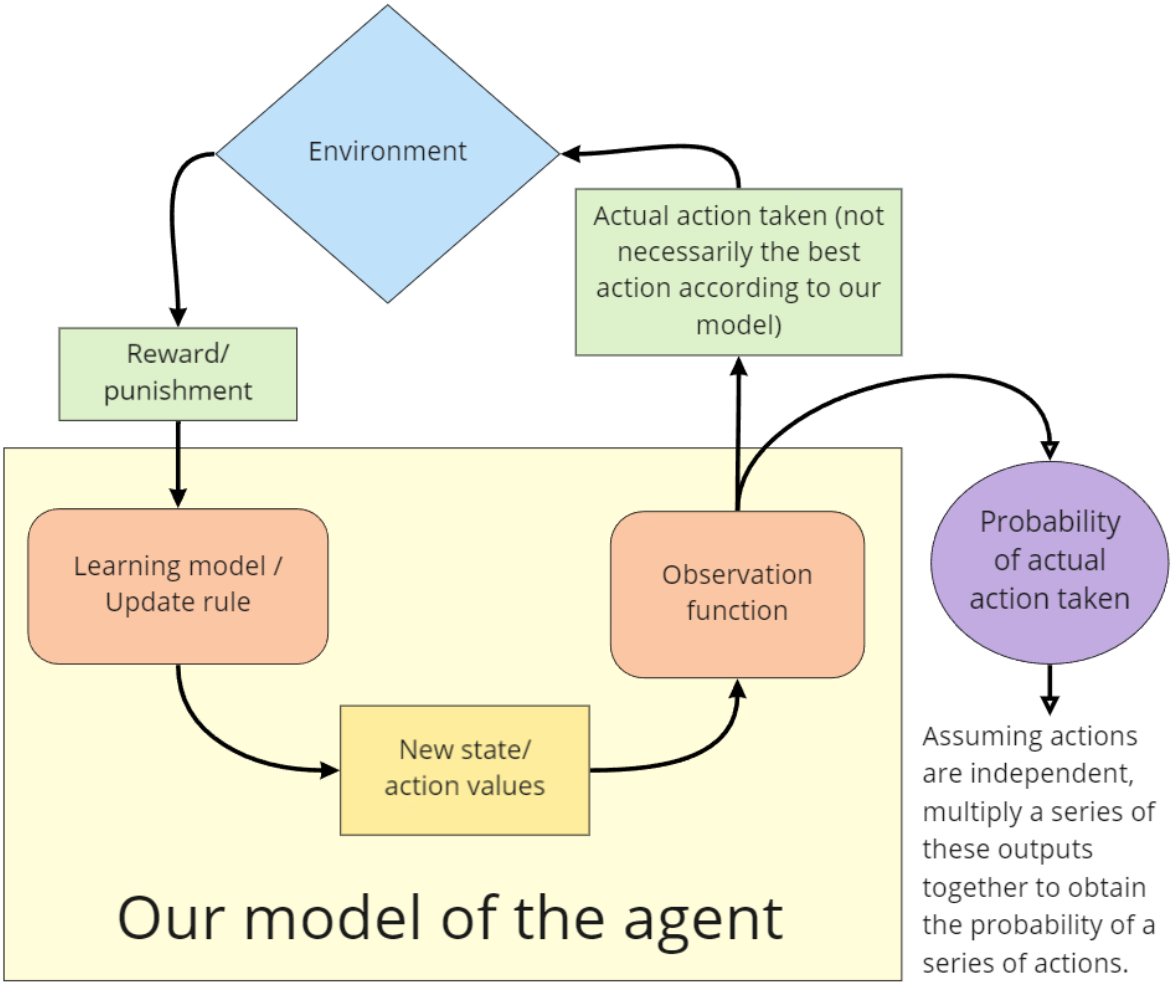


Figure 3.2: The observation function and learning model/update rule are both part of our model of the agent. To fit this model to data, we need to find the probability that the model would result in the chain of actions we have recorded. For each action, we calculate the probability that the model would choose that action given everything that has happened up to that point. Once we have probabilities for all the actions, we usually multiply those probabilities together under the assumption that the actions are independent.

the data. The difference lies in the order of the conditional probabilities: the likelihood describes the probability of the data given the parameters, while the other distribution (often called the posterior) describes the probability of the parameters given the data. The posterior distribution is often what we really want, but because it can be tricky to obtain, people sometimes settle for finding only the likelihood instead.

Now that the concept of the likelihood is hopefully clear, we can discuss model selection and parameter estimation. Model selection involves choosing a number of possible models $\mathcal{M}_j$ and then comparing them to determine which one fits the data set $\mathcal{A}_j$ the best. For example, you might think that both a plain Rescorla Wagner (RW) learning model and a RW learning model with separate learning rates for reward and punishment are feasible, so you fit the models to the data to determine which works best. Once we have chosen a model to fit to our data, we are interested in finding out which parameter values for that model best fit our data. Parameter estimation involves varying the parameters until we find the parameter values that cause the model to fit the data the best. We can choose among various ways of defining what we mean by 'best'. One such way is the maximum likelihood method, where we look for the parameter values that maximise the likelihood function. We might also use the maximum a posteriori method, which makes use of Bayes's formula. We still find the likelihood, but instead of maximising the likelihood, we first use Bayes's formula to find the posterior and then maximise the posterior. I first explain the parameter estimation process and then move on to model comparison.

3.2 Parameter estimation

To explain the parameter estimation process, I will explain how one would fit a simple model to data from a play of the mushroom game described above. For my learning model, I will use a Rescorla Wagner update rule,

$$Q_{t+1}(s_t, a_t) \leftarrow Q_t(s_t, a_t) + \alpha(r_t - Q_t(s_t, a_t)) \tag{3.1}$$

and for my observation function I will use a softmax distribution,

$$p(a_t|s_t) = \frac{e^{\beta Q_t(s_t,a_t)}}{\sum_i e^{\beta Q_t(s_t,a_i)}} \tag{3.2}$$

My free parameters in this case are the learning rate α and the inverse temperature β. For participant i, I collect α_i and β_i into a single vector $\mathbf{h}_i = \langle \alpha_i, \beta_i \rangle$. For each participant i,

parameter estimation will involve finding the values of α_i and β_i that best fit this model. We have two ways we can do this: through maximising likelihood or through maximum a posteriori estimation. I will discuss the maximum likelihood method first.

3.2.1 The maximum likelihood method

The maximum likelihood method involves finding the likelihood $L = p(\mathbf{A}_i|\mathbf{h}_i)$ for a range of parameters and choosing the parameters for which the likelihood is a maximum. The likelihood is the probability that we would observe the data we do observe, given the model we've chosen and certain parameter values for that model. In other words, the likelihood is the probability of the data given the learning model and observation function and particular parameter values. We find it by taking the product of the probabilities of a string of actions taken by the agent. The probabilities for the individual actions are given by the observation function, equation 3.2. The probability of each action is determined by the environment's response to the previous action because it depends on $Q_t(s_t, a_t)$, and $Q_t(s_t, a_t)$ gets updated according to the learning model (equation 3.1 in our example) after each action. We therefore need to update $p(a_t|s_t)$ and $Q_t(s_t, a_t)$ iteratively, as follows:

Algorithm 3: Finding the likelihood $L = p(\mathbf{A}_i|\mathbf{h}_i)$

Initialise $Q(S, A) = 0 \; \forall \; S, \; A$;
Initialise likelihood L to 1;
for *each action* A **do**
  $p(A|S) = \frac{e^{\beta Q(S,A)}}{\sum_i e^{\beta Q(S,A_i)}}$;
  $Q(S, A) \leftarrow Q(S, A) + \alpha(R - Q(S, A))$;
  $L \leftarrow L \times p(A|S)$;
end

Now we can take this likelihood and vary the parameters to see for which values the likelihood is the biggest. Those values then become our most likely estimates for the parameters.

3.2.2 Maximum a posteriori estimation

We may feel that the maximum likelihood method gives too much consideration to parameters that we know from prior experience are very unlikely. We can allow our prior assumptions about the parameters to inform the distribution we maximise by using Bayes's rule:

$$p(\mathbf{h}_i|\mathbf{A}_i) = \frac{p(\mathbf{A}_i|\mathbf{h}_i)p(\mathbf{h}_i|\boldsymbol{\theta})}{\int d\mathbf{h}_i p(\mathbf{A}_i|\mathbf{h}_i)p(\mathbf{h}_i|\boldsymbol{\theta})} \tag{3.3}$$

Above, I write $p(\mathbf{h}_i|\boldsymbol{\theta})$ instead of only $p(\mathbf{h}_i)$ because we are assuming that $\mathbf{h}_i$ is drawn from a distribution with parameters $\boldsymbol{\theta}$. I use $\mathbf{A}_i$ to denote the series of actions performed by subject i. It is important to remember that $\mathbf{h}_i$ is a vector - in my example of two parameters it is $\langle \alpha_i, \beta_i \rangle$ - so $\boldsymbol{\theta}$ contains all the parameters necessary to define the distributions these two parameters are drawn from.

Maximising equation 3.3 with respect to $\mathbf{h}_i$ is what we refer to as maximum a posteriori (MAP) estimation. In practice, because the denominator does not depend on $\mathbf{h}_i$ (it gets integrated out), for the purposes of maximisation we can ignore it and only maximise the numerator. This means that all we need in addition to the likelihood $p(\mathbf{A}_i|\mathbf{h}_i)$ is the prior $p(\mathbf{h}_i|\boldsymbol{\theta})$. The tricky part is coming up with priors. Techniques like expectation-maximisation come in useful in that regard; I will discuss this in the next section.

3.3 Finding priors

Sometimes it is useful to have the parameters of one distribution be drawn from another distribution. Models where this is the case are called hierarchical models. Huys, Cools, et al. (2011) use such a model: they draw their reinforcement learning parameters from prior Gaussian distributions. They refer to the parameters of these prior Gaussian distributions as hyperparameters. A challenging part of the model fitting process is deciding what these hyperparameters should be. To do this, we can employ what Mkrtchian et al. (2017) refer to as a 'hierarchical type II maximum likelihood expectation–maximization approach'. It was first described by Huys, Cools, et al. (2011) and has subsequently been used by Guitart-Masip et al. (2012), Huys (2016), Huys, Cools, et al. (2011), Huys, Moutoussis, et al. (2011), Huys et al. (2013), and Mkrtchian et al. (2017). This technique involves using expectation-maximisation (discussed below) to find the parameters $\boldsymbol{\theta}$ that maximise the joint likelihood $p(\mathcal{A}|\boldsymbol{\theta})$ of all the actions taken by all subjects. One then takes these values of $\boldsymbol{\theta}$ as the hyperparameters (parameters of the prior distributions).

$p(\mathcal{A}|\boldsymbol{\theta})$ can be written as the product of the probabilities of all the action sequences of all the subjects i to N given certain hyperparameters $\boldsymbol{\theta}$,

$$p(\mathcal{A}|\boldsymbol{\theta}) = \prod_{i=1}^{N} p(\mathbf{A}_i|\boldsymbol{\theta}), \tag{3.4}$$

Part of the difficulty in maximising the above expression lies in the fact that we know nothing about $p(\mathbf{A}_i|\boldsymbol{\theta})$. We do not know how the likelihood depends on $\boldsymbol{\theta}$; we only have knowledge about $p(\mathbf{A}_i|\mathbf{h}_i)$. Therefore it is useful to rewrite $p(\mathcal{A}|\boldsymbol{\theta})$ in a more usable form, using $\mathbf{h}_i$ as an intermediate variable between the sequence of actions $\mathbf{A}_i$ and the hyperparameters $\boldsymbol{\theta}$ of the distribution that $\mathbf{h}_i$ is drawn from. We integrate over $\mathbf{h}$ to get $p(\mathbf{A}_i|\boldsymbol{\theta})$ for subject i:

$$\begin{aligned} p(\mathbf{A}_i|\boldsymbol{\theta}) &= \int d\mathbf{h_i}\ p(\mathbf{A}_i, \mathbf{h_i}|\boldsymbol{\theta}) \\ &= \int d\mathbf{h_i}\ p(\mathbf{A}_i|\mathbf{h_i}) p(\mathbf{h_i}|\boldsymbol{\theta}) \end{aligned}$$

Equation 3.4 therefore becomes

$$p(\mathcal{A}|\boldsymbol{\theta}) = \prod_{i=1}^{N} \int d\mathbf{h_i}\ p(\mathbf{A}_i|\mathbf{h_i}) p(\mathbf{h_i}|\boldsymbol{\theta}) \tag{3.5}$$

Notice that now our equation requires $p(\mathbf{A}_i|\mathbf{h}_i)$ and $p(\mathbf{h}_i|\boldsymbol{\theta})$. These are easier to obtain than the original $p(\mathbf{A}_i|\boldsymbol{\theta})$. $p(\mathbf{A}_i|\mathbf{h}_i)$ is the likelihood we obtain by multiplying together the probabilities of all the actions a particular subject took in our experiment, given the reinforcement learning parameters $\mathbf{h}_i$. We assume each reinforcement learning parameter has its own independent Gaussian prior[1], with the parameters for that Gaussian given by the relevant components of the matrix $\boldsymbol{\theta}$, so for $\mathbf{h}_i = \langle \alpha_i, \beta_i \rangle$ we have

$$p(\mathbf{h}_i|\boldsymbol{\theta}) = p(\alpha_i|\boldsymbol{\theta}_\alpha) p(\beta_i|\boldsymbol{\theta}_\beta)$$

We now want to maximise equation 3.5 with respect to $\boldsymbol{\theta}$ to find $\hat{\boldsymbol{\theta}}$:

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\text{argmax}} \left(\prod_{i=1}^{N} \int d\mathbf{h_i}\ p(\mathbf{A}_i|\mathbf{h_i}) p(\mathbf{h_i}|\boldsymbol{\theta}) \right) \tag{3.6}$$

[1] See the answer by user conjugateprior (https://stats.stackexchange.com/users/1739/conjugateprior) (2018), available at https://stats.stackexchange.com/q/329207.

Unfortunately, this product of integrals is not trivial to obtain. Firstly, there is one integral for every subject, so any computation would take about N times as long as it would for a single subject. Secondly, there may be multiple components (parameters) in $\mathbf{h}_i$, and each one adds a dimension to the integral. Lastly, we need to remember that $p(\mathbf{A}_i|\mathbf{h}_i)$ is in fact itself a product of probabilities of all the individual actions in the sequence of actions $\mathbf{A}_i$, and that each action depends on the past in a way determined by the learning model and observation function. All of this needs to be calculated, which results in the process taking too long to be feasible in all but the simplest cases.

3.3.1 Expectation-maximisation

According to Huys, Cools, et al. (2011), expectation-maximisation is one way we can do this maximisation process more efficiently. But how do we know expectation-maximisation can be useful here, and how do we apply it? Let us start by discussing what expectation-maximisation is.

Y. Chen and Gupta (2010) point out that the purpose of expectation-maximisation is to find a set of parameters that maximise a likelihood. Since maximising a likelihood is exactly what we want to do, this already suggests that expectation-maximisation can be useful. Expectation-maximisation gets used instead of ordinary maximum likelihood estimation when we have incomplete data available. Usually this takes the form of a hidden variable, a variable whose value we do not have access to (Russell & Norvig, 2010, p.816). This is the case in our situation because we do not know $\mathbf{h}_i$.

Our ingredients for the expectation-maximisation (EM) process are observed data, hidden variables, and parameters for our probability model. In the case of our example, the observed data are the action sequences $\mathcal{A}$ (the collection of $\mathbf{A}_i$ for all participants i) we observe for the participants in our task. The hidden variables are the collection of parameter values $\mathcal{H}$ for the reinforcement learning model. $\mathcal{H}$ consists of $\mathbf{h}_i$ for all participants i, and in our example each $\mathbf{h}_i$ consists of learning rate α_i and inverse temperature β_i. The parameters for the probability model are the means and variances of the Gaussians we assume that α_i and β_i are drawn from; these means and variances are collectively denoted by $\boldsymbol{\theta}$. Using these variables, and letting (n) indicate the number of iterations that have been completed, the general form of the EM algorithm can be written as

$$\hat{\boldsymbol{\theta}}^{(n+1)} = \underset{\boldsymbol{\theta}}{\operatorname{argmax}} \left(\int d\mathcal{H}\ p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) \right) \tag{3.7}$$

The integral above calculates the expectation value of the log likelihood $\log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta})$, with respect to the probability distribution $p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)})$. Let me unpack what that means. We can think of an expectation value $\mathbb{E}$ as a kind of weighted average. It is usually defined for a continuous variable X with probability density function $f(X)$ in the following way (Ross, 2009, p.190):

$$\mathbb{E}(x) = \int_{\infty}^{\infty} x f(x) dx$$

For a function $g(x)$ (instead of just a variable x), it is defined as

$$\mathbb{E}(g(x)) = \int_{\infty}^{\infty} g(x) f(x) dx$$

It is quite a jump to go from a single variable to the collection of all parameters for all participants $\mathcal{H}$, but for now let us assume that we can also say the following:

$$\mathbb{E}(\mathcal{H}) = \int g(\mathcal{H}) f(\mathcal{H}) \; d\mathcal{H}$$

If we then let

$$g(\mathcal{H}) = \log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta})$$

and

$$f(\mathcal{H}) = p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}),$$

our expectation value becomes

$$\mathbb{E}(\mathcal{H}) = \int d\mathcal{H} \; p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) \tag{3.8}$$

like in equation 3.7. Note that $p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)})$ is the product of probabilities $\prod_{i=1}^{N} p(\mathbf{h}_i|\mathbf{A}_i, \boldsymbol{\theta}^{(n)})$ and $p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta})$ is the product $\prod_{i=1}^{N} p(\mathbf{A}_i, \mathbf{h}_i|\boldsymbol{\theta})$. Here we are assuming that each subject is independent of each other subject.

The E step, or expectation step, consists of doing the integral in equation 3.8. The M step, or maximisation step, involves varying the parameters $\boldsymbol{\theta}$ for the probability model to find the values for which the integral just mentioned is a maximum.

A more rigorous derivation for equations 3.8 and 3.7 is in Appendix A, section A.1. To get to equation 3.7 from the starting point of wanting to maximise equation 3.5, I followed the derivation in Dellaert (2002), but starting with $p(\mathcal{A}|\boldsymbol{\theta})$ instead of $p(\mathcal{A}, \boldsymbol{\theta})$ like they did.

3.3.2 Following the expectation-maximisation process for our example

To perform the maximisation in equation 3.7, we differentiate the product of integrals in that equation and set it equal to zero. The details of this derivation are in Appendix A, section A.2. The equations resulting from this process match those given by Huys, Cools, et al. (2011):

$$\boldsymbol{\mu}^{(n)} = \frac{1}{N}\sum_i \mathbf{m}_i^{(n)} \tag{3.9}$$

$$(\boldsymbol{\nu}^{(n)})^2 = \frac{1}{N}\sum_i \left[(\mathbf{m}_i^{(n)})^2 + \boldsymbol{\Sigma}_i^{(n)}\right] - (\boldsymbol{\mu}^{(n)})^2 \tag{3.10}$$

Above, $\boldsymbol{\mu}$ and $\boldsymbol{\nu}^2$ represent the components of $\boldsymbol{\theta}$ that specify the means and variances of the prior Gaussians respectively. Each is a vector with number of elements equal to the length of the parameter vector $\mathbf{h}_i$. $\mathbf{m}_i$ is a vector where the elements represent the values for which the numerator of our posterior distribution (see equation 3.11 later) is a maximum. The trickiest parameter in this equation is $\boldsymbol{\Sigma}_i$. It quantifies the degree of 'spread' in the posterior distribution. For a single free parameter, it is simply the variance of the posterior distribution, but for more than one parameter it becomes a covariance matrix. I haven't fully figured out the mathematical details here; for example, it is not possible to add a vector ($\mathbf{m}_i$) to a matrix ($\boldsymbol{\Sigma}_i$), so something is still wrong or missing in this explanation. Unfortunately even after a long search of the literature, I have not found clarity on this issue.

To apply the EM process in our particular example, we start by choosing arbitrary hyperparameters. We then repeat the following process as many times as necessary, each time using updated hyperparameters:

We start by guessing arbitrary hyperparameters $\boldsymbol{\theta}$. We then need to use these arbitrary values to evaluate the expectation value of $\log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta})$, but luckily we have already done some of the work by deriving equations 3.9 and 3.10. All we need to do is find new values of $\mathbf{m}_i$ and $\boldsymbol{\Sigma}_i$ to plug into these equations. We do this by finding the numerator of the posterior 3.3 for each participant i,

$$q(\mathbf{h}_i) = p(\mathbf{A}_i|\mathbf{h}_i)p(\mathbf{h}_i|\boldsymbol{\theta}) \tag{3.11}$$

.

