TÉCNICO LISBOA

Mechanistic characterization of reinforcement learning in healthy humans using computational models

Ângelo Rodrigo Neto Dias

Thesis to obtain the Master of Science Degree in

Biomedical Engineering

Supervisors: Prof. Tiago Vaz Maia

Prof. Patrícia Margarida Piedade Figueiredo

Examination Committee

Chairperson: Prof. Raúl Daniel Lavado Carneiro Martins
Supervisor: Prof. Tiago Vaz Maia
Member of the Committee: Prof. João Miguel Raposo Sanches

July 2014

À mulher da minha vida, a minha mãe

Sempre chegamos ao sítio aonde nos esperam.

José Saramago *in A viagem do elefante*

Acknowledgments

Although only my name appears on the cover of this thesis, many more people deserve to be considered for their great support during this hard but enriching period. I would like to thank all people who contributed in some way to the work described in this thesis.

First and foremost, I would like to thank my supervisor, Prof. Tiago Vaz Maia, who has supported me throughout my thesis with his patience and knowledge. Our insightful conversations and his helpful suggestions contributed immensely to the development of this work.

This work would not be possible without the help and support of my fellow labmates. Rita, Inês and I have been working together for more than a year. I am lucky to have met you, and I thank you for your friendship and unyielding support. Thank you for the moments we spent together and for accepting me as I am. Thank you for providing me something much greater during the past year: a friendly smile and a hello every single morning. I thank Lena for her great support and the time she spent in reviewing this thesis. Vielen Dank. Maira and Libby, thanks for your generosity, support and friendship. Ana, Catarina, João and Vasco thanks for always being willing to help and for your useful insights and suggestions.

During the five years I have studied in IST I have made friends who definitely contributed for my success as a student but most important as a person. I will never forget the moments we spent together. Marcia, thanks for giving me the privilege of working with you during the five years I spent in IST. We were undoubtedly an excellent team. Thank you for your great help, for caring about me, for making me laugh and most importantly for being my friend. I would also like to thank my colleagues Mariana, Martina and Diogo who have made this journey unforgettable.

A special thank you goes to my best friend, Ricardo. You gave me your unconditional support through all this process. I am truly grateful for your advices, comprehension and above all for always being there for me.

I would like to thank my amazing family for always supporting me. Without them this work would have been almost impossible to carry out. A special thank you goes to my father for his unconditional love and support.

Finally, I would like to thank the most important person whose contribution was vital not only for the completion of this thesis but also for my academic success, my mother. My mother has been an inspiration throughout my life. She has always supported my dreams and aspirations. Whatever I am today, I owe it to her. Se has lived for me - each day and every day. Thanks mom for your love and support in all those moments I doubted myself and you did not. Thank you so much. I love you.

Abstract

Reinforcement learning has provided a normative framework to analyse decision-making. A wealth of research has linked reinforcement learning to neural substrates, assigning them a particular computational role. Particularly, responses of dopamine neurons can be identified with the prediction errors computed in the temporal-difference learning algorithms. Machine learning literature has proposed different versions of calculating the error signal, associated with different temporal-difference algorithms. Particularly, they can be determined by the value of actions (Q-learning model) or by the value of states (Actor-Critic model). Neuroscientific findings have supported both models, and thus, there is still no commonly accepted mechanism.

The aim of this thesis was to investigate and identify which of these two reinforcement learning models best describes the choices made by healthy humans when performing a modified probabilistic Go/NoGo task. This paradigm has the special feature of orthogonalizing action and valence and thus it enhances some mechanistic differences between the Q-learning and the Actor-critic models.

For this purpose, we employed several statistical methods. Firstly, using a model fitting approach we tried to identify which of the aforementioned models best suited data. Secondly, we performed a Principal Component Analysis in order to find associations among conditions which could also provide evidence towards one of the models.

Both approaches provided evidence towards the Q-learning framework which indicated that the prediction errors are determined by the value of actions. This result was in line with electrophysiological findings in animals.

Keywords

Actor-Critic, Q-learning, Prediction errors, Go/NoGo task, Dopamine

Resumo

A aprendizagem por reforços forneceu uma estrutura normativa para a análise de tomada de decisões. Vários estudos mostraram que existe uma ligação entre a aprendizagem por reforços e algumas estruturas neuronais, sendo que estas estão associadas a uma determinada ação computacional. Nomeadamente, as respostas de neurónios dopaminérgicos estão relacionadas com os erros de previsão utilizados nos algoritmos de diferenças temporais. A literatura em Machine Learning apresentou diferentes maneiras de calcular o erro de previsão, as quais estão associadas a diferentes algoritmos de diferenças temporais. Estes podem ser determinados pelo valor da ação (modelo de Q-learning) ou pelo valor do estado (modelo de Actor-Critic). Evidências neurocientíficas têm apoiado ambos os modelos, e por isso não existe ainda um mecanismo globalmente aceite.

O objetivo desta tese é investigar e identificar qual dos modelos supracitados melhor descreve as escolhas realizadas por humanos saudáveis enquanto estão a executar uma tarefa probabilística Go/NoGo. Este paradigma é capaz de ortogonalizar ação e valência, e por isso, realça algumas diferenças mecanísticas entre os modelos de Q-learning e Actor-Critic.

Para tal, nós utilizámos várias abordagens estatísticas. Em primeiro lugar, recorrendo a uma análise de regressão, tentámos identificar qual dos modelos descrevia melhor os dados do comportamento. Posteriormente, realizámos uma análise de componentes principais a fim de encontrar correlações entre as condições do paradigma, o que poderia fornecer mais uma evidência a favor de um dos modelos. Ambas as abordagens sugeriram que o modelo de Q-learning seria o mais correto, ou seja, os erros de previsão são determinados pelo valor das ação, ao invés, do valor do estado. Este resultado está de acordo com estudos electrofisiológicos feitos em animais.

Palavras Chave

Actor-Critic, Q-learning, Erros de previsão, Tarefa Go/NoGo, Dopamina

Contents

List of Figures

List of Tables

Abbreviations

AC actor-critic

BIC Bayesian information criterion

BG basal ganglia

BMS Bayesian model comparison

CR conditioned response

CS conditioned stimulus

DA dopamine

GPe globbus pallidus pars externa

GPi globus pallidus internal segment

LLH log-Likelihood

LTD long-term depression

LTP long-term potentiation

MDP markov decision process

MLE maximum likelihood estimation

PCA principal component analysis

PE prediction error

QL Q-learning

RL Reinforcement learning

SNc substantia nigra pars compacta

SNr substantia nigra pars reticulata

STN subthalamic nucleus

TD temporal difference

UR unconditioned response

US unconditioned stimulus

List of Symbols

t	discrete time step	9
s_t	state at t	9
S	Set of states	9
a_t	action at t	9
$A(s_t)$	Set of actions in s_t	9
r_{t+1}	reinforcement at t, dependent on a_t and s_t	9
R_t	return (cumulative discounted reinforcement) following t	9
γ	discount factor	9
$R(s, a, s')$	expected immediate reward on transition from s to s' under action a	9
$T(s, a, s')$	probability of transition from state s to state s' under action a	9
$V^\pi(s_{t+1})$	value of state s_{t+1} under policy π	10
π	policy, decision making rule	10
$Q^\pi(s, a)$	value of taking action a in state s under policy π	10
$\pi(s, a)$	probability of taking action a in state s under policy π	10
$\hat{Q}^\pi(s, a)$	estimate of $Q^\pi(s, a)$	11
α	learning rate	11
β	inverse of temperature	12
$\hat{V}^\pi(s)$	estimate of $V^\pi(s)$	12
M	model	27
D_s	set of choices made by subjects s	27
θ_s	set of parameters of subject s adopting a model M	27
μ_k	mean vector of a Gaussian distribution	32
Σ_k	covariance matrix of a Gaussian distribution	32
π_k	mixing coefficients of a Gaussian mixture model	32
$\gamma_k = p(k\|x)$	responsibility of a Gaussian mixture model	32

1

Introduction

Contents

1.1 State of the art

Learning is a process that allows animals to adapt to the environment which surrounds them. Food sources and predators are constantly changing of location and appearance and areas which used to be save can become extremely hazardous. Therefore, learning to make the most favourable decision is essential for creature's survival.

A fundamental question in behavioral neuroscience concerns the decision-making process used by animals and humans for selecting actions in the face of reward and punishment. This process has been extensively investigated through the paradigms of classical and instrumental conditioning.

In classical conditioning, subjects learn the association between events [55]. In contrast, instrumental learning involves learning to select actions[46, 50], and thus, subjects make connections between actions and events. According to the Thorndike's law, in the presence of a reward the connection is strengthened and in the presence of a punishment the connection is weakened [50].