We assume that expression 3.11 is well-approximated by a multidimensional normal distribution. The parameter values $\mathbf{h}_i$ for which expression 3.11 is a maximum become $\mathbf{m}_i$, our new

estimate for $\mathbf{h}_i$ and the mean of the above-mentioned normal distribution. We also need to find the covariance of this normal distribution, which is related in the following way to the Hessian[2] of expression 3.11 evaluated at the maximum:

$$\mathbf{\Sigma} = -\mathbf{H}^{-1}|_{\mathbf{h}_i=\mathbf{m}_i}$$

The above result is derived in Appendix B.

If our reinforcement learning model has only a single free parameter h_i, this implies that

$$\Sigma = \frac{1}{-\frac{\partial^2}{\partial h_i^2} \log p(h_i|\theta)|_{h_i=m_i}}$$

Once we have found $\mathbf{m}_i$ and $\mathbf{\Sigma}_i$ for a particular $\boldsymbol{\theta}$, we can use update equations 3.9 and 3.10 to find updated estimates for $\boldsymbol{\theta}$. We then use these new estimates to repeat this process of maximising the posterior and finding its covariance matrix and find even better estimates for $\boldsymbol{\theta}$. These estimates are guaranteed not to get worse, but that does not mean they are optimal; the algorithm may simply have found a local maximum in the likelihood (Y. Chen & Gupta, 2010). To avoid these local maxima, Y. Chen and Gupta (2010) advise running one's code with multiple starting guesses for $\boldsymbol{\theta}$ and choosing the one that yields the highest likelihood.

Finally, once we have obtained good estimates for $\boldsymbol{\theta}$, we can use those estimates to do a final maximisation of the posterior for each participant i in order to find the best estimates for $\mathbf{h}_i$.

The above explanation was concerned with how to get estimates for reinforcement learning parameters in a particular model. However, often we want to consider several possible models in order to find the one that fits the data the best. The next section discusses how to do that.

3.4 Model comparison

There are multiple possible models to choose from when fitting reinforcement learning models to behavioural data. We need to have a systematic way to decide which model fits our data the best.

We start by considering the posterior probability of a model $\mathcal{M}$ given the data set $\mathcal{A}$. By Bayes' formula,

$$p(\mathcal{M}|\mathcal{A}) = \frac{p(\mathcal{A}|\mathcal{M})p(\mathcal{M})}{\sum_{\mathcal{M}'} p(\mathcal{A}|\mathcal{M}')p(\mathcal{M}')}$$

[2] A matrix of partial second derivatives of a scalar function.

If we are interested in comparing two models $\mathcal{M}_1$ and $\mathcal{M}_2$, we can examine the ratio of their posterior probabilities (Kruschke, 2014, p.268):

$$\frac{p(\mathcal{M}_1|\mathcal{A})}{p(\mathcal{M}_2|\mathcal{A})} = \frac{p(\mathcal{A}|\mathcal{M}_1)p(\mathcal{M}_1)}{p(\mathcal{A}|\mathcal{M}_2)p(\mathcal{M}_2)}$$

The ratio $p(\mathcal{A}|\mathcal{M}_1)/p(\mathcal{A}|\mathcal{M}_2)$ is known as the Bayes factor and often gets used to decide which model fits the data better. A value of this ratio larger than 1 counts in favour of $\mathcal{M}_1$, while a value smaller than 1 counts in favour of $\mathcal{M}_2$.

To calculate $p(\mathcal{A}|\mathcal{M})$, we need to integrate over all possible values of all parameters in our model:

$$p(\mathcal{A}|\mathcal{M}) = \int d\mathcal{H} p(\mathcal{A}|\mathcal{H}, \mathcal{M}) p(\mathcal{H}|\mathcal{M})$$

Since computing these integrals can be very difficult, Huys, Cools, et al. (2011) chose to approximate $\log p(\mathcal{A}|\mathcal{M})$ using the Bayesian Information Criterion (BIC):

$$\log p(\mathcal{A}|\mathcal{M}) \approx -\frac{1}{2}\text{BIC} = \log p(\mathcal{A}|\hat{\boldsymbol{\theta}}) - \frac{1}{2}|\mathcal{M}|\log(|\mathcal{A}|) \tag{3.12}$$

where $p(\mathcal{A}|\hat{\boldsymbol{\theta}})$ is the maximum of $p(\mathcal{A}|\boldsymbol{\theta})$, $|\mathcal{M}|$ is the number of prior parameters in model $\mathcal{M}$ (keeping in mind that each reinforcement learning parameter has two hyperparameters - a mean and a variance) and $|\mathcal{A}|$ is the total number of data points (including all actions by all subjects).

The second term penalises models with a large number of parameters. This is because a model with more parameters is generally expected to perform better as it has more freedom. One way to understand this is to imagine that a model with more parameters has more "levers" one can adjust to fit the data as well as possible. This does not necessarily result in a better model, since a model with too many parameters might fit a specific data set very well but not generalise to other data sets. This phenomenon is called 'over-fitting'.

Let us now look more closely at the first parameter in equation 3.12, $\log p(\mathcal{A}|\hat{\boldsymbol{\theta}})$. Strictly speaking, it needs to be calculated as follows:

$$\log p(\mathcal{A}|\hat{\boldsymbol{\theta}}) = \log \prod_i \int d\mathbf{h}_i p(\mathbf{A}_i|\mathbf{h}_i) p(\mathbf{h}_i|\hat{\boldsymbol{\theta}})$$

where i indexes the participants. Since this is difficult too, Huys, Cools, et al. (2011) instead approximated $\int d\mathbf{h}_i p(\mathbf{A}_i|\mathbf{h}_i) p(\mathbf{h}_i|\hat{\boldsymbol{\theta}})$ by sampling K sets of reinforcement learning parameters

$\mathbf{h}_{\text{sampled}}$ from the prior distributions characterised by $\boldsymbol{\theta}$, calculating $p(\mathbf{A}_i|\mathbf{h}_{\text{sampled}})$, and finding the mean. The result was that they approximated $\log p(\mathcal{A}|\hat{\boldsymbol{\theta}})$ as follows:

$$\log p(\mathcal{A}|\hat{\boldsymbol{\theta}}) \approx \sum_i \log \frac{1}{K} \sum_{k=1}^{K} p(\mathbf{A}_i|\mathbf{h}_{\text{sampled},k})$$

where K is the number of sets of reinforcement learning parameters

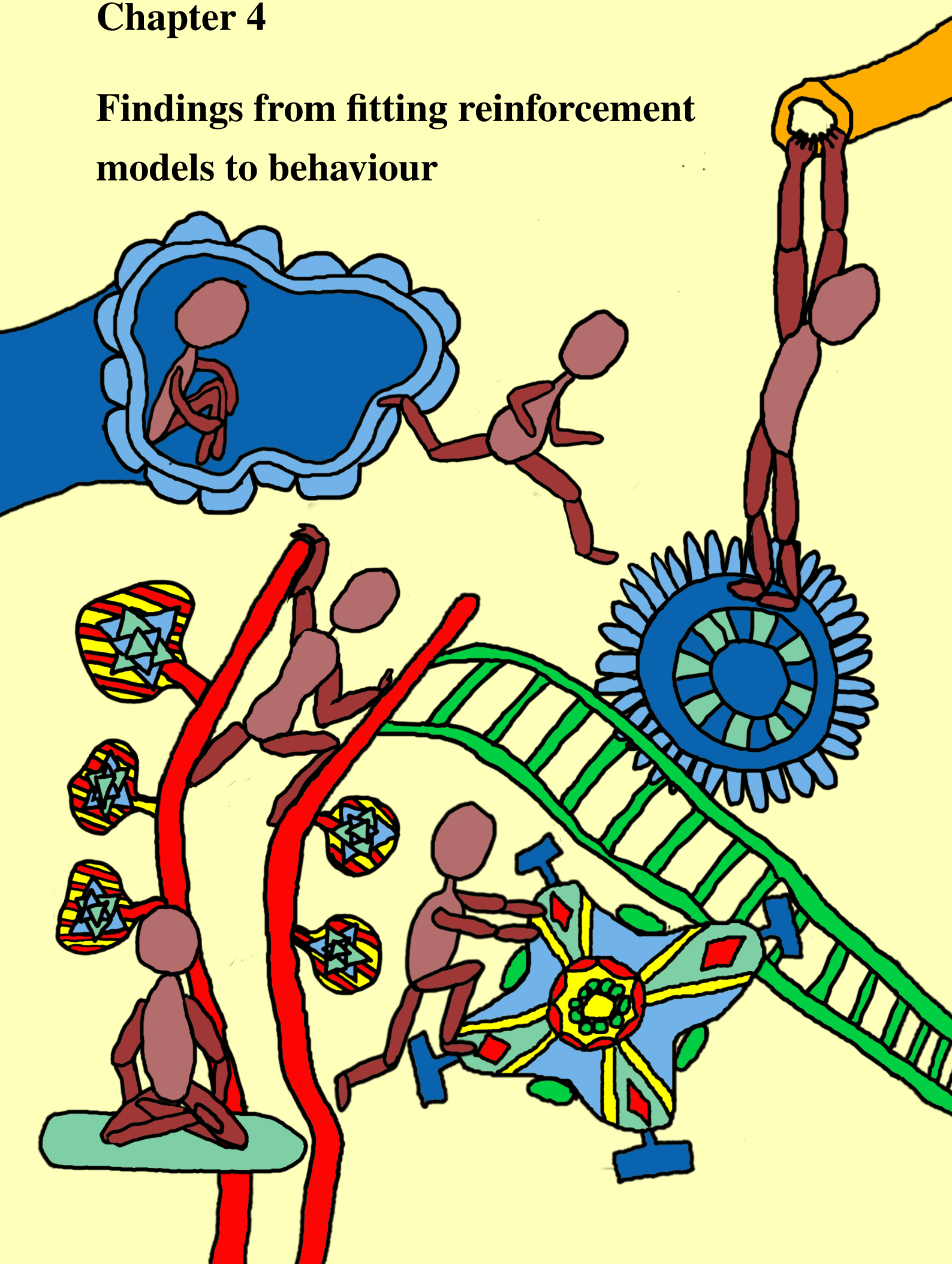
The lower the BIC score calculated using this method, the better the model was assumed to be.

To recap, we have seen that we can use parameter estimation to figure out which parameters are most probable given a data set and a model. To do this, ideally we should maximise the posterior distribution, but this entails coming up with prior distributions. Expectation-maximisation is one way to obtain these. It involves guessing hyperparameters for the prior distributions and using those guesses to calculate new guesses, which then get used to calculate even better guesses, and so on. During model comparison, we want to compare the posterior probabilities of different models given the data. A necessary ingredient for this is the likelihood (the probability of the data given the model), and this can be difficult to calculate. Instead, therefore, people often use approximations like the BIC.

At this point, we have a number of reinforcement learning models in our toolbox, and this chapter has discussed how to choose among those models and how to estimate parameters for a given model. This puts us in a position to understand studies where the authors have fitted reinforcement learning models to behavioural data. Such studies can help us answer some of the questions raised in the introduction, such as which reinforcement learning models explain behavioural data the best, and how their parameters differ between people with and without mental illness. We now look at some results from such studies.

Chapter 4

Findings from fitting reinforcement models to behaviour



In chapter 2, I discussed the fact that animals (including humans) learn from interacting with their environments. Reinforcement learning models, discussed in chapter 2, can in principle be used to model this learning process using the techniques in chapter 3. One can give people a behavioural task to do and fit models to the choices they make. By doing this many times, researchers have been able to uncover patterns in the way people learn. We as humans have much in common when it comes to how we learn and make decisions, and this chapter explores some of those commonalities. At the same time, one can uncover differences in the learning process from person to person and within the same person over time. It is becoming increasingly clear that those differences in learning are tied to other differences we care about, for example differences in emotional state. When one groups people according to various aspects of their psychological functioning and compares the ways the groups learn, one finds trends that suggest relationships between the group traits and aspects of learning. This chapter examines what studies have found regarding this. My goal for this chapter is to lay the groundwork for a discussion (see chapter 5) of how such findings can be used to think in a fresh way about emotional difficulties and mental illness.

Since this chapter is going to make frequent reference to depressive disorders and a related concept called 'anhedonia', I start by briefly clarifying what I mean by them.

4.1 Introduction to depression and anhedonia

I pointed out in the introduction that depressive disorders are mental disorders that have a disturbance in mood at their core. To repeat briefly what I said there, major depressive disorder (MDD) is a depressive disorder that involves depressed mood or anhedonia (reduced interest or pleasure in previously enjoyed activities), as well as other symptoms like reduced or increased appetite, sleep disturbances, difficulty concentrating, feelings of worthlessness, and suicidal thoughts and behaviour. Depressed mood and anhedonia are considered key symptoms of MDD in the DSM-5; at least one must be present for a diagnosis. For a diagnosis to be made, symptoms must also be present for at least two weeks.

Symptoms can be confusing to make sense of, however, because they're simply descriptions of someone's behaviour and experience. They don't often relate in an obvious way to an underlying disease process, so we somehow need to bridge the gap between symptoms and their causes. The concept of endophenotypes is one tool that can serve that purpose (Gottesman & Gould, 2003).

An endophenotype should meet the following criteria (Gottesman & Gould, 2003; Pizzagalli, 2014):

1. Associated with illness.
2. Heritable.
3. Consistently present in an individual whether disease is present or not
4. Cosegregation (occurs more frequently in family members with the disease than those without)

Endophenotypes are more closely related to biological and environmental influences than syndromes such as depression, and there is hope that studying endophenotypes may help us understand psychiatric disorders better than studying the syndromes described by current classification systems (Pizzagalli, 2014). The study of MDD may particularly benefit from such an approach because MDD probably consists of a collection of different pathological processes (Pizzagalli, 2014).

One proposed endophenotype for depression is anhedonia (Pizzagalli, 2014). Anhedonia is defined as 'a loss of interest or pleasure' in activities a person would usually enjoy (Robinson & Chase, 2017). It does not only appear in MDD; it is also a symptom of schizophrenia and addiction (Franken et al., 2007). The fact that it crosses conventional disease boundaries, suggests that understanding it could advance our understanding of more than one disorder. Another advantage to studying anhedonia is that it can be measured in standardised ways, such as using the relevant subscore of the Mood and Anxiety Symptom Questionnaire (MASQ).

Since anhedonia is defined as a loss of interest or pleasure, it is natural to expect that it would be related to how individuals learn from and respond to rewards. It is indeed the case that anhedonia is correlated with at least one reinforcement learning parameter, namely reward sensitivity; see section 4.3 of this chapter for more. Perhaps also to be expected is the fact that the correlation between anhedonia and reward sensitivity in reinforcement learning tasks is stronger than that between MDD and reward sensitivity. MDD is a multi-faceted disorder, so it makes sense that results of a task that addresses only one aspect of the disorder (response to reward and punishment) will be more correlated with that aspect alone than with the diagnosis as a whole. This is obvious but potentially useful: it emphasises the idea that we should be studying aspects of disorders that do correlate well with quantities that have potential explanatory power.

This chapter is about various reinforcement learning parameters that have been studied in MDD and anhedonia. Those parameters include learning rate parameters, reward/punishment sensitivity parameters, and ones related to a concept known as Pavlovian bias. I will start by telling you more about Pavlovian bias and Pavlovian-instrumental transfer, since these concepts may be unfamiliar to you if you are not from a psychology background.

4.2 Pavlovian bias and Pavlovian-instrumental transfer

To start our discussion of Pavlovian bias, we take a dive into what is referred to as 'Pavlovian-instrumental transfer', the idea that sometimes Pavlovian biases influence the instrumental choices people make. Before getting into the details of this phenomenon, let me introduce you to two experiments that have been used to examine this phenomenon. These are the mushroom+fractal go/nogo task first described by Huys, Cools, et al. (2011) and the fractal go/nogo task first described by Guitart-Masip et al. (2011). Here follows a description of the mushroom+fractal go/nogo task.

4.2.1 Mushroom+fractal go/nogo task (Huys, Cools, et al., 2011)

This task is divided into two blocks, an approach block and a withdrawal block, and (as shown in figure 4.1) each block has an instrumental training section, a Pavlovian training section, a Pavlovian query section, and a Pavlovian-instrumental transfer (PIT) section. PIT will be explained in section 4.2.3.

In the instrumental training section of each block, subjects are shown one of six mushrooms at a time. Each block is allocated its own six mushrooms; mushrooms do not overlap between blocks. In the approach block, participants are shown a mushroom inside a box (see the top part of panel A in figure 4.1). They are asked to click on the mushroom (make a 'go' decision) when they think this might lead to a reward and to do nothing (make a 'nogo' decision) when they think not approaching it might be the most rewarding course of action. In the avoidance block, participants are asked to click away from (or click and drag to throw away) a mushroom when they think withdrawal is the best action to take ('go') and to do nothing ('nogo') when they think not withdrawing is the best option to take. The 'click away from' option in the withdrawal block is depicted in the middle part of panel A of figure 4.1, while the 'throw away' option is

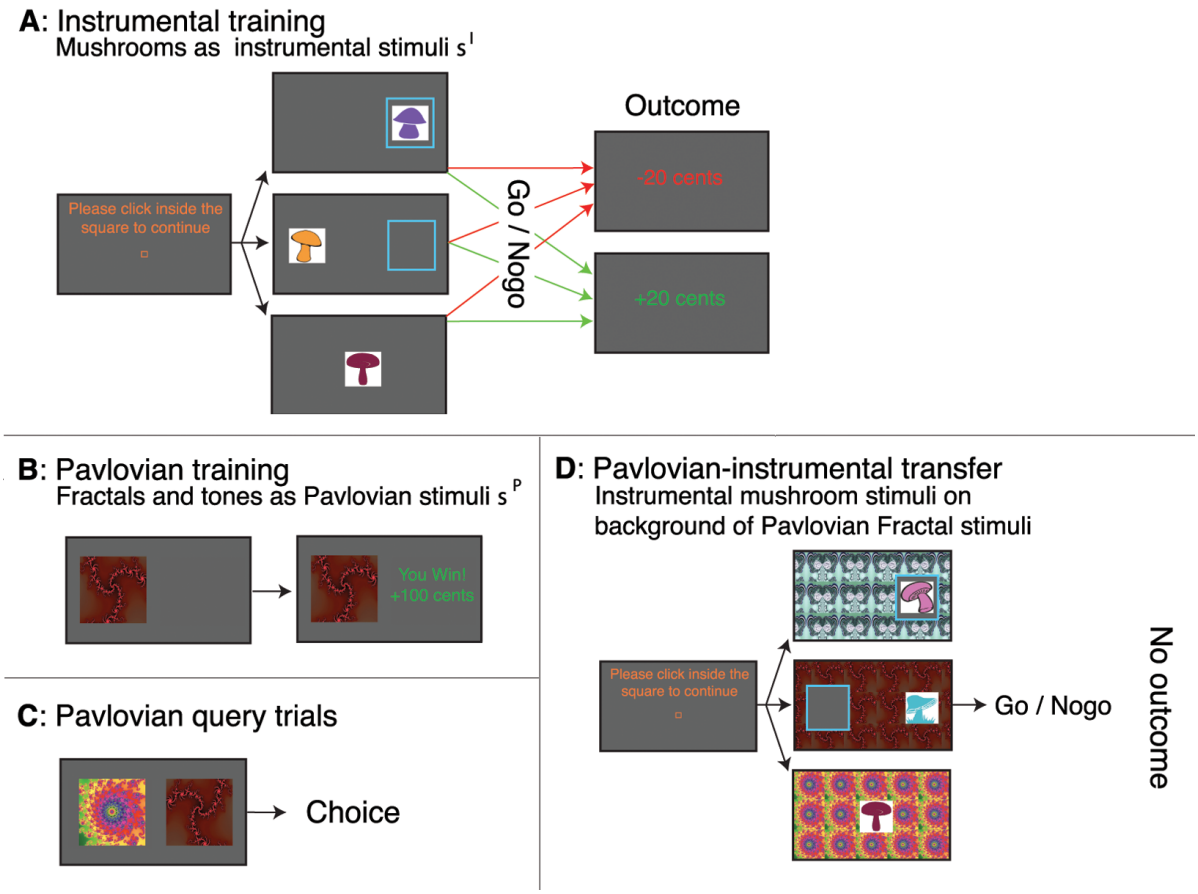


Figure 4.1: **Depiction of the mushroom+fractal go/nogo task** This image has been modified from Huys, Cools, et al. (2011) and shows the task they gave to their participants. **Panel A:** The top part of panel A shows what a participant might see during the instrumental training part of the approach block. Here they are being asked either to click on the mushroom or ignore it. The middle and bottom parts of panel A show what a participant might see during the instrumental training part of the withdrawal block. In the middle part they are being asked either to click away from the mushroom or ignore it, while in the bottom part they are being asked either to drag the mushroom off the screen or ignore it. **Panel B:** Participants passively watch as fractal images appear on the screen and are followed by rewards or punishments. **Panel C:** Participants are asked to choose which of two fractals is the 'best'. **Panel D:** Participants are asked to perform the same task as in panel A, but now the fractal images from panel B are tiled in the background. They also no longer receive feedback on their decisions.

depicted at the bottom of panel A.[1] In the instrumental section of each block, for three of the mushrooms, go is the correct action and for the other three, nogo is the correct action. Correct actions have a 70% chance of being rewarded and a 30% chance of being punished. Incorrect actions have a 70% chance of being punished and a 30% chance of being rewarded. Both the approach and withdrawal block have a Pavlovian training section (panel B in figure 4.1), where participants passively watch as fractal images (from a selection of five) are randomly displayed on the screen one at a time, each image followed one second later by reward or punishment. Each block also has a Pavlovian query section (panel C in figure 4.1), where participants are quizzed on the comparative values of Pavlovian images to check that they paid attention during the Pavlovian training section. Finally, panel D of figure 4.1 shows the PIT section of the task. In the Pavlovian-instrumental transfer section of each block, Huys, Cools, et al. (2011) combined the mushroom images with the fractal images; the fractals were displayed as tiled backgrounds to the mushrooms. This allowed them to study the effect the previously learned associations with the fractal background images would have on participants' decisions about selecting or rejecting mushrooms. No feedback was provided in this section. The findings of this study are discussed in section 4.2.3.