This trial and error procedure is also found in temporal-difference reinforcement learning algorithms which are commonly employed in artificial systems, such as robots, in order to make them capable of learning to select actions [26, 48]. Therefore, they have gained popularity in behavioral neuroscience to explain conditioning behavior. These models learn how to make a decision by predicting the value of taking an action from a recognized situation. The subject can thus choose the action which maximizes that value. The value is then updated through the prediction error defined as the difference between the expected and the observed value.

In the last decade, several observations showed a good parallel between neurobiological processes in the brain and the computational steps of Reinforcement learning algorithms. The most notable finding was the relationship between dopamine and prediction errors. Namely, it was proved that phasic responses of the midbrain dopamine code reinforcement prediction errors [29, 30, 43]. Another important evidence came from the studies done by Wickens and colleagues who found that the plasticity of corticostriatal synapses is weighted by dopamine input from midbrain dopamine neurons [36–38, 53]. Additionally, findings suggest that striatum plays a major role in linking the value to action selection [20, 23, 24].

These observations are all unified in the basal ganglia Go/NoGo neurocomputational model which explains how temporal difference methods are implemented in the basal ganglia[1] and how they can generate action.

Machine learning literature has proposed different versions of calculating the prediction error, associated with different temporal difference algorithms (e.g. [48]). In the actor-critic algorithms [3], the prediction error ignores actions altogether and thus it is determined by the value of the situation. In the other two classes of algorithms, Q-learning [51] and SARSA (state-action-reward-state-action) [35], prediction errors are determined by the action value which is also called Q-value. In the Q-learning approach, the prediction errors associated to a decision is determined by the Q-value of the better option rather than the one actually chosen. On the other hand, in SARSA algorithms the prediction errors use the Q-value of the chosen option.

Neuroanatomical findings support the actor-critic model [3], whereas electrophysiological evidences

[1] Basal ganglia consists of a group of interconnected subcortical nuclei, such as the striatum.

are in line with the other two classes of algorithms. Recent evidence from primate study seems to support SARSA [31]. On the other hand, evidence from a rodent study favours Q-learning [40].

Resolving this discrepancy will thus necessitate further experiments and computational investigation. All previous studies were performed in animals, and, although the characteristics of human conditioning are similar to those of animal conditioning, as far as we know, no study has focused on investigating the type of temporal difference algorithm implemented in humans.

1.2 Objective

This thesis aims to investigate whether prediction errors, in humans, are determined by the value of the situations (actor-critic) or by the value of the actions. The prediction errors determined by the value of the actions can further reflect the Q-value of the better option (Q-learning) or the Q-value of the chosen option (SARSA). This work is not concerned about the latter distinction and thus will consider the class formed by the Q-learning and SARSA algorithms instead of considering them separately. Since both approaches involve Q-values, this class will henceforth be denominated as Q-learning.

In brief, this work will try to determine whether basal ganglia implements an actor-critic or an Q-learning approach in humans.

1.3 Thesis Outline

This thesis starts by clarifying and detailing the link between conditioning and Reinforcement learning in chapter 2. In the same chapter, the biological foundations of Reinforcement learning in the brain and the mathematical framework of the temporal difference algorithms used in this work are presented.

The methods and models are fully described in chapter 3. It describes the details of the modified Go/NoGo learning paradigm used in this study. Furthermore, a description is provided of a flexible algorithm which allows the design of different type of paradigms within the Go/NoGo family, including the task employed in this study. The models adopted to explain the behavioral data are also fully defined in this chapter. The statistical approach used to fit temporal difference algorithms to the behavioral data, as well as a description of the current statistical methods of model comparison are also provided. Finally, other statistical procedures also employed in this work, namely principal component analysis and Gaussian mixture models, are fully covered.

After that, the statistical analysis of the behavioral data is conducted in chapter 4 in order to verify the performance of the subjects in all task conditions. Two different approaches were employed. The first approach did not consider the tendency to respond whereas the second approach took into consideration the Go bias when determining whether subjects have learned each condition.

Finally, in chapter 5 the results from the model fitting to the behavioral data are described. Several versions of the Q-learning and actor-critic algorithms are compared in order to determine the approach that best describes the raw data. Additionally, principal component analysis and Gaussian mixture models were also applied to the behavioral raw data in order to extract some relationships among conditions. Since these associations depend on whether the subjects follow a Q-learning or actor-critic model, they

can provide evidence towards one of the models.