Another important task in the literature is the fractal go/nogo task, described below.

4.2.2 Fractal go/nogo task (Guitart-Masip et al., 2011)

This task was originally described by Guitart-Masip et al. (2011) and subsequently used by Guitart-Masip et al. (2012) and Mkrtchian et al. (2017). Fractal images (selected from a pool of four) are shown on a screen one at a time. Each fractal is followed by a target detection task, which involves either selecting the side of the screen a circle is on or doing nothing. Pressing a key to indicate which side the circle is on is counted as a 'go' response and doing nothing is counted as a 'nogo' response.

Each image corresponds to one of the following conditions (see figure 4.2):

- 'Go' to get reward
- 'Go' to avoid penalty

[1] For our purposes, we can consider the 'click away from' and 'throw away' options to be equivalent. Huys, Cools, et al. (2011) investigated whether these two modes of rejecting mushrooms would lead to different results, and found that they were equivalent.

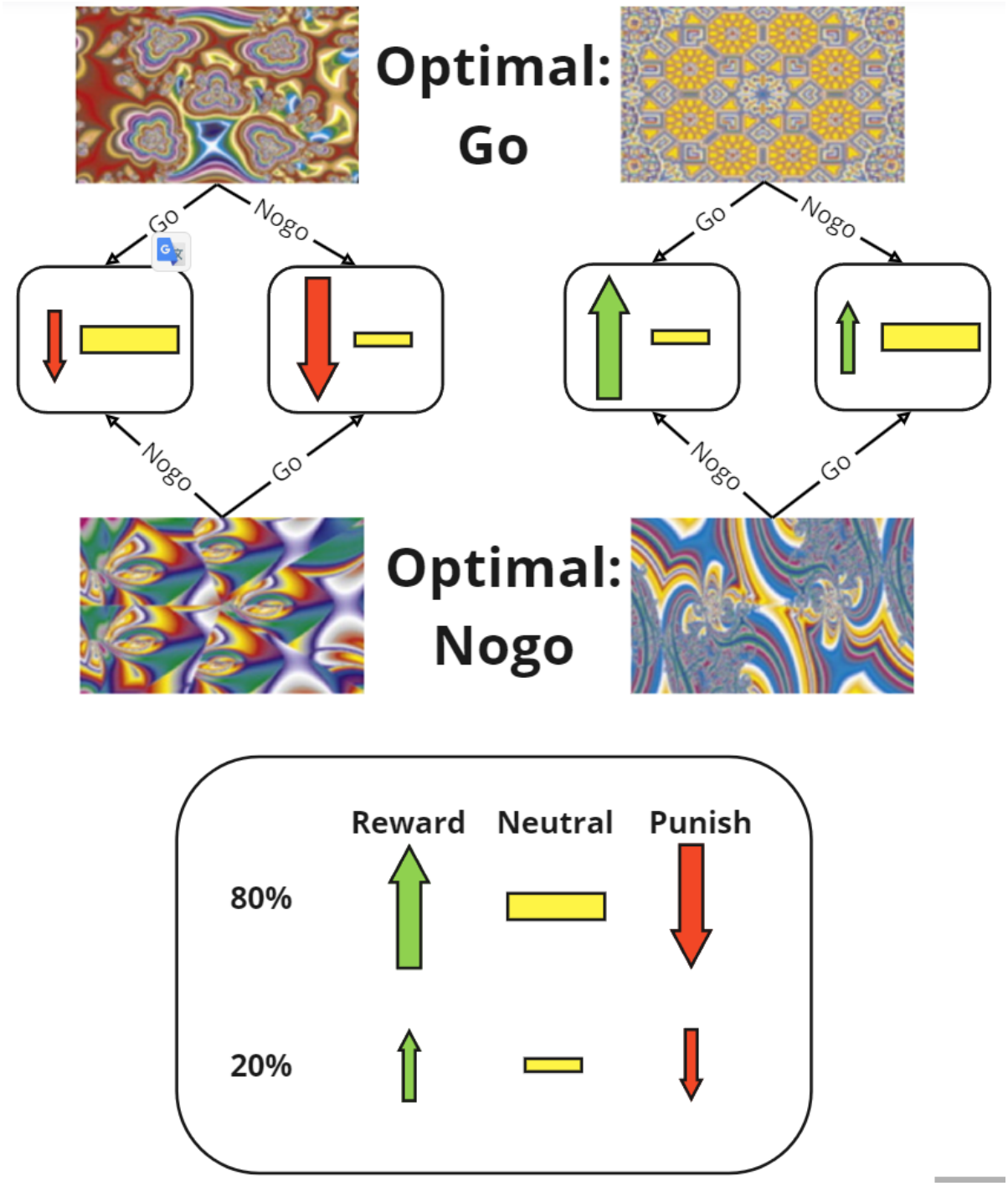


Figure 4.2: **Image pool and outcome probabilities for the fractal go/nogo task.** This image was adapted from Guitart-Masip et al. (2012). The image shows four fractal images one might present to participants in the fractal go/nogo task. The optimal response for a participant to make is indicated between the relevant images. Arrows point to the outcome probabilities when a participant makes a go response vs. a nogo response. A large arrow or dash indicates an 80% probability, while a small arrow or dash indicates a 20% probability. A green up arrow refers to a reward, a down red arrow to a punishment, and a yellow dash to neutral feedback.

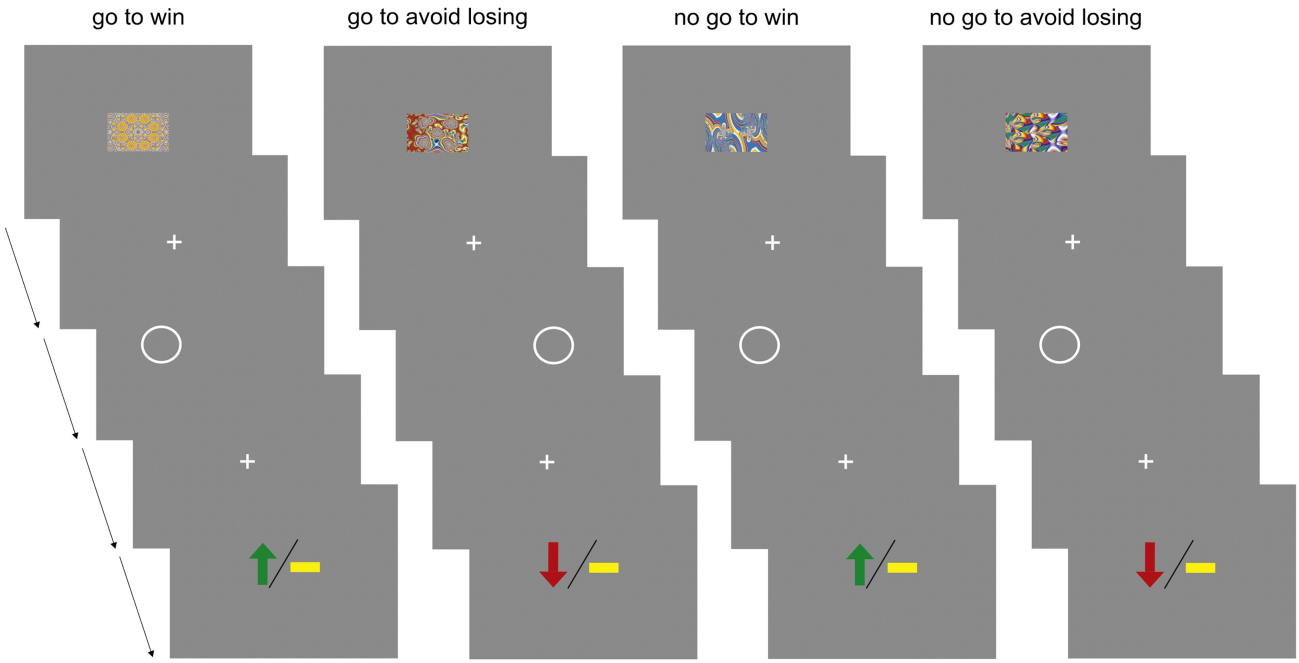


Figure 4.3: **Task sequence for the fractal go/nogo task** Subjects are shown one fractal image on the screen at a time. They are then shown a circle either on the left or the right of the screen, and they must decide whether to respond by saying which side the circle is on or not to respond. Each image has a different set of rules for generating feedback, but the subjects do not know what it is. They are then given feedback in the form of a green arrow pointing up, a red arrow pointing down or a yellow line. Image modified from Guitart-Masip et al. (2012).

- 'Nogo' to get reward
- 'Nogo' to avoid penalty

Guitart-Masip et al. (2011) did not provide clarity on their reward/penalty contingencies, so here I will describe the Guitart-Masip et al. (2012) version of the task. In 'go to get reward' and 'nogo to get reward' trials, 80% of correct responses get rewarded and 20% receive neutral feedback. In 'go to avoid penalty' and 'nogo to avoid penalty' trials, 80% of correct responses get neutral feedback and 20% receive a penalty.

Now that we have at our disposal two tasks for studying Pavlovian-instrumental transfer, let us specify precisely what this is.

4.2.3 Pavlovian bias and Pavlovian-instrumental transfer in healthy individuals

Recall from section 2.6 that Pavlovian (classical) conditioning occurs when an animal learns about stimulus-outcome relationships in situations where their actions have no effect on the outcomes. Instrumental (operant) conditioning occurs when an animal's actions do affect the outcomes it faces. The term Pavlovian bias refers to the observation that animals are inclined to approach rewards (or stimuli that predict rewards) and withdraw from punishments (or stimuli that predict punishments) (Robinson & Chase, 2017). A consequence of this is that a phenomenon called Pavlovian-instrumental transfer occurs when an animal's Pavlovian bias towards a stimulus prompts it to behave differently in an instrumental context from the way it would behave otherwise (Cartoni et al., 2013). For example, in the mushroom+fractal go/nogo task, an individual might learn through Pavlovian conditioning that a particular fractal image predicts punishment. In a separate training session, the individual might learn through instrumental conditioning that approaching a particular picture of a mushroom tends to lead to reward. If one then puts the aversive Pavlovian fractal behind the mushroom, participants become less likely to approach the mushroom even though it is to their advantage to do so. In the fractal go/nogo task, the same phenomenon gets triggered in a different way: here, the fractal images trigger both a Pavlovian and instrumental response.[2] Each fractal image has a state value $V(S)$ that depends on the degree to which it has been followed by rewards or punishments in the past. Each image also has an action value $Q(S, A)$ that specifies the desirability of performing action A when presented with that fractal. A particular fractal might be associated with an overall risk of loss, giving it a negative state value, but it may be that choosing to 'go' leads to a relatively better outcome than choosing to 'nogo'. In such a case, like in the mushroom+fractal example above, subjects are less likely to go in this situation than they would be if the image had been associated with reward.

Studies have examined how decisions of both healthy individuals and individuals with mental illness are influenced by Pavlovian-instrumental transfer. Guitart-Masip et al. (2012) and Huys, Cools, et al. (2011) have provided evidence for this effect in healthy individuals. Huys, Cools, et al. (2011) gave healthy subjects the mushroom+fractal go/nogo task described in section 4.2.1.

[2] It can be tricky to see why this task has a Pavlovian component as well as an instrumental component. After all, participants are asked to respond to the images shown to them, and are then given rewards or punishments in response. However, the Pavlovian component lies in the fact that no matter what the participant does, two of the images will never give rewards and the other two will never give punishment. One can see this in figure 4.2

An important facet of this task is that it had an approach block and a withdrawal block, whereas Guitart-Masip et al. (2012) did not distinguish between approach and withdrawal. Huys, Cools, et al. (2011) found that, in the approach block, positive Pavlovian stimuli in the background made it more likely that participants would 'go' in response to images of mushrooms, while negative Pavlovian stimuli made participants less likely to 'go'. In the withdrawal block, positive Pavlovian stimuli made 'nogo' more likely, while negative Pavlovian stimuli made 'go' more likely. The best-fitting reinforcement learning model to their data was the RW model that included separate reward and punishment sensitivity parameters ρ_{pun} and ρ_{rew} (see equation 2.5), and a single learning rate α for both reward and punishment. This implies that their participants distinguished between reward and punishment when it came to reward sensitivity, but not when it came to learning rate; in future, it may be worth thinking about why that is the case. The best-fitting model also had separate fixed general bias parameters for approach (b_{app}) and withdrawal (b_{wth}). This implies that participants had different degrees of bias towards 'go' in approach situations than in withdrawal situations. Unfortunately, the paper did not specify whether the bias was larger in approach situations or withdrawal situations.

Guitart-Masip et al. (2012) used the fractal go/nogo task described in section 4.2.2. Unlike Huys, Cools, et al. (2011), they did not distinguish between approach and withdrawal, and their fractal images served the dual purpose of signalling whether to 'go' or 'nogo' in a subsequent target detection task and of being associated (in a Pavlovian way) with either reward or punishment. They found that participants found it easier to 'go' when faced with reward than to 'go' when faced with punishment. Their best-fitting model was one with a single feedback sensitivity ρ and a single learning rate α; in other words, ρ and α were the same for reward and punishment. The single feedback sensitivity is in contrast to the findings of Huys, Cools, et al. (2011), who found that separate reward and punishment sensitivities fit their data the best.[3] Their best-fitting model also included a noise parameter ξ and a static general bias parameter b in favour of 'go'. Most importantly for this discussion, it had separate Pavlovian bias parameters ($\kappa_{\mathsf{approach}}$ and κ_{avoid}) for rewarded and punished trials respectively (Pavlovian bias parameters are explained in more detail in section 2.9). This implies that there was some difference in the degrees to which positive and negative Pavlovian stimuli strengthened or inhibited the tendency to 'go'; i.e. the amount by which positive stimuli strengthened the tendency to 'go' was different from the amount by which negative stimuli weakened the tendency to 'go'.

Now that we have gained an overview of how Pavlovian-instrumental transfer effects operate

[3] It's not clear why these two studies differ in this way.

in mentally healthy individuals, let us look at how these effects differ in individuals with mental illness.

4.2.4 Pavlovian-instrumental transfer (PIT) in mental illness

It has been suggested that disruptions in Pavlovian bias and PIT may be related to mental illness (Huys, Gölzer, et al., 2016; Metts et al., 2022). Normally, Pavlovian bias ensures that people are reflexively inclined to approach stimuli that have been previously associated with rewards (Huys, Gölzer, et al., 2016). If this effect is disrupted, it may require more conscious effort to approach rewards, and therefore individuals may find it more difficult to acquire rewards and may in fact experience them less frequently (Huys, Gölzer, et al., 2016). In particular, disruption in Pavlovian bias may be associated with anhedonia in the following way. Recall that anhedonia is defined as a loss of interest or pleasure. If someone finds it difficult to approach potential rewards, this could manifest as a loss of interest. It is also possible that subtle deficits in approach behaviour could manifest in a way that appears to be a loss of pleasure; for example, someone with such deficits who usually enjoys soccer might force themselves to go out and play, but they might fail to immerse themselves in the game fully and therefore fail to reap the same rewards as usual. Afterwards, they report experiencing little pleasure from the game and feel less inclined to play soccer in future. This could lead to a vicious cycle where difficulty in approaching rewards leads to actually receiving fewer rewards, which in turn leads to the individual being less inclined to approach rewards.

Three of the four studies I have found that have studied the relationship between PIT and mental illness (Huys, Gölzer, et al., 2016; Metts et al., 2022; Mkrtchian et al., 2017; Nord et al., 2018) do not do the same trial-by-trial data analysis using reinforcement learning models as the studies discussed in section 4.2.3, making these groups of studies difficult to compare with one another. This may be because the authors considered the average biases in subjects' responses more pertinent than how their responses changed over time. Mkrtchian et al. (2017) is a notable exception. They studied healthy subjects and subjects with mood and anxiety symptoms, using the same task as Guitart-Masip et al. (2012), except that they administered it both under a safe condition and a threat-of-shock condition. Their best-fitting model was similar to the best-fitting model in Guitart-Masip et al. (2012) in the sense that it was also a RW model with a noise parameter, a static general bias parameter, and separate Pavlovian bias parameters for approach and avoidance. The differences were that this model included separate reward and punishment

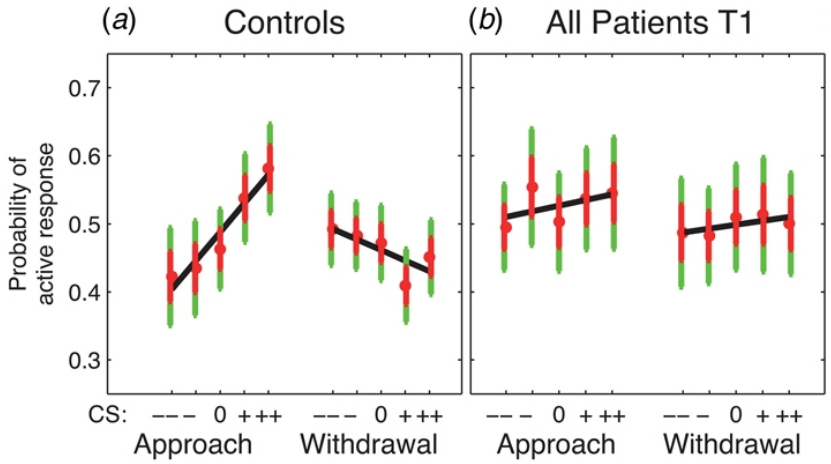


Figure 4.4: **Probability of a 'go' response relative to stimulus valence in Huys, Gölzer, et al. (2016)** This image is taken from Huys, Gölzer, et al. (2016). These plots show the probabilities of participants giving 'go' responses to five Pavlovian stimuli with valences ranging from very negative ('--') to very positive ('++'). A probability of 0.5 corresponds to no bias for or against a stimulus. The authors have plotted best-fit linear functions to the 'go' probabilities as a function of valence. In the approach part of panel (a), we see that there is a positive correlation between valence and 'go' responses. In the withdrawal part of panel (a), there is a negative correlation between valence and 'go' responses. In panel (b), showing results for depressed individuals, the slopes of both lines are slightly positive, but it is not clear from the paper whether the authors found these slightly positive slopes indicative of positive correlations between valence and 'go' responses. What the authors do point is that the slopes here are not significantly different from each other.

sensitivities and separate learning rates for reward and punishment. The authors found that subjects in their mood and anxiety group had a higher Pavlovian avoidance bias parameter κ_{avoid} than the healthy subjects did. The mood and anxiety group also had a larger increase than the healthy group in the avoidance parameter when comparing a safe situation to a situation where they were under threat.

Although the studies I will mention in the rest of this section did not do trial-by-trial data analysis, their results are relevant to depression and anhedonia and so I will include them here. Huys, Gölzer, et al. (2016) and Nord et al. (2018) both used the mushroom+fractal go/nogo task described in section 4.2.1. In their PIT stage, they displayed previously conditioned Pavlovian stimuli (fractal images tiled in the background) along with previously conditioned instrumental stimuli (pictures of mushrooms). Their experiments had both an approach block (where participants had to decide whether to approach a stimulus or not) and a withdrawal block (where participants had to decide whether to withdraw from a stimulus or not).

Huys, Gölzer, et al. (2016) plotted the probability of an active response (either approach or withdrawal) with respect to the valence[4] of the Pavlovian CS, and compared the linear trend of this relationship for combinations of healthy versus depressed patients and approach versus withdrawal. Their plots are shown in figure 4.4. It seems qualitatively clear from the plot for controls in the approach block that there is a positive correlation between stimulus valence and 'go' responses, and this is indeed what the paper claims to have found. They also found a negative correlation between valence and 'go' responses for controls in the withdrawal block. Comparing the slopes of the plots for approach and withdrawal in controls, they found that the slopes were significantly different. The authors refer to this phenomenon as **action specificity**. To calculate the action specificity measure, they took the slope of the best-fit line for approach and subtracted the slope of the best-fit line for withdrawal. Checking for action specificity in depressed participants revealed no effect; that is, the slopes for the approach and withdrawal blocks were not significantly different.