2

Background

Contents

This chapter starts by clarifying and detailing the link between conditioning and Reinforcement learning. The biological foundations of Reinforcement learning in the brain and the mathematical framework of the temporal difference algorithms used in this work are also presented.

2.1 Instrumental conditioning and Reinforcement Learning

There are two types of conditioning: classical and instrumental. In classical conditioning a conditioned stimulus (CS), that does not elicit a particular response, is associated to an unconditioned stimulus (US), that does elicit a specific response (unconditioned response (UR)). After pairing them repeatedly, the CS starts evoking a conditioned response (CR) [11]. There were two interpretations about how the CS produces responding. One idea establishes that the CS directly elicits a response, giving rise to a new stimulus-response (S-R). A second idea does not connect directly the CS and the CR, but instead claims that the CS activates a representation or memory of the US which produces a stimulus-stimulus (S-S). A variety of classical conditioning situations have shown evidence for S-S learning rather than S-R learning [11]. The mechanism underlying this learning process is thought to be based on unexpected events. The idea is that the subject just learns when the US is different from what is expected, which activates processes leading to new learning. This theory is mathematically translated by the fundamental equation of the Rescorla-Wagner model 2.1.

$$\triangle V = k(\lambda - V) \tag{2.1}$$

λ represents the US value and V the associative value of the CS. Therefore, $(\lambda - V)$ represents the discrepancy between the expected outcome or CS and the unexpected outcome US. According to the model, at the beginning there is a high level of surprise, because one is expecting nothing until an outcome is delivered. Therefore the discrepancy between CS and US is very high. At the asymptote of learning, both stimuli will share the same value which means that $(\lambda - V)$ will tend towards zero [11, 34]. k is a constant related to the salience of the CS and US.

In classical conditioning, the subject does not have to perform any particular response to obtain an outcome or US. For instance, learning to predict that when clouds turn gray it is likely to rain does not require any kind of response. On the other hand, there are situations in which responding is necessary to produce a desired outcome [11], such as learning how to open a door. This process is called instrumental conditioning and it is formally theorized in Thorndike's law of effect [11, 26]. The law of effect states that if a response in the presence of a stimulus is followed by a positive outcome, the S-R connection is strengthened. On the other hand, if it is followed by a negative outcome, the S-R connection is weakened. It is important to stress that the outcome is not one of the elements of this association. It just strengthens or weakens the S-R connection, thus the law of effect only explains S-R learning, often called habits [11, 26]. A habit is a devaluation-insensitive behaviour, i.e. even when the outcome value changes, subjects persist in performing actions whose outcome is no longer of interest [9, 26]. At the other end of the instrumental behaviour spectrum, goal-directed behaviour is devaluation-sensitive. In this case, the

subject's behaviour rapidly shifts according to the new outcome, because responses are guided by the response-outcome (R-O) and the stimulus-response-outcome (S-R-O) associations [26].

Reinforcement learning (RL) [48] is a branch of artificial intelligence which is concerned about how an agent learns to make optimal decisions, based on predictions of long-run future consequences, aiming to maximize rewards and minimize punishments. Therefore, this computational field has provided a normative framework to understand conditioned behaviour in humans and animals. RL methods, namely the temporal difference (TD) models, similarly to the Rescorla-Wagner equation 2.1, have in common the use of the difference between the expected outcome and the unexpected outcome to update their predictions about a certain decision, or in a instrumental conditioning framework, to update the S-R association. This discrepancy is also know as prediction error (PE).

Reinforcement learning allows us to analyse and interpret mechanistically animal and human conditioning, namely it suggests a means by which optimal prediction and action selection can be achieved and exposes explicitly the computations that must be realized in order to achieve that [33].

Neuroscientific evidence from lesion studies to pharmacological manipulations and electrophysiological recording in animals have proved a link between neural structures and the variables used by TD learning methods [8, 43, 49, 52]. This connection has also been shown in humans in studies using functional magnetic resonance imaging [34, 42]. Particularly, it has been observed that phasic dopamine (DA) conveys PEs [43] which suggests that the brain may implement this kind of strategy.