Huys, Gölzer, et al. (2016) also compared the slope for healthy participants to the slope for depressed participants, both within the approach block and within the withdrawal block. In the approach block, they found a difference between the trends for healthy versus depressed participants, but no such difference in the withdrawal block.[5] They suggest that their results might mean that depressed people's choices are guided less by Pavlovian stimuli than those of healthy people.

Nord et al. (2018) did the same experiment, and also did linear fits to plots of 'go' probability versus valence so that they could compare their results with those of Huys, Gölzer, et al. (2016). As we can see from figure 4.5, they found action specificity in their MDD patients, but not controls. This is in direct contradiction to the findings of Huys, Gölzer, et al. (2016), who found action-specific PIT in controls but not patients. Nord et al. (2018) suggest that this may be because the patients in their study were unmedicated, while Huys, Gölzer, et al. (2016) studied a mixture of medicated and unmedicated patients. Another difference between the experimental groups is that Huys, Gölzer, et al. (2016) studied both patients with MDD and dysthymia, while Nord et al. (2018) only studied patients with MDD.

Let us now take a step away from action specificity and take a look directly at the diagrams in figure 4.6, taken from Nord et al. (2018). Based on previous research on healthy individuals

[4]The word 'valence' refers to the degree of attractiveness of a stimulus.

[5]Visually, it looks as though the trends for the two groups in the withdrawal block are different too because the slopes are different, but apparently the statistics showed no significant difference.

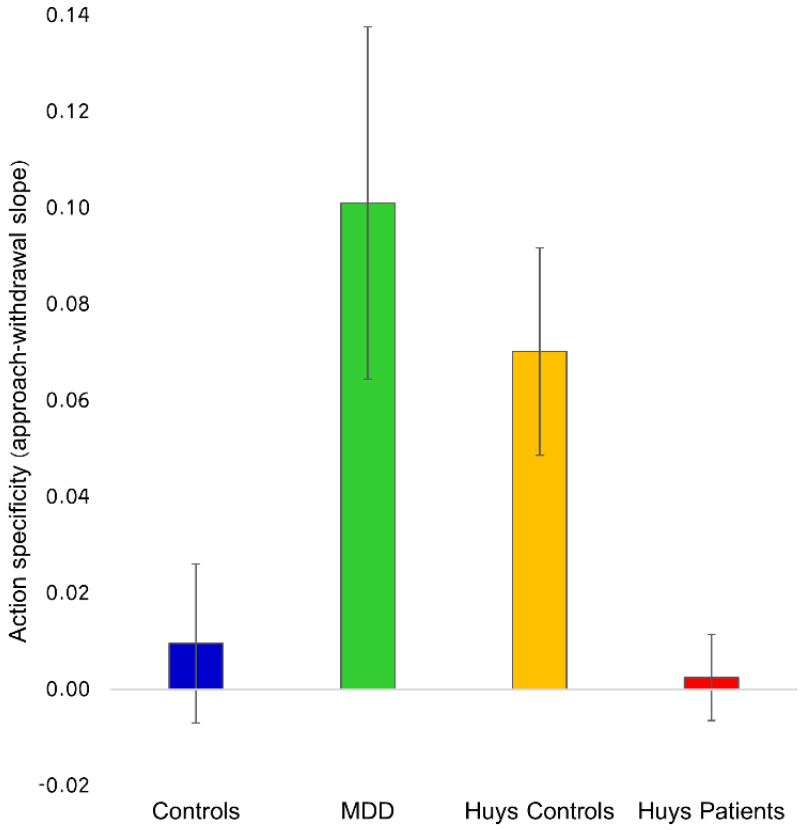


Figure 4.5: **Comparing action specificity in Huys, Gölzer, et al. (2016) and Nord et al. (2018)**. This is a plot from the supplementary materials of Nord et al. (2018), comparing action specificities for patients and controls in their paper to those in Huys, Gölzer, et al. (2016). Here we can see the clear way the results of Nord et al. (2018) contradict those of Huys, Gölzer, et al. (2016). While Huys, Gölzer, et al. (2016) found action specificity in controls but not patients, Nord et al. (2018) found it in patients but not controls. Action specificity was calculated by subtracting the slope of the best-fit line for the withdrawal block from the slope of the best-fit line for the approach block.

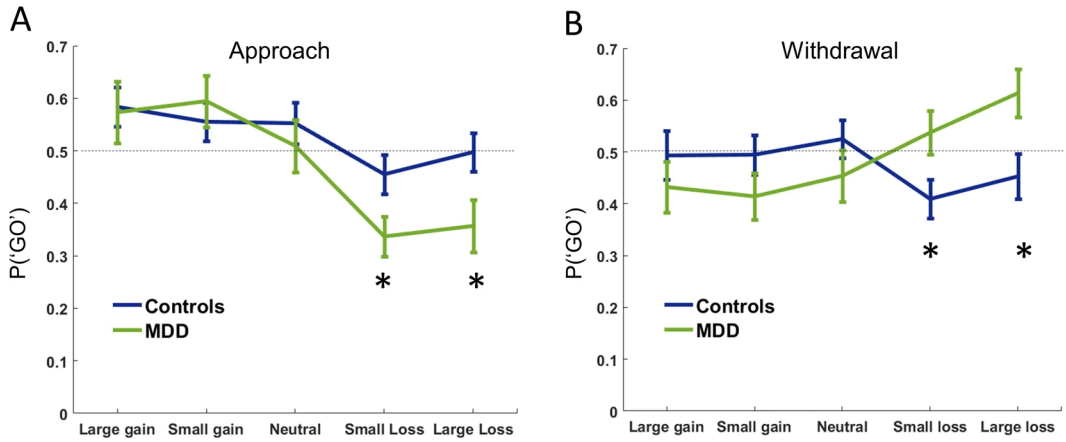


Figure 4.6: **Probability of a 'go' response relative to stimulus valence in Nord et al. (2018)** This image is taken from Nord et al. (2018). Similarly to figure 4.4, these plots show the probabilities of participants giving 'go' responses to five Pavlovian stimuli with valences ranging from very positive ('large gain') to very negative ('large loss'). A probability of 0.5 corresponds to no bias for or against a stimulus. We can see from **plot A** that both control and MDD participants were biased towards 'go' when faced with positively valenced stimuli in the approach block of the experiment. When it came to negatively valenced stimuli in the approach block, participants with MDD were significantly more biased towards 'nogo' than those without MDD. For the withdrawal block (**plot B**), we see that, in the presence of positively valenced stimuli, MDD participants were biased towards 'nogo' but healthy participants were not. In response to negatively valenced stimuli, MDD participants were biased more towards 'go' than healthy participants.

(Huys, Cools, et al., 2011), we can expect to see a PIT effect in the controls: in approach situations, the Huys, Cools, et al. (2011) participants were biased towards 'go' when faced with rewards and towards 'nogo' when faced with punishments. In withdrawal situations, individuals were biased towards 'go' when faced with punishments and towards 'nogo' when faced with rewards. In Nord et al. (2018), this effect surprisingly turns out not to be clear from the plots for controls, but one can see it clearly for participants with MDD. From the approach block diagram in Nord et al. (2018), we can see that both controls and patients with MDD were biased towards 'go' when faced with positively valenced stimuli (gains). Patients with MDD were biased towards 'nogo' when faced with negatively valenced stimuli (losses) - significantly more so than controls. In the withdrawal block, patients were biased towards 'nogo' when faced with rewards and towards 'go' when faced with loss. These results are indicative of a PIT effect, and, confirming statistically what the plots suggest visually, the paper found that this effect was stronger in MDD patients than controls. Importantly, through statistical analysis the authors found that this effect was due mainly to stronger responses to negatively valenced stimuli in the

MDD group.

It would have been interesting to compare the best-fitting reinforcement learning models, as well as the values of Pavlovian bias parameters, between depressed and healthy individuals using more than just one study (Mkrtchian et al., 2017). It is intriguing that Mkrtchian et al. (2017) found increased reliance on a Pavlovian avoidance bias parameter in their mood and anxiety group compared to healthy controls, and it would have been interesting to know whether this was the case in the unmedicated MDD patients in Nord et al. (2018) as well. Unfortunately, Huys, Gölzer, et al. (2016) and Nord et al. (2018) did not do trial-by-trial data analysis with reinforcement learning models, so such a comparison is not possible.

4.3 Findings on learning rate and reward/punishment sensitivity

We have just discussed the complex ways mental illness can affect PIT and the Pavlovian bias parameter. Now we move on to learning rate and reward sensitivity.

4.3.1 Introduction to learning rate and reward/punishment sensitivity

Reward learning Some people have more trouble than others at learning from rewards. Reduced reward learning means that someone is less likely to change their behaviour in response to rewards. When individuals with reduced reward learning are given a task where they are rewarded for correct decisions, these rewards have a smaller impact on their future decisions than for healthy individuals (Henriques et al., 1994). Vrieze et al. (2013) found that inpatients with MDD had impaired reward learning compared to healthy controls and that patients with high anhedonia were worse at learning from rewards than patients with low anhedonia. Impaired reward learning increased the probability that patients would still be depressed after 8 weeks of treatment. Kumar et al. (2018) and Reinen et al. (2021) also found that individuals with MDD are impaired when it comes to learning from rewards . Impairment at reward learning cuts across psychiatric disorders; for example, people with schizophrenia are also worse than healthy individuals at learning from rewards (Reinen et al., 2014).

We saw above that anhedonia and MDD are associated with reduced reward responsiveness. However, the cause of this observation is not clear. Reinforcement learning models can provide

a way to distinguish between two upstream parameters that might cause reduced reward responsiveness: reduced reward sensitivity and reduced learning rate. Reward prediction error can be written as in equation 2.4,

$$\delta_t = \rho r_t - V_t(S).$$

If we then update the state value according to RW as described in chapter 2,

$$V_{t+1}(S) = V_t(S) + \alpha\delta_t$$

where α is the learning rate, it becomes clear that both α and the reward sensitivity ρ can affect the way the value of a state gets updated.

The probabilistic reward task from section 2.9.1 can be used to study reward sensitivity and learning rate in response to rewards. Other tasks have been used to distinguish between responses to reward and punishment, for example the probabilistic selection task and the probabilistic reversal learning task.

The probabilistic selection task gets used by Chase et al. (2010), Frank et al. (2004), and Kunisato et al. (2012). Participants are given three pairs of pictures to choose from, 'AB', 'CD', and 'EF'. In the case of each pair, one picture is more likely to lead to reward than the other; for example, when the 'AB' pair gets presented, choosing 'A' leads to positive feedback 80% of the time and negative feedback 20% of the time, while choosing 'B' leads to positive feedback 20% of the time and negative feedback 80% of the time. The same idea holds for the other pairs, but for 'CD' the percentages are split 70%/30% and for 'EF' they're split 60%/40%. Participants are then given new combinations of stimuli to choose between, for example 'AC', 'AD', etc. Their choices for these new pairs then allow researchers to determine whether they learn more from positive or from negative feedback.

Another relevant task is the probabilistic reversal learning task: Participants are shown two pictures simultaneously and are asked to choose between them. One picture mostly gives pleasant feedback and the other mostly aversive feedback. At various points in the experiment, the probabilities switch, and to maximise their reward participants have to realise this and switch their choice. This is essentially a two-armed bandit task (Sutton & Barto, 2020, chapter 2). Now we move on to some results from the probabilistic reward task, the probabilistic selection task and the probabilistic reversal learning task.

Huys et al. (2013) used the probabilistic reward task from section 2.9.1 to study the differences in learning rates and reward sensitivities between groups of people. They found that

anhedonic depression, as measured by a subscore of the Mood and Anxiety Symptom Questionnaire (MASQ), was significantly correlated with lower reward sensitivity ρ.

Using the probabilistic selection task, Chase et al. (2010) found no asymmetry between learning rates for reward and punishment in participants with MDD or in healthy controls. They found that subjects with high levels of anhedonia had smaller learning rates for both reward and punishment. This supports the idea that anhedonia is related to blunting, where blunting refers to a lowered effect of feedback on behaviour. In their discussion they point out that a reasonable explanation for their data might be that MDD is related to blunting via anhedonia; that is, people with MDD often suffer from anhedonia, and anhedonia is related to blunting. An apparent contradiction between Chase et al. (2010) and Huys et al. (2013) is that the latter found differences in reward sensitivity while the former found differences in learning rate. However, note that the model used by Chase et al. (2010) did not include a reward or punishment sensitivity parameter, and recall that in a Rescorla Wagner (RW) model, $Q(s_t, a_t)$ is incremented by $\alpha\delta$. So what appeared to them to be a difference in the learning rate α could conceivably have arisen from a difference in reward prediction error δ, which in turn could have been caused by changes in sensitivity to reward or punishment.

It is useful to distinguish between reward sensitivity and learning rate because they are likely to have different causes and treatments (Huys et al., 2013). It is significant that anhedonia is more strongly correlated with reward learning than a diagnosis of major depressive disorder (MDD). Perhaps this suggests that, if we are interested in studying neural correlates of impaired reward learning, anhedonia is a more useful quantity than MDD to measure and according to which to group people for research purposes. The presence of reduced reward learning being present across several psychiatric disorders suggests that reward learning may serve as a common thread that can help us understand better the interrelatedness of psychiatric illnesses.

4.3.2 'Liking' and 'wanting'

The concepts of 'liking' and 'wanting' are related to reward sensitivity and tend to come up in papers on reward learning (C. Chen et al., 2015; Huys et al., 2013; Treadway et al., 2009), so it is worth having a sense of what they mean. We can divide reward-seeking into two separate strands: 'liking' and 'wanting' (Treadway et al., 2009). As described in C. Chen et al. (2015), 'liking' refers to the response of the nervous system once a reward is received. It is a measure of the pleasure obtained from receiving the reward. 'Wanting', on the other hand, refers to the

degree of motivation the individual feels to approach a reward (C. Chen et al., 2015).

How liking and wanting are affected in depression and anhedonia Several papers have measured the degrees of 'liking' and 'wanting' in different groups of individuals. Treadway et al. (2009) found that anhedonia is linked to decreased 'wanting' as opposed to 'liking', while C. Chen et al. (2015) reported decreased 'wanting' in individuals with depression and in particular those with suicidal behaviours.

Linking liking and wanting and reward sensitivity Reward sensitivity ρ and 'liking' can be viewed as the same quantity (C. Chen et al., 2015; Huys et al., 2013). Huys et al. (2013) describe reward sensitivity as the 'internal worth of an external reward'. Huys et al. (2013) connect reward sensitivity ('liking'), to 'wanting' in the following way: a higher degree of 'liking' means attaching higher internal worth to an external reward. This, in turn, means that in future the individual will be more motivated to obtain that reward, hence a higher degree of 'wanting'. In other words, the cause of impaired reward learning is not that the individual is not learning sufficiently from the internal rewards they are experiencing, but rather that the internal reward signals are too low to begin with.

4.3.3 A note on TD learning models

The review by C. Chen et al. (2015) points out that most of the studies they reviewed used either the original RW update rule or modified versions of it. Only two studies in their review used TD (Gradin et al., 2011; Kumar et al., 2008); I will elaborate on those below. Similarly, the only reinforcement learning models that the review by Robinson and Chase (2017) addresses are variations of RW. The scarcity of studies that consider TD models may be due to the fact that existing tasks tend to have single-step trials, and the time between the stimulus and feedback delivery is generally considered irrelevant. Since TD models are distinguished from RW models by their consideration of expectations of future feedback, there is no point in using them to model a single-step task unless one cares about the timing of feedback delivery.

In cases where tasks do contain more than one step, researchers have indeed put to use models that consider expectations of the future. For example, Daw et al. (2011) used a SARSA(λ) algorithm to model human behaviour in a two-step task. Unfortunately they did not report the values of their RL parameters, so their results are outside the scope of this dissertation. There are also some studies that used single-step tasks, but where the researchers were interested in people's learning about the timing of feedback delivery. Two examples are Gradin et al. (2011)

and Kumar et al. (2008).

Kumar et al. (2008) let individuals participate in a Pavlovian learning study, and examined how measured brain activity correlated with predictions made by the TD model described in section 2.11. They found differences in prediction errors between depressed individuals and healthy controls in several brain regions. Their results are complex, however, and to discuss them I would have to delve into details of neuroimaging and brain anatomy, which fall outside the scope of this project. The only point I will highlight here is that when they examined how the predictions of their model changed when they varied their learning rate between 0.1 and 0.4 and their discount factor between 1.0 and 0.4, they found little difference. This would indicate that their model was insensitive to specific RL parameters anyway and would therefore be of limited usefulness for our purposes.

Gradin et al. (2011) gave their participants an instrumental learning task, as opposed to Kumar et al.'s (2008) Pavlovian learning study. They studied three groups: healthy controls, people with a diagnosis of MDD, and people with a diagnosis of schizophrenia. Like Kumar et al. (2008), they studied how brain activity correlated with prediction errors given by an RL model. They compared the RL parameters learning rate and inverse temperature among their three groups and found no significant differences.

The fact that Kumar et al. (2008) found that their model was insensitive to which RL parameter values they used is interesting. When I run my implementation of the same model, described in section 2.11, I do find clear differences between a learning rate of 0.1 and 0.4 and a discount factor of 1.0 and 0.4. Figure 4.7 shows the prediction errors given by my model when I alter the learning rate α and the discount factor γ. This discrepancy could be due to a coding mistake by either them or me, or perhaps I am misunderstanding what they mean.

In this chapter, we took a whirlwind tour through a variety of topics. I started by introducing you to the concepts of anhedonia and endophenotypes to help motivate the idea that it may make more sense to study correlations between RL parameters and anhedonia than RL parameters and depression. I then moved on to Pavlovian-instrumental transfer (PIT) and how aspects of it change when we become mentally ill. We saw limited evidence that the Pavlovian avoidance bias parameter in RL models might be different between health and illness. We then considered other parameters, namely learning rate, reward sensitivity and punishment sensitivity. Fortunately there was more literature to discuss when it came to these parameters than there was for Pavlovian

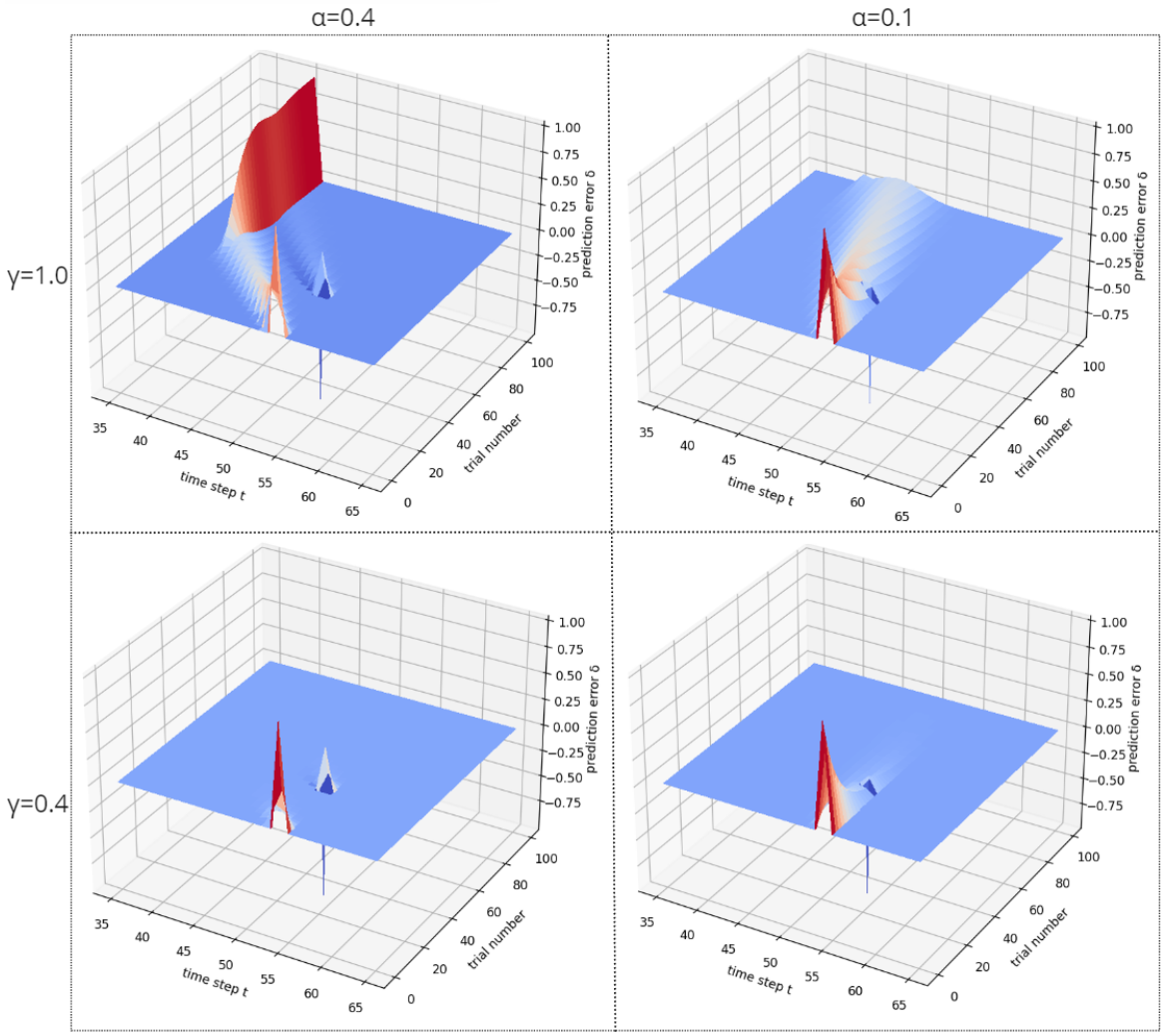


Figure 4.7: This grid shows results of my implementation of the TD model described in section 2.11 for different values of learning rate α and discount factor γ. One can see clear differences in the modelled prediction errors for different parameter combinations, contradicting what Kumar et al. (2008) found.

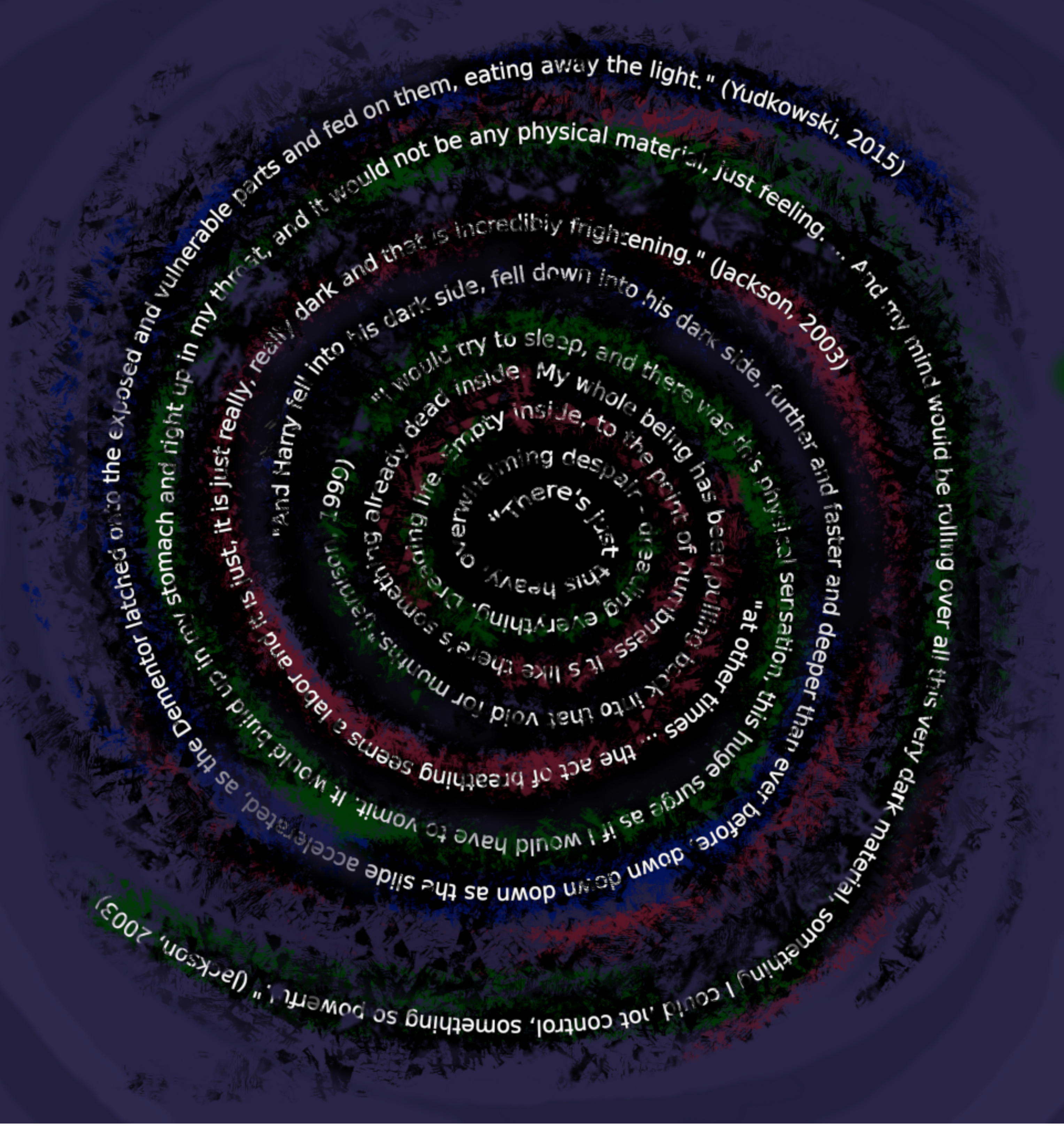
avoidance bias, but unfortunately the studies were difficult to compare with one another[6] and sometimes contradicted one another. Perhaps a careful reading and critical analysis of all the literature on reward and punishment learning (that is, not only literature that explicitly involves RL modelling) would result in more clarity, but that would be beyond the scope of this dissertation. If you are interested in undertaking that task, section 4.3.2 may help you relate RL parameters to non-RL terminology.

Now that this chapter has put a smorgasbord of facts on the table, let us move on to discussing what these facts imply for the way we conceptualise depression and how (or even if) we should fit it into our classification systems for mental illness.

[6]The major obstacles were that different tasks and models were used between studies.

Chapter 5

Discussion: using behavioural findings to improve classification and treatment of mental disorders



As we saw in section 1.4, treatment for depressive disorders is still far from adequate. In this chapter I argue that that is partly due to the fact that we do not even have a clear understanding of what depression is, let alone how to treat it. Algorithmic models like reinforcement learning may help improve this situation by uncovering key latent variables that underlie clusters of symptoms. Later in this chapter I suggest a possible way to gather data about the way someone learns and makes decisions so that models can be fitted to the data and the values of latent variables inferred. First, however, let us look at the problem of defining depression.

Our difficulty in treating depressive disorders seems unsurprising when we consider how little we know about their aetiology or even what they are. The common thread among the depressive disorders in the DSM-5 is that they all involve the entity we refer to as 'depression', but it is surprisingly difficult to define that entity. Perhaps, if we can understand better what we mean when we say 'depression', that might lead to a clearer understanding of what causes it, which might in turn eventually lead to better treatment. Let us consider the various things people mean when they talk about depression.

5.1 A deeper look at what depression is

When talking and reading about depressive disorders like MDD and dysthymia, one repeatedly comes across the words 'depressed' and 'depression'. But what *is* depression? While researching this masters, I have found it surprisingly difficult to answer that question. Even scientific studies use these words without clearly defining them; for example, most of the studies included in the meta-analysis by Arroll et al. (2005) included 'patients thought by their general practitioner to be depressed', but Arroll et al. (2005) give no definition of 'depressed'. Its meaning seems to be assumed to be commonly understood, but as I tried to explain it here, I realised that there were complexities I had not thought about before. Does the word refer to a particular psychiatric disorder? Or to the disorders listed under 'Depressive Disorders' in, for example, the DSM-5? But then that would make its definition dependent on one particular edition of one diagnostic manual, which would not be consistent with the universality of its use. Could it rather make sense to define it as a score above a certain number on an inventory like the Mood and Anxiety Symptom Questionnaire (MASQ) (Watson & Clark, 1991)? Again, given that likely only a tiny fraction of people who use the word 'depression' are referring to a number on a scale, that seems unreasonable as a definition. Eventually I accepted that I was not going to find a widely agreed-upon definition, and I started to consider that perhaps the meaning of the word is left vague in

the literature because of how amorphous a concept depression is. A. T. Beck and Alford (2009, p.1) strikingly express the limitations of our understanding of depression:

> Although depression (or melancholia) has been recognized as a clinical syndrome for over 2,000 years, as yet no completely satisfactory explanation of its puzzling and paradoxical features has been found. There are still major unresolved issues regarding its nature, its classification, and its etiology.

In short, depression is an entity that clinicians recognise when they see it, but they would be hard-pressed to say exactly what it is, how to classify it or what causes it. Part of the reason for this is certainly our limited knowledge about this entity, but A. T. Beck and Alford (2009, p.8) point out that another reason it is difficult to conceptualise depression is that the word has been used in three ways: firstly, as a feeling; secondly, as a symptom complex or syndrome; and thirdly, as a clearly defined disorder. Colloquially, people often refer to depression in the first sense, as a feeling. The Oxford Advanced Learner's Dictionary defines this sense of the word 'depression' as 'the state of feeling very sad and without hope'. Other sources use depression in the third sense, as a particular disorder; for example, an educational page for patients on the website of the American Psychiatric Association defines depression as synonymous with MDD (American Psychiatric Association, n.d.). The second sense is the one in which I use the word in this dissertation, as referring to a syndrome. (When I want to refer to particular disorders, I explicitly use the names of those disorders, for example major depressive disorder (MDD).) I believe there is value in referring to this somewhat vaguely defined concept because, due to our limited current understanding of the underlying disease process, any clear definition would necessarily sometimes exclude cases where it is at work and at other times include cases where it is not. Let me therefore proceed to describe the syndrome of depression as clearly as I am able, given the reality that nobody knows exactly what it is.

Signs and symptoms of depression According to A. T. Beck and Alford (2009, p.12), there is broad agreement on the core signs and symptoms of depression. These include 'low mood, pessimism, self-criticism, and psychomotor retardation or agitation'. A. T. Beck and Alford (2009) arrived at a list of symptoms to include in their book by first examining several sources to see which symptoms others had attributed to depression, and then studying a group of patients and a group of controls to determine which symptoms occurred more often in depressed than non-depressed individuals. There is clearly circularity here: a conceptualisation of depression would have been necessary to identify individuals to put in the depressed group, but then the

differences between the groups were used to update the conceptualisation of depression. This kind of circularity was perhaps inevitable given the fact that depression is a syndrome whose characteristics are arrived at by consensus. A. T. Beck and Alford (2009) classify depression symptoms into emotional, cognitive, motivational and physical. The emotional manifestations include, among others, dysphoric mood, negative feelings towards the self, loss of pleasure, and loss of interest in other people or activities. Cognitive manifestations include low self-esteem, hopelessness, and high levels of self-criticism. In the motivational domain, depressed people tend to struggle to mobilise themselves to engage in their usual activities. This lack of motivation may extend to no longer having the will to live, resulting in suicidal thoughts. Finally, depression is associated with physical manifestations: patients may lose their appetite, suffer from disturbed sleep, lose interest in sex, and feel constantly tired. Clearly not all of these symptoms need to be present in a patient for us to recognise the syndrome of depression in them; this is simply a list of symptoms which frequently occur together. As soon as we start to ask how many and which symptoms must be present (and to what degree) for someone to qualify as being depressed, we enter the realm of diagnosis, discussed in the next section.

5.2 Current classification of mental disorders

There are two main approaches to psychiatric diagnosis currently in use: categorical and dimensional. Categorical approaches focus on placing behaviours and symptoms into discrete disorder categories that are qualitatively different from one another (Shea, 2017, p.302). To decide into which category (or categories) to place a particular set of difficulties a person is facing, categorical diagnostic systems use lists of criteria that must be met for inclusion into disorder categories (Shea, 2017, p.302). Such diagnostic systems usually list a number of symptoms for each disorder, and specify which symptoms are most important and how many must be present for a diagnosis. Both the DSM-5 (American Psychiatric Association, 2013) and International Classification of Diseases (World Health Organization, 2018) use a categorical approach as their primary approach to diagnosis of mental disorders. For example, according to the DSM-5, a diagnosis of MDD requires either depressed mood or anhedonia (or both) to be present, along with three or four other symptoms so that there is a total of five symptoms.

Dimensional approaches to diagnosis, on the other hand, do not simply mark signs and symptoms as either present or not present; instead, they recognise the fact that most psychological traits exist on a continuum between perfect health and complete dysfunction. Professionals

using such approaches consider a number of important characteristics and rate the degree to which each characteristic is present, in order to obtain a holistic understanding of the individual. Such systems are still in development and are not yet in wide clinical use for most areas of psychological functioning (Cuthbert & Insel, 2013). A notable exception is the DSM-5's dimensional approach to assessing personality dysfunction (American Psychiatric Association, 2013, pp.761-781), which allow clinicians and researchers to rate individuals on a scale from 'little to no impairment' to 'severe impairment' in various areas of personality functioning. Currently, the most prominent project working towards developing dimensional diagnostic systems is the research domain criteria (RDoC) (Cuthbert & Insel, 2013) project. The RDoC project was started in 2009 by the United States' National Institute of Mental Health (NIMH) in an attempt to overcome some of the problems with existing categorical diagnostic systems (Cuthbert & Insel, 2013). It does not aim to be a diagnostic system in itself, but rather to identify key dimensions that are related to the 'primary behavioral functions that the brain has evolved to carry out'. The hope is that organising research around these dimensions may assist us in understanding the pathological processes underlying mental illness.

Both dimensional and categorical diagnostic systems have advantages and disadvantages. A notable drawback of the categorical approach is that it requires clinicians to place patients' difficulties into categories that may not correspond exactly with underlying disease processes. There may be more than one distinct underlying process causing the symptoms of MDD, for example, or a single underlying process might be causing symptoms associated with several different disorders. The categorical approach encourages us to think of psychiatric disorders as 'things', while it is more accurate to think of them as processes that influence the ways we interact with and experience the world (Shea, 2017). Shea (2017) goes as far as saying that they 'become a way of living'. Dimensional approaches honour this insight by acknowledging that each person has a multitude of characteristics that apply to that person to various degrees at a given time. The problem with such diagnostic systems is that they require clinicians to rate each patient on a large number of characteristics, which can take an impractical amount of time. Despite their drawbacks, categorical diagnostic systems have the advantages that they are quick to use and have a high inter-rater reliability (Shea, 2017). Hence they are currently the dominant diagnostic systems in clinical use.

5.3 Improving classification with latent variables

I contend that it may be possible to overcome some of the drawbacks of existing categorical and dimensional diagnostic systems through the use of the algorithmic modelling approach discussed in chapters 2 and 3 of this dissertation. Let me elaborate on that idea.

Instead of basing diagnosis on clusters of symptoms (that is, directly observed data), we might find it useful to base it on the values of latent variables (variables that are not directly observed) like learning rate, reward and punishment sensitivity, and Pavlovian bias. Making our diagnoses based on the values of latent variables might make it easier to organise our thinking when it comes to mental disorders and help us understand how seemingly unrelated symptoms fit together. For example, we saw in chapter 4 that the Pavlovian bias parameter for avoidance (κ_{avoid}) differed between patients with mood and anxiety symptoms and healthy individuals (Mkrtchian et al., 2017). If this turns out to be a distinguishing factor between depressed and non-depressed individuals, could an altered Pavlovian avoidance bias parameter become one of the signs of depression? A categorical approach could list a significantly altered κ_{avoid} as one of the symptoms required for a diagnosis of depression.[1] A dimensional approach to diagnosis, on the other hand, might list κ_{avoid} as one of the dimensions that need to be rated in order to arrive at a full clinical picture of an individual. Similarly, altered learning rate and reward sensitivity could be used as symptoms. We have seen in chapter 4 that anhedonia is related to altered reward sensitivity, but not learning rate. If high levels of anhedonia is indeed a useful clinical construct, then measuring a low reward sensitivity in an individual through game-based assessments (discussed below) could be a way to identify anhedonia that is complementary to the traditional approach of interviewing and observation of real-world behaviour.

If one were to let individuals perform multiple tasks to measure different aspects of their reinforcement learning process, one might end up with a large number of parameters of possible relevance to the individuals' behaviour. One might then want to determine which reinforcement learning parameters contribute the most to variation between individuals. A possible way to achieve this might be to perform principal component analysis (PCA) on the parameters to determine relationships between them, and in this way isolate the variables for which it would be most helpful to find values.[2] An example where this might be useful is in the case of reward

[1] It could also be listed as a symptom of a new disorder, for example if we end up replacing depression with a different disorder that serves patients better.

[2] One of my examiners suggested that it may be worth looking into other dimensionality reduction techniques, too,

sensitivity ρ, learning rate α and inverse temperature β. Huys, Cools, et al. (2011) pointed out that these parameters are difficult to untangle, and PCA might help us work out which of those conveys the most information.

5.4 A practical suggestion to assist with diagnosis and possibly treatment

It might be possible to implement such an approach to diagnosis by finding a fun and accessible way to gather large amounts of data about how people learn and make decisions. One such way might be to design an app that gathers the information we need as the user plays games.

Such an app might in future even be useful not only for diagnosis but also for practicing thinking and behavioural processes with which a particular person is having trouble (Mkrtchian et al., 2017). Behavioural and cognitive therapies already operate on the premise that changing the way we think and behave can change our feelings (J. S. Beck, 2011). For example, if someone is biased towards noticing negative social feedback, a therapist might encourage them to recall positive social feedback they have received recently. Or if a depressed person is isolating themselves from others, they could be encouraged to go out and interact with people more. Could these principles also work if new thoughts and behaviours were practiced in a highly artificial context such as that of a computer game? Moreover, could it be useful to practice not only thoughts and behaviours that are directly related to an individual's symptoms, but also to practice thoughts and behaviours that are linked to latent variables like Pavlovian avoidance bias, which are in turn linked to symptoms? If so, this could provide an opportunity to develop smartphone-based therapies that complement existing cognitive-behavioural therapies. These therapies could potentially help people who do not feel comfortable speaking to a therapist, and the fact that these therapies would address latent variables may mean that people could improve their mental health without explicitly having to think about their symptoms at all. This could be one way to reach people who are reluctant to seek help, and might help address the problem of delayed help-seeking for mental illness (P. S. Wang et al., 2007).

Before such treatments can become a reality, the field of computational psychiatry will need develop further. In particular, we will need to continue to identify relevant latent variables and strengthen our understanding of how they connect to psychiatric problems. It is possible that,

for example Uniform Manifold Approximation and Projection (UMAP) and Linear Discriminant Analysis (LDA).

given the high complexity of real-world decision-making, more complex tasks with multiple stages will be needed to fulfil this task adequately. More sophisticated reinforcement learning models may be needed to model learning and decision-making in such tasks.

A related point is that, to become widely used, an app that aims to gather large amounts of data on decision-making would need to be fun to use. The simple tasks currently given to participants in experiments are unlikely to have the same appeal as the games that people tend to play for fun. Part of the fun in recreational electronic games lies in the complexity of the decision-making required. At the same time, real-world decision-making is highly complex and involves multiple steps. Increasing the complexity of the tasks might therefore have the dual purpose of making the tasks more realistic and increasing the amount of data gathered.

5.5 The importance of subjectivity

If inferring knowledge about people from behavioural tasks becomes a widely used practice, there may be a temptation to think of psychiatric problems purely in terms of measurable learning and decision-making processes, forgetting the phenomenological experience of mental illness. This approach treats personal experience with a double whammy of reductionism: it first aims to reduce highly personal experiences to behaviour, and then further aims to reduce that behaviour to latent variables. Taken too far, I believe such an approach can become both unhelpful and unethical: unhelpful because we might discard crucial information, and unethical because we may be tempted to reduce people to computational processes and ignore their agency and experience.

Let us consider the practical side before we move on to the moral side. An overly reductionist approach could limit the search for treatments by leaving out important information. It is worth reminding ourselves that all models are wrong (Box, 1979), in the sense that models are simplifications that leave out varying amounts of detail, and sometimes we will leave out details that are important for solving a problem. This is not to say that we should not create models, but rather that we should be conscious of their limitations. It makes sense to me occasionally to return to the "raw data" - how an individual experiences their illness and how they interact with the real world - and check that we are including the most relevant information in our models. Since my guess is that my readers will be already convinced of the value of measuring real-world behaviour (Nickels et al., 2021), I will only discuss the potential value of personal experience here.

I think it is worth exploring the possibility that listening to people's stories, told in their own words, might make us more effective at helping them even if we consider ourselves scientists, not therapists. Patients' stories are certainly already being used to help them in some contexts. It is common knowledge that clinical psychologists and psychiatrists listen to people's stories in their own words. These professionals then handle this "free-form data" in various ways, depending on factors like their theoretical orientation, what approach they believe will be helpful in a given situation, and what is required by regulations. Sometimes they do no "data processing" at all; they simply listen to the story. Other times they use the story to infer symptoms and use those symptoms to make a diagnosis. Or, if they are applying a psychodynamic lens to the story, they might make inferences about underlying psychological processes. In a qualitative research context, they might apply qualitative coding to the data and in that way uncover underlying themes (Medelyan, n.d.).

In a behavioural modelling context, researchers apply the above techniques to various degrees. In some cases they do not gather free-form data (i.e. stories) from participants at all; they get participants to input data in an already-structured format such as by filling in a rating scale and performing a task. They might also perform a structured interview that allows free-form responses, but then abstract away most of the detail by summarising the findings according to a decided structure. For example, someone might tell a story about several horrible nights of being unable to sleep, and this gets recorded simply as 'insomnia'. Regardless of the details of the process followed in a particular behavioural modelling study, a lot of information gets discarded. The tone and posture of the insomnia sufferer might give information about how badly the experience affected their life, and they may give details about their experience, such as how many times a night they woke up or how long it took them to fall asleep, that don't fit into the structured data format.