2.2 Reinforcement learning in the brain

This section deals with the biological foundations of reinforcement learning in the brain. It starts by describing anatomically the brain structures most important to reinforcement learning, i.e. basal ganglia (BG). Then it explains the effects of DA as a neuromudolator in the BG and how it conveys PEs.

2.2.1 Basal ganglia

The BG consist of a group of interconnected subcortical nuclei located in the telencephalon, diencephalon and midbrain. It includes the striatum, the pallidum, the substantia nigra and the subthalamic nucleus [1]. These structures may be divided into primary input strucutures, primary output structures and intrinsic nuclei. The primary input structures receive excitatory input from the cerebral cortex and they are composed by the striatum and subthalamic nucleus (STN). The former receives input from virtually all areas of cerebral cortex and the latter from motor areas of the frontal lobe [28]. The output structures are the globus pallidus internal segment (GPi) and the substantia nigra pars reticulata (SNr), which inhibit thalamocortical projections to the prefrontal, premotor and motor cortical areas responsible for the generation of purposive movements [17, 28]. The intrinsic nuclei are the globbus pallidus pars externa (GPe) and the substantia nigra pars compacta (SNc).

In the BG, there are two main output pathways of the striatum: the direct (or Go pathway) and indirect pathway (or NoGo pathway) [17, 27]. In the direct pathway, the striatal medium spiny neurons (Go neurons) inhibit directly the output structures which in turn disinhibit the thalamus, thereby facilitating movement. On the other hand, spiny neurons in the indirect pathway (NoGo neurons) provide excita-

tory input to the output structures through the inhibition of the GPe. The NoGo neurons inhibits the GPe, leading to the deshinibition of the output structures which in turn suppresses the thalamocortical projections, and thus restrains movement.

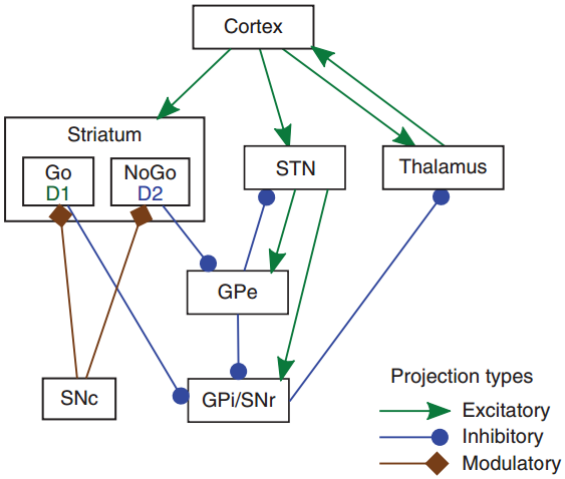

Figure 2.1: Diagram of the direct (Go) and indirect (NoGo) pathways. Adapted from Maia&Frank, 2011 [27].

In the hyperdirect pathway, neurons in the STN receive direct input from the cortex and exert diffuse powerful excitatory effects on the output structures of the basal ganglia. It has suggested that this third pathway is responsible for providing tonic inhibition to the thalamocortical system to suppress movement in the resting condition [17]. It has also been shown that tonic inhibition prevents premature suboptimal responding. [27, 32].

2.2.2 Basal ganglia and dopamine

DA acts within the striatum predominantly through D1 and D2 receptors [17]. Striatal medium spiny neurons are heterogeneous in their expression of DA receptors: neurons from the direct pathway expresses predominantly D1 receptors whereas neurons from the indirect pathway expresses D2 receptors [17, 45]. D1 receptor signalling promotes long-term potentiation (LTP), which strengthens the direct pathway, and D2 signalling promotes long-term depression (LTD), which weakens the indirect pathway [27, 45]. Tonic DA levels in the striatum are sufficient to keep high-affinity D2 receptors active, but not low-affinity D1 receptors. Therefore, it is hypothesized that D1 stimulation depends on phasic DA bursts [45]. The magnitude of the bursts are thus crucial for D1-mediated LTP, with higher burst producing greater stimulation [27]. On the other hand, D2-mediated LTP receptors depend on DA dips. It has been shown that phasic DA bursts in striatum quantitatively represent positive PEs [43]. However, there is less evidence that negative PEs are encoded by phasic DA reductions. This might be due to the low tonic firing rate which limits such reductions. It seems, however, that the duration of DA dips represent quantitatively the negative PEs [27]. As explained in section 2.1, PEs are used to learn the values of state-stimulus (classical conditioning) and state-response pairs (instrumental conditioning). These values can then be used to select an action. The mechanism used by the BG to select an action finds formal expression in the basal ganglia Go/NoGo model [15]. According to this model, the direct and indirect pathway can learn which actions to facilitate and suppress in each state, respectively, using the PEs

encoded by phasic DA. When an action is followed by a positive PE, there is a phasic DA burst which will strengthen the corticostriatal synapses of the direct pathway via D1-mediated LTP and weaken the corticostriatal synapses of the indirect pathway via D2-mediated LTD. When there is a negative PE, the reverse occurs [15, 27].