How can we identify which information is important so that we don't discard clinically useful information? Natural language processing (NLP) might be a way to analyse vast amounts of free-form data and identify features of that data (B. L. Cook et al., 2016). Maybe we are heading for a future where we can come up with new psychodynamic theories with the aid of deep learning models: instead of "making up" important features based on experience and intuition, like when psychodynamic theorists claim that childhood experience X can result in some latent characteristic Y in the person's personality, which results in behaviour Z, we can let our models come up with features based on more data than an individual therapist can reasonably store in their head. Perhaps one day we will be able to talk to a bot about our lives and have it analyse

the themes in our story to give us feedback on what we can do to improve certain aspects of our lives. Of course, whether we would want to do so and whether we would get the same emotional benefits from the relationship, are different questions.

5.6 Ethical implications

Even if we are engaged in an area of research where patients' subjective perspectives are not practically useful, these perspectives are still worth considering in the spirit of ethical AI (Floridi et al., 2018). Firstly, as I discussed above, considering such perspectives may help us help people more than we would otherwise, and helping people more is clearly ethically desirable. But, somewhat separately from the issue of what works, there is also the issue of how people want to be treated. It seems plausible that many (perhaps most) people would prefer their mental health professionals to have personal interaction with them and give them the opportunity to talk about their experiences. If this type of interaction makes people feel valued and understood, I believe there is value in it for that reason alone. I am not proposing that we value that type of interaction so highly that we spend no time collecting data in other ways, but rather that we should use all the approaches at our disposal. These approaches include structured or unstructured interviews, self-report forms, and behavioural tasks.

Outside the boundaries of psychiatry, behavioural modelling also holds great potential for better understanding people and the ways they interact with the world around them. If the field were to get to a point where we could make inferences about the way someone would behave in real-world scenarios based on their behaviour in electronic games, that would give those who collect such data enormous power. For example, given knowledge about how someone responds to positively versus negatively valenced information, one might be able to target advertisements to that person that exploit either approach or withdrawal behaviour to optimise the chance of them buying a given product. If the field advanced to the point where repeatedly playing specially designed games (or even repeatedly interacting with websites and apps that are not overtly game-based) could actually change the way someone responded to stimuli in the real world, the potential impact would be greater still. On the negative side, there would be the potential to change the behaviour of users of apps and services in ways that optimise revenue for the creators of those apps and services at the expense of the users' well-being. Something like this may be happening already through social media platforms ("The Social Dilemma", 2020). Carl Rogers may have foreshadowed this back in 1961 when he said, 'We can choose to use our

growing knowledge to enslave people in ways never dreamed of before, depersonalizing them, controlling them by means so carefully selected that they will perhaps never be aware of their loss of personhood.' On the other hand, there would be great potential to change people's responses to stimuli in ways that would help them interact more optimally with their environments and achieve their goals. Rogers (1961) asked the question of whether science can 'discover the methods by which man can most readily become a continually developing and self-transcending process, in his behavior, his thinking, his knowledge'. It seems that we are getting closer to being able to answer 'yes'. It is important that we think about the potential impacts of this field and take steps to ensure that our advances in this field improve the well-being of people instead of harming it.

Chapter 6

Conclusions

6.1 Summary

This dissertation has been concerned with the use of reinforcement learning models to gain insight into learning and decision-making processes surrounding mental health. It had three aims: firstly, to explain the technicalities of fitting reinforcement learning models to behavioural data; secondly, to discuss which reinforcement learning parameters are important and how they differ between those with and without mental illness; and thirdly, to make suggestions for how reinforcement learning parameters might play a role in the classification of mental illness. Chapters 2 and 3 addressed the first aim. Chapter 2 started with an introduction to reinforcement learning, and then highlighted a number of algorithms that would be particularly relevant later. These included various flavours of Rescorla Wagner (RW) and a version of temporal difference (TD) learning that is useful for predicting neural firing when prediction errors occur. Chapter 3 covered the model fitting and comparison techniques used to evaluate the performance of models on various tasks. This is a key chapter because it shows that models and the parameters they include are not chosen in a haphazard way, but are arrived at through a systematic process. If we are to start using reinforcement learning parameters in clinical applications, it is crucial that our choices of models and parameters are based in sound scientific reasoning. Chapter 4 addressed the second aim of the dissertation, surveying findings from studies that have fitted reinforcement learning models to behavioural data. It surveyed differences in reinforcement learning parameters between healthy individuals and those with mental illness, focusing particularly on Pavlovian bias, learning rate and reward sensitivity. I will discuss the answers we found in the next section. Chapter 5 addressed my last aim, arguing that findings like these can be potentially useful in designing new diagnostic systems for mental disorders.

6.2 Going back to the research questions

Let me summarise some of the findings from chapter 4 and relate them to the questions raised in the introduction.

With regards to which reinforcement learning model(s) fit the data the best, there were no clear answers, but a few general points can be made. Firstly, Huys, Cools, et al. (2011) found that their best-fitting model had separate fixed general bias parameters towards 'go' for approach and withdrawal, suggesting a difference in the degree of bias towards 'go' in approach and withdrawal respectively. Secondly, the evidence for separate versus common learning rates for reward and punishment respectively is ambiguous, with Guitart-Masip et al. (2012) and Huys, Cools, et al. (2011) finding that a common learning rate fit the data best, while Mkrtchian et al. (2017) found that separate learning rates for reward and punishment worked best. For separate versus common feedback sensitivities for reward and punishment the evidence is also conflicting: Huys, Cools, et al. (2011) and Mkrtchian et al. (2017) found that separate reward and punishment sensitivities fit the data best, while Guitart-Masip et al.'s 2012 best model had a common feedback sensitivity for reward and punishment.

Another question raised in the introduction is which, if any, reinforcement learning parameters differ between healthy and depressed individuals. Unfortunately, the studies on Pavlovian-instrumental transfer (PIT) that compared healthy and depressed individuals did not fit reinforcement learning models to their data, so I could not draw any conclusions about Pavlovian bias parameters from them. However, a general finding that transpired from section 4.2.4 is that unmedicated patients with MDD showed a stronger PIT effect than healthy controls, and that this effect is driven by stronger responses to negatively, not positively, valenced Pavlovian stimuli (Nord et al., 2018). When it comes to learning rate and reward sensitivity, we saw in section 4.3 that the evidence is clearer (Huys et al., 2013): individuals with high levels of anhedonia have lower reward sensitivity than individuals with low levels of anhedonia, while there does not seem to be a significant difference in learning rate between those groups. These are unfortunately the only examples I have found, but, given the rapid growth of the field of computational psychiatry, I hope that many more examples will be discovered in the coming years.

The questions related to the third goal raised in the introduction, exploring implications for classification of mental illness, were discussed in chapter 5. Section 5.3 suggested that it might be feasible for reinforcement learning parameters to play a role in diagnosis, either by playing the role of dimensions to be rated in a dimensional diagnostic system or by qualifying as symptoms

in a categorical system when their values fall outside certain ranges. I suggested that it might be possible to use principal component analysis (PCA) to identify the most relevant parameters to avoid having to consider an impractical number of them. Finally, I suggested that data for finding values for these parameters could potentially be obtained through electronic games.

6.3 Limitations and suggestions for future research

This dissertation has only begun to explore the potential for reinforcement learning models to help us make sense of behavioural data and to improve the ways we classify mental illness. It left out many potentially relevant topics due to time and space constraints.

In reviewing literature for this dissertation, I focused on papers that fitted model-free reinforcement learning algorithms to data and explored the resulting parameter values. I paid particular attention to those that compared reinforcement learning parameters between individuals with and without depression, and left out many that discussed other disorders. Given that one of the aims of this field of research is to move beyond categorical diagnosis, it would make sense in future to select studies not based on a categorical diagnosis like MDD, but rather based on which transdiagnostic dimension, such as anhedonia, is being studied. This would be challenging because many studies still focus on categorical diagnoses, so one would have to look carefully at which aspects of a disorder were being studied and infer from there which transdiagnostic dimensions are relevant. Such an approach would be worthwhile because it would explore the full scope of variation in a particular domain of functioning instead of narrowly focusing on how that domain is impaired in a particular categorical diagnosis.

I argued in chapter 5 that in future we might use the values of latent (for example reinforcement learning) variables as part of a diagnostic system. This would avoid some of the problems with categorical diagnostic systems, and techniques such as PCA could help limit the number of characteristics for researchers and clinicians to assess. However, the research domain criteria (RDoC) framework (Cuthbert & Insel, 2013) has already taken the first steps towards implementing such an approach, and I did not delve deeply into what researchers have already done in that area. In future, researchers should continue to explore how reinforcement learning variables that have been studied in the field of computational psychiatry relate to existing RDoC dimensions, and, in cases where a variable is not addressed in RDoC, consider making proposals to include it.

Using reinforcement learning variables as part of assessing the way a person interacts with the world would of course entail finding values for those variables. I suggested earlier that data for this could be gathered through letting people play electronic games. A future avenue of research would then be to design games that are both engaging and that reliably measure the parameters of interest. There may be a trade-off between making a game fun to play and accurately and reliably measuring what one wants to measure. One would have to study carefully the impact that new features in the game design have on the results it gives. Another area that would be worth exploring is whether repeated game-play can change the values of some of these variables and whether this might translate into real-world changes in behaviour.

In order to model decision-making on multi-step tasks like electronic games, one would need reinforcement learning algorithms that distinguish between time steps within trials. Rescorla Wagner would therefore need to be replaced with algorithms like temporal difference learning. Due to constraints on the scope of this dissertation, I left out versions of TD algorithms (for example SARSA(λ)) used to model behaviour in multi-step tasks, but it would be valuable to include them in future work examining more complex tasks. Deciding exactly which algorithm to use where would be an open question and deserving of research.

6.4 Final thoughts

The study of RL parameters in mental illness is still new. Existing studies are few and often seem to contradict one another. This dissertation has aimed to survey the limited conclusions that have been drawn, as well as to introduce readers to the techniques used in such studies in the hope that they might eventually use those techniques to carry out studies of their own. In chapter 5, I tabled some ideas on what the future of mental health diagnosis and treatment might look like if this field continues to develop.

However, I do not think it is likely that RL model fitting alone is going to have a revolutionary impact on psychiatry. More likely, it will be a tiny piece of an intricate clockwork of insights and techniques from multiple fields. I have found it personally helpful to put a magnifying glass to this unfinished piece of machinery and study it in detail, not because I believe it alone is going to fix everything, but because it reassured me that incremental progress is being made. And even if one leaves the progress aside, it's encouraging to know that people are trying.

The following quote is from my favourite fan-fiction, *Harry Potter and the Methods of Rationality* (Yudkowski, 2015). For our purposes here, we can consider Dementors metaphors

for depression.

> Harry's eyes went back to the tall tattered cloak, almost absentmindedly, and without really being aware of what he was speaking, Harry said, "It shouldn't ought to exist."
>
> "Ah," said a dry, precise voice. "I thought you might say that. I am very sorry to tell you, Mr. Potter, that Dementors cannot be killed. Many have tried."
>
> "Really?" Harry said, still absentmindedly. "What did they try?"
>
> ...
>
> "Hm," Harry said. "Suppose you threw it into the Sun? Would it be destroyed?"
>
> "Throw it into the Sun?" squeaked Professor Flitwick, looking like he wanted to faint.
>
> "It seems unlikely, Mr. Potter," Professor Quirrell said dryly. "The Sun is very large, after all; I doubt the Dementor would have much effect on it. But it is not a test I would like to try, Mr. Potter, just in case."

Perhaps throwing a Dementor into the sun wasn't the most practical idea, and perhaps the ideas in this dissertation aren't either. But if we generate enough ideas (Brewer, 2019), maybe eventually we'll come up a few that have Dementor-destroying powers.

Appendix A

Derivation of expectation-maximisation update equations

A.1 Getting to the EM starting equation

With minor differences, this section follows the process in Dellaert (2002).

Huys, Cools, et al. (2011) found their hyperparameters by maximising $p(\mathcal{A}|\boldsymbol{\theta})$ with respect to $\boldsymbol{\theta}$, where $\mathcal{A}$ is the collection of action sequences for all the participants. To do this, they had to take into account the latent reinforcement learning parameters used to compute the probability of each action. The vector of reinforcement learning parameters for participant i is denoted later in this appendix by $\mathbf{h}_i$, but in the meantime it will be simpler to refer to the collection of all $\mathbf{h}_i$ values as $\mathcal{H}$.

We want to maximise $p(\mathcal{A}|\boldsymbol{\theta})$, but this is the same as maximising $\log p(\mathcal{A}|\boldsymbol{\theta})$. Since the latter is easier[1], that is what we will do. We marginalise over the latent variables $\mathcal{H}$ and then introduce a new probability distribution $f(\mathcal{H})$:

$$\begin{aligned}\log p(\mathcal{A}|\boldsymbol{\theta}) &= \log \int d\mathcal{H} p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) \\ &= \log \int d\mathcal{H} f(\mathcal{H}) \frac{p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta})}{f(\mathcal{H})}\end{aligned}$$

We now want to write down a function $B(\boldsymbol{\theta}, \boldsymbol{\theta}^{(n)})$ that is always going to be smaller than the above function and that will be easier to maximise. Jensen's inequality allows us to do this:

$$B(\boldsymbol{\theta}, \boldsymbol{\theta}^{(n)}) = \int d\mathcal{H} f(\mathcal{H}) \log \frac{p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta})}{f(\mathcal{H})} \leq \log \int d\mathcal{H} f(\mathcal{H}) \frac{p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta})}{f(\mathcal{H})}$$

[1] Logs turn products into sums, for example.

We now maximise $B(\boldsymbol{\theta}, \boldsymbol{\theta}^{(n)})$ instead of our original function. We do this using the method of Lagrange multipliers, keeping in mind the constraint that $\int_{\mathcal{H}} d\mathcal{H} f(\mathcal{H}) = 1$ because $f(\mathcal{H})$ is a probability distribution.

We can write a Lagrangian function with $f(\mathcal{H})$ and a Lagrange multiplier λ as parameters:

$$\begin{aligned}\mathcal{L}(f(\mathcal{H}), \lambda) &= B(\boldsymbol{\theta}, \boldsymbol{\theta}^{(n)}) + \lambda \left(1 - \int d\mathcal{H} f(\mathcal{H})\right) \\ &= \int d\mathcal{H} f(\mathcal{H}) \log \frac{p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta})}{f(\mathcal{H})} + \lambda \left(1 - \int d\mathcal{H} f(\mathcal{H})\right) \\ &= \int d\mathcal{H} f(\mathcal{H}) \log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) - \int d\mathcal{H} f(\mathcal{H}) \log f(\mathcal{H}) + \lambda \left(1 - \int d\mathcal{H} f(\mathcal{H})\right)\end{aligned} \tag{A.1}$$

Differentiating equation A.1 with respect to $f(\mathcal{H})$ gives

$$\frac{\partial \mathcal{L}}{\partial f(\mathcal{H})} = -\lambda + \log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}^{(n)}) - \log f(\mathcal{H}) - 1$$

Setting this to zero and solving for $f(\mathcal{H})$ gives

$$f(\mathcal{H}) = e^{-\lambda - 1} p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}^{(n)}). \tag{A.2}$$

Substituting this into our constraint $\int d\mathcal{H} f(\mathcal{H}) = 1$ gives

$$\begin{aligned} & e^{-\lambda-1} \int d\mathcal{H} p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}^{(n)}) = 1 \\ \implies \qquad & e^{-\lambda-1} = \frac{1}{\int d\mathcal{H} p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}^{(n)})} \end{aligned}$$

Substituting the above back into equation A.2 gives

$$f(\mathcal{H}) = \frac{p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}^{(n)})}{\int d\mathcal{H} p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}^{(n)})} = p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \tag{A.3}$$

This means that $B(\boldsymbol{\theta}, \boldsymbol{\theta}^{(n)})$ becomes

$$\begin{aligned} B(\boldsymbol{\theta}, \boldsymbol{\theta}^{(n)}) &= \int d\mathcal{H} p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \log \frac{p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta})}{p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)})} \\ &= \int d\mathcal{H} p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) - \int d\mathcal{H} p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \log p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \end{aligned} \tag{A.4}$$

Since the second term in equation A.4 is independent of $\boldsymbol{\theta}^{(n)}$, only the first term is relevant when maximising $B(\boldsymbol{\theta}, \boldsymbol{\theta}^{(n)})$ with respect to $\boldsymbol{\theta}$. Therefore the maximisation (M) step of the EM algorithm involves finding

$$\boldsymbol{\theta}^{(n+1)} = \underset{\boldsymbol{\theta}}{\text{argmax}} \int d\mathcal{H} p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) \tag{A.5}$$

A.2 Deriving the update equations for the mean and variance of the prior

Equations we're aiming for:

$$\boldsymbol{\mu}^{(n)} = \frac{1}{N} \sum_i \mathbf{m}_i^{(n)}$$

$$(\boldsymbol{\nu}^{(n)})^2 = \frac{1}{N} \sum_i \left[(\mathbf{m}_i^{(n)})^2 + \boldsymbol{\Sigma}_i^{(n)} \right] - (\boldsymbol{\mu}^{(n)})^2$$

We want to maximise equation A.5.

We need to differentiate the integral in equation A.5, set the derivative to zero and solve for $\boldsymbol{\theta}$ in order to find the value of $\boldsymbol{\theta}$ for which the expression is a maximum. Because $p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)})$ depends on $\boldsymbol{\theta}^{(n)}$, not $\boldsymbol{\theta}$,

$$\partial_{\boldsymbol{\theta}} \int d\mathcal{H}\ p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) = \int d\mathcal{H}\ p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \partial_{\boldsymbol{\theta}} \log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) \tag{A.6}$$

In the box below, I rewrite $\log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta})$ in a different form and differentiate it.

$$\log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) = \log \prod_{j=1}^{N} p(\mathbf{A}_j, \mathbf{h}_j|\boldsymbol{\theta}) \tag{A.7}$$

$$= \sum_{j=1}^{N} \log p(\mathbf{A}_j, \mathbf{h}_j|\boldsymbol{\theta}) \tag{A.8}$$

By chain rule: $\boxed{p(\mathbf{A}_j, \mathbf{h}_j|\boldsymbol{\theta}) = p(\mathbf{A}_j|\mathbf{h}_j, \boldsymbol{\theta})p(\mathbf{h}_j|\boldsymbol{\theta})}$ Once we are given some $\boldsymbol{h}_j$, specifying $\boldsymbol{\theta}_j$ gives no additional information, so

$$p(\mathbf{A}_j|\mathbf{h}_j, \boldsymbol{\theta}) = p(\mathbf{A}_j|\mathbf{h}_j)$$

So

$$\log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) = \sum_{j=1}^{N} \log p(\mathbf{A}_j|\mathbf{h}_j)p(\mathbf{h}_j|\boldsymbol{\theta}) \tag{A.9}$$

Now we differentiate:

$$\partial_{\boldsymbol{\theta}} \log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) = \sum_{j=1}^{N} \left(\cancelto{0}{\partial_{\boldsymbol{\theta}} \log p(\mathbf{A}_j|\mathbf{h}_j)} + \partial_{\boldsymbol{\theta}} \log p(\mathbf{h}_j|\boldsymbol{\theta}) \right)$$

$$= \sum_{j=1}^{N} \frac{\partial_{\boldsymbol{\theta}} p(\mathbf{h}_j|\boldsymbol{\theta})}{p(\mathbf{h}_j|\boldsymbol{\theta})} \tag{A.10}$$

We can also write $p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)})$ in a different way:
$p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)})$ is actually a product of the probabilities $p(\mathbf{h}_i|\mathbf{A}_i, \boldsymbol{\theta}_n)$ for all individuals i,

$$p(\mathcal{H}|\mathbf{A}, \boldsymbol{\theta}^{(n)}) = \prod_{i=1}^{N} p(\mathbf{h}_i|\mathbf{A}_i, \boldsymbol{\theta}_n) \tag{A.11}$$

We assume that each $p(\mathbf{h}_i|\mathbf{A}_i, \boldsymbol{\theta}_n)$ is actually a normal distribution with mean $\mathbf{m}_i$ and variance $\boldsymbol{\Sigma}_i$,

$$p(\mathcal{H}|\mathbf{A}, \boldsymbol{\theta}^{(n)}) \approx \prod_{i=1}^{N} \mathcal{N}(\mathbf{m}_i, \boldsymbol{\Sigma}_i)$$

Using the ingredients in the above boxes, equation A.6 becomes

$$\partial_{\boldsymbol{\theta}} \int d\mathcal{H}\ p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \log p(\mathcal{A}, \mathcal{H}|\boldsymbol{\theta}) = \int d\mathbf{h}_1 \cdots d\mathbf{h}_N \prod_{i=1}^{N} \mathcal{N}(\mathbf{m}_i, \Sigma_i) \sum_{j=1}^{N} \frac{\partial_{\boldsymbol{\theta}} p(\mathbf{h}_j|\boldsymbol{\theta})}{p(\mathbf{h}_j|\boldsymbol{\theta})}, \tag{A.12}$$

keeping in mind that $\mathcal{N}(\mathbf{m}_i, \boldsymbol{\Sigma}_i)$ is a function of $\mathbf{h}_i$.

Dealing with $p(\mathbf{h}_j|\boldsymbol{\theta})$ and $\partial_{\boldsymbol{\theta}} p(\mathbf{h}_j|\boldsymbol{\theta})$...
$p(\mathbf{h}_j|\boldsymbol{\theta})$ is a Gaussian with parameters $\boldsymbol{\theta}$, where $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\nu}^2)$, so I can write

$$p(\mathbf{h}_j|\boldsymbol{\theta}) = \frac{1}{\sqrt{2\pi\boldsymbol{\nu}^2}} e^{-\frac{(\boldsymbol{\mu}-h_j)^2}{2\boldsymbol{\nu}^2}}.$$

Then we need to differentiate with respect to $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{\nu}^2)$. Let us start by differentiating with respect to $\boldsymbol{\mu}$...

$$\partial_{\boldsymbol{\mu}} p(\mathbf{h}_j|\boldsymbol{\theta}) = \frac{-2(\boldsymbol{\mu} - \mathbf{h}_j)}{2\boldsymbol{\nu}^2} \frac{1}{\sqrt{2\pi\boldsymbol{\nu}^2}} e^{-\frac{(\boldsymbol{\mu}-h_j)^2}{2\boldsymbol{\nu}^2}} \tag{A.13}$$

$$= \frac{-(\boldsymbol{\mu} - \mathbf{h}_j)}{\boldsymbol{\nu}^2} p(\mathbf{h}_j|\boldsymbol{\theta}) \tag{A.14}$$

$$\tag{A.15}$$

Now differentiate with respect to $\boldsymbol{\nu}^2$:

$$\begin{aligned}
\partial_{\boldsymbol{\nu}^2} p(\mathbf{h}_j|\boldsymbol{\theta}) &= \frac{1}{\sqrt{2\pi\boldsymbol{\nu}^2}\boldsymbol{\nu}} e^{-\frac{(\boldsymbol{\mu}-\mathbf{h}_j)^2}{2\boldsymbol{\nu}^2}} + \frac{1}{\sqrt{2\pi\boldsymbol{\nu}^2}} \left(\frac{(\boldsymbol{\mu} - \mathbf{h}_j)^2}{\boldsymbol{\nu}^3} e^{-\frac{(\boldsymbol{\mu}-\mathbf{h}_j)^2}{2\boldsymbol{\nu}^2}} \right) \\
&= \frac{1}{\sqrt{2\pi\boldsymbol{\nu}^2}} e^{-\frac{(\boldsymbol{\mu}-\mathbf{h}_j)^2}{2\boldsymbol{\nu}^2}} \left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_j)^2}{\boldsymbol{\nu}^3} \right) \\
&= p(\mathbf{h}_j|\boldsymbol{\theta}) \left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_j)^2}{\boldsymbol{\nu}^3} \right)
\end{aligned} \tag{A.16}$$

For the derivative $\partial_{\boldsymbol{\mu}} p(\mathbf{h}_j|\boldsymbol{\theta})$, Equation A.12 becomes

$$\partial_{\boldsymbol{\theta}} \int d^N\mathbf{h}\; p(\mathbf{h}|\mathbf{A}, \boldsymbol{\theta}_n) \log p(\mathbf{A}, \mathbf{h}|\boldsymbol{\theta}) = \int d^N\mathbf{h} \prod_{i=1}^{N} \mathcal{N}(\mathbf{m}_i, \Sigma_i) \sum_{j=1}^{N} \frac{\partial_{\boldsymbol{\theta}} p(\mathbf{h}_j|\boldsymbol{\theta})}{p(\mathbf{h}_j|\boldsymbol{\theta})} \tag{A.17}$$

$$= \int d^N\mathbf{h} \prod_{i=1}^{N} \mathcal{N}(\mathbf{m}_i, \Sigma_i) \sum_{j=1}^{N} \frac{\frac{-(\boldsymbol{\mu}-\mathbf{h}_j)}{\boldsymbol{\nu}^2} p(\mathbf{h}_j|\boldsymbol{\theta})}{p(\mathbf{h}_j|\boldsymbol{\theta})} \tag{A.18}$$

$$= \int d^N\mathbf{h} \prod_{i=1}^{N} \mathcal{N}(\mathbf{m}_i, \Sigma_i) \sum_{j=1}^{N} \frac{-(\boldsymbol{\mu} - \mathbf{h}_j)}{\boldsymbol{\nu}^2} \tag{A.19}$$

Now we need to solve the above integral, but because this is confusing to do for $N > 2$, I will do it for $N = 2$.

The above integral for $N = 2$:

$$\begin{aligned}
&\int d^N\mathbf{h} \prod_{i=1}^{N} \mathcal{N}(\mathbf{m}_i, \Sigma_i) \sum_{j=1}^{N} \frac{-(\boldsymbol{\mu} - \mathbf{h}_j)}{\boldsymbol{\nu}^2} \\
&= -\int d\mathbf{h}_1 d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_1, \Sigma_1)\mathcal{N}(\mathbf{m}_2, \Sigma_2) \left(\frac{(\boldsymbol{\mu} - \mathbf{h}_1) + (\boldsymbol{\mu} - \mathbf{h}_2)}{\boldsymbol{\nu}^2} \right) \\
&= -\frac{1}{\boldsymbol{\nu}^2} \int d\mathbf{h}_1 d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_1, \Sigma_1)\mathcal{N}(\mathbf{m}_2, \Sigma_2) \left((\boldsymbol{\mu} - \mathbf{h}_1) + (\boldsymbol{\mu} - \mathbf{h}_2) \right) \\
&= -\frac{1}{\boldsymbol{\nu}^2} \left(\int d\mathbf{h}_1 d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_1, \Sigma_1)\mathcal{N}(\mathbf{m}_2, \Sigma_2)(\boldsymbol{\mu} - \mathbf{h}_1) + \int d\mathbf{h}_1 d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_1, \Sigma_1)\mathcal{N}(\mathbf{m}_2, \Sigma_2)(\boldsymbol{\mu} - \mathbf{h}_2) \right) \\
&= -\frac{1}{\boldsymbol{\nu}^2} \left(\int d\mathbf{h}_1 \mathcal{N}(\mathbf{m}_1, \Sigma_1)(\boldsymbol{\mu} - \mathbf{h}_1) \int d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_2, \Sigma_2) + \int d\mathbf{h}_1 \mathcal{N}(\mathbf{m}_1, \Sigma_1) \int d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_2, \Sigma_2)(\boldsymbol{\mu} - \mathbf{h}_2) \right)
\end{aligned} \tag{A.20}$$

Integrating a normal distribution with respect to its own variable gives 1, so $\int d\mathbf{h}_1 \mathcal{N}(\mathbf{m}_1, \Sigma_1)(\boldsymbol{\mu} - \mathbf{h}_2) = \boldsymbol{\mu} - \mathbf{h}_2$.

Thus, carrying on from Equation A.20,

$$= -\frac{1}{\boldsymbol{\nu}^2} \left(\int d\mathbf{h}_1 \mathcal{N}(\mathbf{m}_1, \Sigma_1)(\boldsymbol{\mu} - \mathbf{h}_1) + \int d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_2, \Sigma_2)(\boldsymbol{\mu} - \mathbf{h}_2) \right) \tag{A.21}$$

Now we deal with the terms in brackets in Equation A.21:

$$\begin{aligned}
\int d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i)(\boldsymbol{\mu} - \mathbf{h}_i) &= \int d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i)\boldsymbol{\mu} - \int d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i)\mathbf{h}_i \\
&= \boldsymbol{\mu} - \int d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i)\mathbf{h}_i \\
&= \boldsymbol{\mu} - \int d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i)(\mathbf{h}_i - \mathbf{m}_i + \mathbf{m}_i) \\
&= \boldsymbol{\mu} - \int d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i)(\mathbf{h}_i - \mathbf{m}_i) - \mathbf{m}_i \int d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i) \\
&= \boldsymbol{\mu} - \int d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i)(\mathbf{h}_i - \mathbf{m}_i) - \mathbf{m}_i \\
&= \boldsymbol{\mu} - 0 - \mathbf{m}_i \\
&= \boldsymbol{\mu} - \mathbf{m}_i
\end{aligned}$$

$\int d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i)(\mathbf{h}_i - \mathbf{m}_i) = 0$

Carrying on from Equation A.21 gives

$$\begin{aligned} &= -\frac{1}{\boldsymbol{\nu}^2}\left(\boldsymbol{\mu} - \mathbf{m}_1 + \boldsymbol{\mu} - \mathbf{m}_2\right) \\ &= -\frac{1}{\boldsymbol{\nu}^2}\left(2\boldsymbol{\mu} - (\mathbf{m}_1 + \mathbf{m}_2)\right) \end{aligned} \tag{A.22}$$

We generalise the above to $N \in \mathbb{N}$ and conclude that

$$\partial_{\boldsymbol{\theta}} \int d^N\mathbf{h}\; p(\mathbf{h}|\mathbf{A}, \boldsymbol{\theta}_n) \log p(\mathbf{A}, \mathbf{h}|\boldsymbol{\theta}) = -\frac{1}{\boldsymbol{\nu}^2}\left(N\boldsymbol{\mu} - \sum_{i=1}^{N} \mathbf{m}_i\right)$$

We set the above to zero in order to maximise the integral:

$$\begin{aligned} -\frac{1}{\boldsymbol{\nu}^2}\left(N\boldsymbol{\mu} - \sum_{i=1}^{N} \mathbf{m}_i\right) &= 0 \\ \implies \qquad N\boldsymbol{\mu} &= \sum_{i=1}^{N} \mathbf{m}_i \\ \implies \qquad \boldsymbol{\mu} &= \frac{1}{N}\sum_{i=1}^{N} \mathbf{m}_i \end{aligned} \tag{A.23}$$

This is the first m-step equation in Huys, Cools, et al. (2011). Now we tackle the second. We now sub the derivative $\partial_{\boldsymbol{\nu}^2} p(\mathbf{h}_j|\boldsymbol{\theta})$ (Equation A.16) into Equation A.12:

$$\begin{aligned} \partial_{\boldsymbol{\theta}} \int d\mathcal{H}\; p(\mathcal{H}|\mathcal{A}, \boldsymbol{\theta}^{(n)}) \log p(\mathbf{A}, \mathbf{h}|\boldsymbol{\theta}) &= \int d\mathbf{h}_1 \cdots d\mathbf{h}_N \prod_{i=1}^{N} \mathcal{N}(\mathbf{m}_i, \Sigma_i) \sum_{j=1}^{N} \frac{\partial_{\boldsymbol{\theta}} p(\mathbf{h}_j|\boldsymbol{\theta})}{p(\mathbf{h}_j|\boldsymbol{\theta})} \\ &= \int d\mathbf{h}_1 \cdots d\mathbf{h}_N \prod_{i=1}^{N} \mathcal{N}(\mathbf{m}_i, \Sigma_i) \sum_{j=1}^{N} \frac{p(\mathbf{h}_j|\boldsymbol{\theta})\left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu}-\mathbf{h}_j)^2}{\boldsymbol{\nu}^3}\right)}{p(\mathbf{h}_j|\boldsymbol{\theta})} \\ &= \int d\mathbf{h}_1 \cdots d\mathbf{h}_N \prod_{i=1}^{N} \mathcal{N}(\mathbf{m}_i, \Sigma_i) \sum_{j=1}^{N} \left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_j)^2}{\boldsymbol{\nu}^3}\right) \end{aligned} \tag{A.24}$$

Now we solve integral A.24! Again, I will simplify things by setting $N = 2$ to start with. Then the integral becomes

$$\begin{aligned}
&\int d\mathbf{h}_1 \int d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_1, \Sigma_1)\mathcal{N}(\mathbf{m}_2, \Sigma_2) \left(\left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_1)^2}{\boldsymbol{\nu}^3} \right) + \left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_2)^2}{\boldsymbol{\nu}^3} \right) \right) \\
&= \int d\mathbf{h}_1 \mathcal{N}(\mathbf{m}_1, \Sigma_1) \int d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_2, \Sigma_2) \left(\left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_1)^2}{\boldsymbol{\nu}^3} \right) + \left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_2)^2}{\boldsymbol{\nu}^3} \right) \right) \\
&= \int_{-\infty}^{\infty} d\mathbf{h}_1 \mathcal{N}(\mathbf{m}_1, \Sigma_1) \left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_1)^2}{\boldsymbol{\nu}^3} \right) \int_{-\infty}^{\infty} d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_2, \Sigma_2) \\
&\quad + \int_{-\infty}^{\infty} d\mathbf{h}_1 \mathcal{N}(\mathbf{m}_1, \Sigma_1) \int_{-\infty}^{\infty} d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_2, \Sigma_2) \left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_2)^2}{\boldsymbol{\nu}^3} \right) \qquad \text{(A.25)}
\end{aligned}$$

As earlier, use that $\int_{-\infty}^{\infty} d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i) = 1$, and (A.25) becomes

$$\int_{-\infty}^{\infty} d\mathbf{h}_1 \mathcal{N}(\mathbf{m}_1, \Sigma_1) \left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_1)^2}{\boldsymbol{\nu}^3} \right) + \int_{-\infty}^{\infty} d\mathbf{h}_2 \mathcal{N}(\mathbf{m}_2, \Sigma_2) \left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_2)^2}{\boldsymbol{\nu}^3} \right) \quad \text{(A.26)}$$

The box below deals with $\int_{-\infty}^{\infty} d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i) \left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_i)^2}{\boldsymbol{\nu}^3} \right)$:

$$\begin{aligned}
&\int_{-\infty}^{\infty} d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i) \left(-\frac{1}{\boldsymbol{\nu}} + \frac{(\boldsymbol{\mu} - \mathbf{h}_i)^2}{\boldsymbol{\nu}^3} \right) \\
&= -\frac{1}{\boldsymbol{\nu}} \int_{-\infty}^{\infty} d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i) + \frac{1}{\boldsymbol{\nu}^3} \int_{-\infty}^{\infty} d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i)(\boldsymbol{\mu} - \mathbf{h}_i)^2 \\
&= -\frac{1}{\boldsymbol{\nu}} + \frac{1}{\boldsymbol{\nu}^3} \int_{-\infty}^{\infty} d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i)(\boldsymbol{\mu} - \mathbf{h}_i)^2 \\
&= -\frac{1}{\boldsymbol{\nu}} + \frac{1}{\boldsymbol{\nu}^3} \int_{-\infty}^{\infty} d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i) \left(\boldsymbol{\mu}^2 - 2\boldsymbol{\mu}\mathbf{h}_i + \mathbf{h}_i^2 \right) \qquad \text{(A.27)}
\end{aligned}$$

The following integrals will now be useful:

$$\int_{-\infty}^{\infty} d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i) \mathbf{h}_i = \mathbf{m}_i$$

and

$$\int_{-\infty}^{\infty} d\mathbf{h}_i \mathcal{N}(\mathbf{m}_i, \Sigma_i) \mathbf{h}_i^2 = \mathbf{m}_i^2 + \Sigma_i$$

Using these integrals allows (A.27) to become

$$-\frac{1}{\boldsymbol{\nu}} + \frac{\boldsymbol{\mu}^2}{\boldsymbol{\nu}^3} - 2\frac{\boldsymbol{\mu}\mathbf{m}_i}{\boldsymbol{\nu}^3} + \frac{\mathbf{m}_i^2 + \Sigma_i}{\boldsymbol{\nu}^3}$$

(A.26) becomes

$$\sum_{i=1}^{2} -\frac{1}{\boldsymbol{\nu}} + \frac{\boldsymbol{\mu}^2}{\boldsymbol{\nu}^3} - 2\frac{\boldsymbol{\mu}\mathbf{m}_i}{\boldsymbol{\nu}^3} + \frac{\mathbf{m}_i^2 + \Sigma_i}{\boldsymbol{\nu}^3}$$

for $N = 2$; for $N \in \mathbb{N}$ it becomes

$$\begin{aligned}
&\sum_{i=1}^{N} -\frac{1}{\boldsymbol{\nu}} + \frac{\boldsymbol{\mu}^2}{\boldsymbol{\nu}^3} - 2\frac{\boldsymbol{\mu}\mathbf{m}_i}{\boldsymbol{\nu}^3} + \frac{\mathbf{m}_i^2 + \Sigma_i}{\boldsymbol{\nu}^3} \\
&= -\frac{N}{\boldsymbol{\nu}} + \frac{N\boldsymbol{\mu}^2}{\boldsymbol{\nu}^3} - 2\frac{N\boldsymbol{\mu}\mathbf{m}_i}{\boldsymbol{\nu}^3} + \frac{1}{\boldsymbol{\nu}^3} \sum_{i=1}^{N} (\mathbf{m}_i^2 + \Sigma_i) \\
&= -\frac{N}{\boldsymbol{\nu}} - \frac{N\boldsymbol{\mu}^2}{\boldsymbol{\nu}^3} + \frac{1}{\boldsymbol{\nu}^3} \sum_{i=1}^{N} (\mathbf{m}_i^2 + \Sigma_i)
\end{aligned}$$

Setting the above equal to zero (to do the maximisation step) and doing some algebra, we get

$$0 = -\frac{N}{\boldsymbol{\nu}} - \frac{N\boldsymbol{\mu}^2}{\boldsymbol{\nu}^3} + \frac{1}{\boldsymbol{\nu}^3}\sum_{i=1}^{N}(\mathbf{m}_i^2 + \Sigma_i)$$

$$N\boldsymbol{\nu}^2 = -N\boldsymbol{\mu}^2 + \sum_{i=1}^{N}(\mathbf{m}_i^2 + \Sigma_i)$$

$$\boldsymbol{\nu}^2 = \frac{1}{N}\sum_{i=1}^{N}(\mathbf{m}_i^2 + \Sigma_i) - \boldsymbol{\mu}^2$$

which gives us the second maximisation equation in Huys, Cools, et al. (2011).

Appendix B

Relationship between the Hessian and covariance matrix of a Gaussian

The Hessian of the log of a multidimensional Gaussian is related to the covariance matrix as follows. For an m-dimensional Gaussian,

$$p(\mathbf{h}_i|\boldsymbol{\theta}) = \frac{1}{(2\pi)^{m/2}} \frac{1}{|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{h}_i - \boldsymbol{\mu})^\top \Sigma^{-1}(\mathbf{h}_i - \boldsymbol{\mu})\right)$$

where $\boldsymbol{\theta} = \langle \boldsymbol{\mu}, \boldsymbol{\nu} \rangle$.

We want to differentiate the log of this with respect to $\mathbf{h}_i$. I temporarily drop the subscript i (indicating the subject in question) to avoid confusion when I use index notation. To simplify matters further, let $F(\mathbf{h}) = (\mathbf{h} - \boldsymbol{\mu})^\top \Sigma^{-1}$. This gives

$$\begin{aligned}
\frac{\partial \log p(\mathbf{h}|\boldsymbol{\theta})}{\partial \mathbf{h}} &= -\frac{1}{2}\frac{\partial}{\partial \mathbf{h}} F(\mathbf{h})(\mathbf{h} - \boldsymbol{\mu}) \\
&= -\frac{1}{2}\left(\frac{\partial F(\mathbf{h})}{\partial \mathbf{h}}(\mathbf{h} - \boldsymbol{\mu}) + F(\mathbf{h})\frac{\partial (\mathbf{h} - \boldsymbol{\mu})}{\partial \mathbf{h}}\right) \\
&= -\frac{1}{2}\left(\frac{\partial F(\mathbf{h})}{\partial \mathbf{h}}(\mathbf{h} - \boldsymbol{\mu}) + F(\mathbf{h})\right) \qquad \text{(B.1)}
\end{aligned}$$

Let us deal with the derivative of F. First write F in index notation as

$$F_j = \sum_l (h_l - \mu_l)\Sigma^{-1}_{lj}$$

Then

$$\begin{aligned}
\frac{\partial F_j}{\partial h_k} &= \frac{\partial \left(\sum_l (h_l - \mu_l)\Sigma^{-1}_{lj}\right)}{\partial h_k} \\
&= \Sigma^{-1}_{kj} \qquad \text{(B.2)}
\end{aligned}$$

Therefore

$$\frac{\partial F(\mathbf{h})}{\partial \mathbf{h}} = \Sigma^{-1}$$

Continuing from equation B.1,

$$\frac{\partial \log p(\mathbf{h}|\boldsymbol{\theta})}{\partial \mathbf{h}} = -\frac{1}{2}\left(\Sigma^{-1}(\mathbf{h}-\boldsymbol{\mu}) + (\mathbf{h}-\boldsymbol{\mu})^{\top}\Sigma^{-1}\right)$$

Then the Hessian $\mathbf{H}$ of the log of $p(\mathbf{h}|\boldsymbol{\theta})$ is

$$\begin{aligned}\mathbf{H} = \frac{\partial^2 \log p(\mathbf{h}|\boldsymbol{\theta})}{\partial^2 \mathbf{h}} &= -\frac{1}{2}\left(\Sigma^{-1} + \Sigma^{-1}\right) \\ &= -\Sigma^{-1}\end{aligned}$$

and so the covariance matrix is

$$\Sigma = -\mathbf{H}^{-1}$$

Appendix C

Eligibility traces

Eligibility traces provide a way of letting a prediction error at time step t affect the value of states our agent visited more than one time step earlier. If we are not using function approximation, an eligibility trace is a vector or matrix of the same dimensions as the vector or matrix that keeps track of our values. If we are using function approximation, the eligibility trace is the same length as the weight vector (Sutton & Barto, 2020, p.287).

Imagine several CS inputs x_i (for example, one for every row of matrix 2.8) and a reward/punishment input going into a processing unit, generating an output. I like to imagine the inputs 'lighting up' at time steps when they are active (non-zero).

Now imagine keeping a record over time of which inputs have been active in the past. Every time an input lights up, some amount gets added to the record for that input. As time passes, the records fade away, so that recent events contribute more to the record. What we can do then is use this record to decide which weights to update instead of using only the inputs that are active this instant. This allows past events to influence weight updates to varying degrees based on how recently they occurred. We store this record as a vector that I will denote by $\boldsymbol{z}$.

If a component of $\mathbf{x}$ has recently been active, then that component of $\mathbf{w}$ is more likely than others to be updated in the near future. A factor λ determines how quickly the trace vector falls back to zero. (Sutton & Barto, 2020, p.287)

We update the eligibility trace $\mathbf{z}$ as follows:

$$\mathbf{z}_t = \gamma\lambda\mathbf{z}_{t-1} + \mathbf{x}$$

Eligibility traces are at the core of reinforcement learning methods like TD(λ) and SARSA(λ), but those methods are outside the scope of the current discussion.

Bibliography

American Psychiatric Association. (n.d.). *What is depression?* Retrieved February 2, 2022, from https://www.psychiatry.org/patients-families/depression/what-is-depression

American Psychiatric Association. (2013). *Diagnostic and statistical manual of mental disorders, fifth edition*. American Psychiatric Publishing.

Arroll, B., Macgillivray, S., Ogston, S., Reid, I., Sullivan, F., Williams, B., & Crombie, I. (2005). Efficacy and tolerability of tricyclic antidepressants and SSRIs compared with placebo for treatment of depression in primary care: A meta-analysis. *The Annals of Family Medicine*, *3*(5), 449–456.

Beck, A. T., & Alford, B. A. (2009). *Depression: Causes and treatment* (2nd ed.). University of Pennsylvania Press.

Beck, J. S. (2011). *Cognitive behavior therapy, second edition: Basics and beyond* (2nd ed.). Guilford Press.

Box, G. E. (1979). *Robustness in the strategy of scientific model building*. Elsevier.

Brewer, J. (2019). *More ideas means better ideas, says wetransfer's ideas report*. https://www.itsnicethat.com/news/wetransfer-ideas-report-creative-survey-2019-281119

Brosh, A. (2011). *Adventures in depression*. http://hyperboleandahalf.blogspot.com/2011/10/adventures-in-depression.html

Brosh, A. (2013). *Depression part two*. http://hyperboleandahalf.blogspot.com/2013/05/depression-part-two.html

Cartoni, E., Puglisi-Allegra, S., & Baldassarre, G. (2013). The three principles of action: A pavlovian-instrumental transfer hypothesis. *Frontiers in Behavioral Neuroscience*, *7*, 153.

Chase, H., Frank, M., Michael, A., Bullmore, E., Sahakian, B., & Robbins, T. (2010). Approach and avoidance learning in patients with major depression and healthy controls: Relation to anhedonia. *Psychological medicine*, *40*(3), 433.

Chen, C., Takahashi, T., Nakagawa, S., Inoue, T., & Kusumi, I. (2015). Reinforcement learning in depression: A review of computational research. *Neuroscience & Biobehavioral Reviews*, *55*, 247–267. https://doi.org/https://doi.org/10.1016/j.neubiorev.2015.05.005

Chen, Y., & Gupta, M. R. (2010). EM demystified: An expectation-maximization tutorial. *UWEE Technical Report Series*.

Cipriani, A., Furukawa, T. A., Salanti, G., Chaimani, A., Atkinson, L. Z., Ogawa, Y., Leucht, S., Ruhe, H. G., Turner, E. H., Higgins, J. P., et al. (2018). Comparative efficacy and acceptability of 21 antidepressant drugs for the acute treatment of adults with major depressive disorder: A systematic review and network meta-analysis. *Focus*, *16*(4), 420–429.

conjugateprior (https://stats.stackexchange.com/users/1739/conjugateprior). (2018, February 17). *About specifying independent priors for each parameter in bayesian modeling*. Retrieved February 11, 2022, from https://stats.stackexchange.com/q/329207

Cook, A., & Sheikh, A. (2000). Descriptive statistics (Part 2): Interpreting study results. *Primary Care Respiratory Journal*, *8*(1), 16–17.

Cook, B. L., Progovac, A. M., Chen, P., Mullin, B., Hou, S., & Baca-Garcia, E. (2016). Novel use of natural language processing (nlp) to predict suicidal ideation and psychiatric symptoms in a text-based mental health intervention in madrid. *Computational and Mathematical Methods in Medicine*, *2016*.

Cuthbert, B. N., & Insel, T. R. (2013). Toward the future of psychiatric diagnosis: the seven pillars of RDoC. *BMC medicine*, *11*(1), 1–8.

Daw, N. D. (2011). Trial-by-trial data analysis using computational models. *Decision making, affect, and learning: Attention and performance XXIII*, *23*(1).

Daw, N. D., Gershman, S. J., Seymour, B., Dayan, P., & Dolan, R. J. (2011). Model-based influences on humans' choices and striatal prediction errors. *Neuron*, *69*(6), 1204–1215. https://doi.org/10.1016/j.neuron.2011.02.027

Dayan, P., & Abbott, L. F. (2001). *Theoretical neuroscience: Computational and mathematical modeling of neural systems*. The MIT Press.

Dellaert, F. (2002). *The expectation maximization algorithm* (tech. rep.). College of Computing, Georgia Institute of Technology. https://smartech.gatech.edu/bitstream/handle/1853/3281/02-20.pdf

DeNero, J., & Klein, D. (2014, August 26). *Project 3: Reinforcement learning, UC Berkeley CS188 Intro to AI - course materials* (Version 1.001). http://ai.berkeley.edu/reinforcement.html

Floridi, L., Cowls, J., Beltrametti, M., Chatila, R., Chazerand, P., Dignum, V., Luetge, C., Madelin, R., Pagallo, U., Rossi, F., et al. (2018). Ai4people—an ethical framework for a good ai society: Opportunities, risks, principles, and recommendations. *Minds and Machines*, *28*(4), 689–707.

Frank, M. J., Seeberger, L. C., & O'reilly, R. C. (2004). By carrot or by stick: Cognitive reinforcement learning in parkinsonism. *Science*, *306*(5703), 1940–1943. https://doi.org/10.1126/science.1102941

Franken, I. H. A., Rassin, E., & Muris, P. (2007). The assessment of anhedonia in clinical and non-clinical populations: Further validation of the Snaith–Hamilton Pleasure Scale (SHAPS). *Journal of affective disorders*, *99*(1-3), 83–89.

Friston, K. J., Stephan, K. E., Montague, R., & Dolan, R. J. (2014). Computational psychiatry: The brain as a phantastic organ. *The Lancet Psychiatry*, *1*(2), 148–158. https://doi.org/10.1016/S2215-0366(14)70275-5

Gershman, S. J. (2016). Empirical priors for reinforcement learning models. *Journal of Mathematical Psychology*, *71*, 1–6.

Gillan, C. M., Kosinski, M., Whelan, R., Phelps, E. A., & Daw, N. D. (2016). Characterizing a psychiatric symptom dimension related to deficits in goal-directed control. *Elife*, *5*, e11305.

Gottesman, I. I., & Gould, T. D. (2003). The endophenotype concept in psychiatry: Etymology and strategic intentions. *American journal of psychiatry*, *160*(4), 636–645.

Gradin, V. B., Kumar, P., Waiter, G., Ahearn, T., Stickle, C., Milders, M., Reid, I., Hall, J., & Steele, J. D. (2011). Expected value and prediction error abnormalities in depression and schizophrenia. *Brain*, *134*(6), 1751–1764.

Guitart-Masip, M., Fuentemilla, L., Bach, D. R., Huys, Q. J. M., Dayan, P., Dolan, R. J., & Duzel, E. (2011). Action dominates valence in anticipatory representations in the human striatum and dopaminergic midbrain. *Journal of Neuroscience*, *31*(21), 7867–7875.

Guitart-Masip, M., Huys, Q. J. M., Fuentemilla, L., Dayan, P., Duzel, E., & Dolan, R. J. (2012). Go and no-go learning in reward and punishment: Interactions between affect and effect. *Neuroimage*, *62*(1), 154–166.

Henriques, J. B., Glowacki, J. M., & Davidson, R. J. (1994). Reward fails to alter response bias in depression. *Journal of Abnormal Psychology*, *103*(3), 460.

Huang, Y., Yaple, Z. A., & Yu, R. (2020). Goal-oriented and habitual decisions: Neural signatures of model-based and model-free learning. *NeuroImage*, *215*, 116834.

Huys, Q. J. M. (2016). *Fitting models to behaviour*. http://www.ccnss.org/wp-content/uploads/2016/07/rlfit.pdf

Huys, Q. J. M., Cools, R., Gölzer, M., Friedel, E., Heinz, A., Dolan, R. J., & Dayan, P. (2011). Disentangling the roles of approach, activation and valence in instrumental and pavlovian responding. *PLoS Comput Biol*, *7*(4), e1002028.

Huys, Q. J. M., Gölzer, M., Friedel, E., Heinz, A., Cools, R., Dayan, P., & Dolan, R. J. (2016). The specificity of pavlovian regulation is associated with recovery from depression. *Psychological medicine*, *46*(5), 1027–1035.

Huys, Q. J. M., Maia, T. V., & Frank, M. J. (2016). Computational psychiatry as a bridge from neuroscience to clinical applications. *Nature neuroscience*, *19*(3), 404–413. https://doi.org/10.1038/nn.4238

Huys, Q. J. M., Moutoussis, M., & Williams, J. (2011). Are computational models of any use to psychiatry? *Neural Networks*, *24*(6), 544–551. https://doi.org/10.1016/j.neunet.2011.03.001

Huys, Q. J. M., Pizzagalli, D. A., Bogdan, R., & Dayan, P. (2013). Mapping anhedonia onto reinforcement learning: A behavioural meta-analysis. *Biology of Mood & Anxiety Disorders*, *3*(1), 12.

Institute for Health Metrics and Evaluation. (2021). *GBD Compare*. Retrieved December 8, 2021, from https://vizhub.healthdata.org/gbd-compare/

Jackson, S. P. (1998). Bright star — black sky: A phenomenological study of depression as a window into the psyche of the gifted adolescent. *Roeper Review*, *20*(3), 215–221.

Jackson, S. P., & Peterson, J. (2003). Depressive disorder in highly gifted adolescents. *Journal of Secondary Gifted Education*, *14*(3), 175–186.

Jamison, K. R. (1996). *An unquiet mind: A memoir of moods and madness*. Vintage.

Jamison, K. R. (1999). *Night falls fast: Understanding suicide*. Alfred A. Knopf.

Kool, W., Cushman, F. A., & Gershman, S. J. (2016). When does model-based control pay off? *PLoS computational biology*, *12*(8), e1005090.

Kruschke, J. (2014). *Doing Bayesian data analysis: A tutorial with R, JAGS, and Stan*. Academic Press.

Kumar, P., Waiter, G., Ahearn, T., Milders, M., Reid, I., & Steele, J. (2008). Abnormal temporal difference reward-learning signals in major depression. *Brain*, *131*(8), 2084–2093.

Kumar, P., Goer, F., Murray, L., Dillon, D. G., Beltzer, M. L., Cohen, A. L., Brooks, N. H., & Pizzagalli, D. A. (2018). Impaired reward prediction error encoding and striatal-midbrain connectivity in depression. *Neuropsychopharmacology*, *43*(7), 1581–1588.

Kunisato, Y., Okamoto, Y., Ueda, K., Onoda, K., Okada, G., Yoshimura, S., Suzuki, S.-i., Samejima, K., & Yamawaki, S. (2012). Effects of depression on reward-based decision making and variability of action in probabilistic learning. *Journal of behavior therapy and experimental psychiatry*, *43*(4), 1088–1094.

Maia, T. V., & Frank, M. J. (2011). From reinforcement learning models to psychiatric and neurological disorders. *Nature Neuroscience*, *14*(2), 154–162. https://doi.org/10.1038/nn.2723

Maia, T. V., Huys, Q. J. M., & Frank, M. J. (2017). Theory-Based Computational Psychiatry. *Biological Psychiatry*, *82*(6), 382–384. https://doi.org/10.1016/j.biopsych.2017.07.016

Marr, D. (1982). *Vision: A computational investigation into the human representation and processing of visual information*. W. H. Freeman.

Medelyan, A. (n.d.). *Thematic analysis software: How it works and why you need it*. Retrieved August 31, 2022, from https://getthematic.com/insights/coding-qualitative-data/

Metts, A., Arnaudova, I., Staples-Bradley, L., Sun, M., Zinbarg, R., Nusslock, R., Wassum, K. M., & Craske, M. G. (2022). Disruption in pavlovian-instrumental transfer as a function of depression and anxiety. *Journal of Psychopathology and Behavioral Assessment*, 1–15.

Mkrtchian, A., Aylward, J., Dayan, P., Roiser, J. P., & Robinson, O. J. (2017). Modeling avoidance in mood and anxiety disorders using reinforcement learning. *Biological Psychiatry*, *82*(7), 532–539.

Mnih, V., Kavukcuoglu, K., Silver, D., Graves, A., Antonoglou, I., Wierstra, D., & Riedmiller, M. (2013). Playing Atari with deep reinforcement learning. *CoRR*, *abs/1312.5602*. http://arxiv.org/abs/1312.5602

Montague, P. R., Dayan, P., & Sejnowski, T. J. (1996). A framework for mesencephalic dopamine systems based on predictive hebbian learning. *Journal of Neuroscience*, *16*(5), 1936–1947.

Montague, P. R., Dolan, R. J., Friston, K. J., & Dayan, P. (2012). Computational psychiatry. *Trends in cognitive sciences*, *16*(1), 72–80. https://doi.org/10.1016/j.tics.2011.11.018

Nickels, S., Edwards, M. D., Poole, S. F., Winter, D., Gronsbell, J., Rozenkrants, B., Miller, D. P., Fleck, M., McLean, A., Peterson, B., et al. (2021). Toward a mobile platform for real-world digital measurement of depression: User-centered design, data quality, and behavioral and clinical modeling. *JMIR Mental Health*, *8*(8), e27589.

Nord, C. L., Lawson, R. P., Huys, Q. J. M., Pilling, S., & Roiser, J. P. (2018). Depression is associated with enhanced aversive pavlovian control over instrumental behaviour. *Scientific Reports*, *8*(1), 1–10.

Pizzagalli, D. A. (2014). Depression, stress, and anhedonia: Toward a synthesis and integrated model. *Annual Review of Clinical Psychology*, *10*, 393–423.

Pizzagalli, D. A., Iosifescu, D., Hallett, L. A., Ratner, K. G., & Fava, M. (2008). Reduced hedonic capacity in major depressive disorder: Evidence from a probabilistic reward task. *Journal of Psychiatric Research*, *43*(1), 76–87.

Reinen, J., Smith, E. E., Insel, C., Kribs, R., Shohamy, D., Wager, T. D., & Jarskog, L. F. (2014). Patients with schizophrenia are impaired when learning in the context of pursuing rewards. *Schizophrenia Research*, *152*(1), 309–310.

Reinen, J., Whitton, A. E., Pizzagalli, D. A., Slifstein, M., Abi-Dargham, A., McGrath, P. J., Iosifescu, D. V., & Schneier, F. R. (2021). Differential reinforcement learning responses to positive and negative information in unmedicated individuals with depression. *European Neuropsychopharmacology*, *53*, 89–100.

Robinson, O. J., & Chase, H. W. (2017). Learning and choice in mood disorders: Searching for the computational parameters of anhedonia. *Computational Psychiatry*, *1*, 208–233.

Rogers, C. R. (1961). *On becoming a person: A therapist's view of psychotherapy*. Houghton Mifflin Company: Boston.

Ross, S. (2009). *A first course in probability 8th edition*. Pearson.

Rush, A. J., Warden, D., Wisniewski, S. R., Fava, M., Trivedi, M. H., Gaynes, B. N., & Nierenberg, A. A. (2009). STAR*D: Revising conventional wisdom. *CNS Drugs*, *23*(8), 627–647.

Russek, E. M., Momennejad, I., Botvinick, M. M., Gershman, S. J., & Daw, N. D. (2017). Predictive representations can link model-based reinforcement learning to model-free mechanisms. *PLOS Computational Biology*, *13*(9), 1–35. https://doi.org/10.1371/journal.pcbi.1005768

Russell, S. J., & Norvig, P. (2010). *Artificial intelligence: A modern approach* (Third international ed.). Pearson Education, Upper Saddle River, NJ, USA.

Schultz, W., Dayan, P., & Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, *275*(5306), 1593–1599. https://doi.org/10.1126/science.275.5306.1593

Shea, S. C. (2017). *Psychiatric interviewing: The art of understanding*. Elsevier.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., et al. (2016). Mastering the game of go with deep neural networks and tree search. *Nature*, *529*(7587), 484–489.

The social dilemma. (2020).

Solomon, A. (2014). *The noonday demon: An atlas of depression*. Simon; Schuster.

Stephan, K. E., & Mathys, C. (2014). Computational approaches to psychiatry. *Current opinion in neurobiology*, *25*, 85–92. https://doi.org/10.1016/j.conb.2013.12.007

Sutton, R. S., & Barto, A. G. (1987). A temporal-difference model of classical conditioning. *Proceedings of the ninth annual conference of the cognitive science society*, 355–378.

Sutton, R. S., & Barto, A. G. (2020). *Reinforcement Learning: an introduction* (2nd ed.). MIT Press.

Thorndike, E. L. (1911). *Animal intelligence*. The Macmillan Company.

Treadway, M. T., Buckholtz, J. W., Schwartzman, A. N., Lambert, W. E., & Zald, D. H. (2009). Worth the 'EEfRT'? The effort expenditure for rewards task as an objective measure of motivation and anhedonia. *PloS One*, *4*(8), e6598.

Vigo, D., Thornicroft, G., & Atun, R. (2016). Estimating the true global burden of mental illness. *The Lancet Psychiatry*, *3*(2), 171–178.

Vrieze, E., Pizzagalli, D. A., Demyttenaere, K., Hompes, T., Sienaert, P., de Boer, P., Schmidt, M., & Claes, S. (2013). Reduced reward learning predicts outcome in major depressive disorder. *Biological Psychiatry*, *73*(7), 639–645.

Walker, E. R., McGee, R. E., & Druss, B. G. (2015). Mortality in mental disorders and global disease burden implications: A systematic review and meta-analysis. *The Lancet Psychiatry*, *72*(4), 334–341.

Wang, P. S., Angermeyer, M., Borges, G., Bruffaerts, R., Chiu, W. T., De Girolamo, G., Fayyad, J., Gureje, O., Haro, J. M., Huang, Y., et al. (2007). Delay and failure in treatment seeking after first onset of mental disorders in the world health organization's world mental health survey initiative. *World Psychiatry*, *6*(3), 177.

Wang, X.-J., & Krystal, J. H. (2014). Computational psychiatry. *Neuron*, *84*(3), 638–654. https://doi.org/10.1016/j.neuron.2014.10.018

Watson, D., & Clark, L. A. (1991). *The mood and anxiety symptom questionnaire* [University of Iowa, Department of Psychology, Iowa City].

WHO. (2021). *Disability-adjusted life years (DALYs)*. https://www.who.int/data/gho/indicator-metadata-registry/imr-details/158

World Health Organization. (2018). *International classification of diseases for mortality and morbidity statistics* (11th). https://icd.who.int/browse11/l-m/en

Yudkowski, E. (2015). *Harry potter and the methods of rationality*. Retrieved December 20, 2021, from https://cdn.rawgit.com/rjl20/hpmor/0c10d2e8b6bd68e88fd2fc6e6b233140917e7314/out/hpmor.pdf

Zhang, J. (2019). *Reinforcement learning - implement grid world: Introduction of value iteration*. Retrieved December 22, 2021, from https://towardsdatascience.com/reinforcement-learning-implement-grid-world-from-scratch-c5963765ebff