In addition to the role of phasic DA, tonic DA increases the excitability of the indirect pathway and decreases excitability in the indirect pathway, because D1 receptors are excitatory and D2 receptors are inhibitory. This way, high levels of tonic DA increases the tendency to respond (Go bias) whereas low levels reduces the tendency to respond (NoGo bias) [27, 42]. Therefore, tonic DA modulates the prior learning whereas phasic DA is responsible for the actual learning process [27].

2.3 Temporal Difference learning

So far, we have gone through the psychological and biological reasons for using reinforcement learning models to mechanistically describe the procedure that humans use while learning to make decisions. It is thus plausible to assess which candidate models best predict the data that is actually observed. In order to better comprehend the TD learning models used to explain behavioural data, it is worth to have an insight over the basics of reinforcement learning.

Firstly, a brief introduction to the mathematical framework of RL is presented. Afterwards, we will focus on the three most popular TD algorithms: actor-critic (AC), Q-learning (QL) and SARSA.

2.3.1 Reinforcement learning framework

Similarly to the instrumental conditioning, in RL an agent interacts with the environment in order to maximize rewards and minimize punishments. More specifically, the agent and the environment interact at discrete time steps $t \in \mathbb{N}$. At each time, the environment presents to the agent a state $s_t \in S$, where S is the possible set of states, and the agent must select an action $a_t \in A(s_t)$, where $A(s_t)$ is the set of allowable actions in state s_t. This action makes the agent to transit to another state s_{t+1} and to receive a reinforcement r_{t+1}. At each time step, the agent implements a policy which is a mapping from states $s \in S$ and actions $a \in A(s)$ to the probability of taking action a when in state s, seeking to maximize a discounted sum of future reinforcements (rewards and punishments), also known as return (R_t).

$$R_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k+1} \tag{2.2}$$

γ is the discount factor, $0 \leqslant \gamma \leqslant 1$. This parameter translates the effect of time on reinforcements: a reinforcement received later in the future is worth less than the same reinforcement received immediately.

The environment of a reinforcement learning problem can usually be defined as a markov decision process (MDP). An MDP is defined by a set of states and actions, as previously described, and two functions, $R(s, a, s')$ and $T(s, a, s')$. The former is the reinforcement that an agent receives when it is in state s, performs action a and transits to state s'. The reinforcement might be stochastic, in those cases, $R(s, a, s')$ is the expected reinforcement. $T(s, a, s')$ determines the transition probabilities, i.e. the

probability of transitioning from state s to state s' when action a is performed. An MDP obeys to the Markov property which states that the future of the system is independent of the past events, depending just on the present. This means that the transition probabilities depend solely on the action a_t selected in the current state s_t, and do not depend on the previous set of actions and states. Note that the agent does not follow the Markov property, but only the environment does. In fact, the agent learns the consequences of certain actions in certain states while interacting with the environment, meaning that the agent's behaviour depends on its history.

This way, an action a_t taken in state s_t will dictate the probability of transitioning to a next state s_{t+1}, which will thus have an impact on the future return. Therefore, when an agent decides which action to choose in a given state, it has to take into account not only the immediate outcome, given by $R(s_t, a_t, s_{t+1})$, but also the value of the next state under a specific policy π, $V^\pi(s_{t+1})$. The value of a state s is defined as the expected value of the discounted sum of futures reinforcements that the agent will receive when it starts from that state, under policy π (equation 2.3). It is also denominated as state-value function.

$$V^\pi(s) = E_\pi\{R_t|s_t = s\} = E_\pi\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1}|s_t = s\} \tag{2.3}$$

$E_\pi\{\}$ denotes the expected value given that the agent follows policy π.

Similarly, we can define the value of taking an action a in state s under policy π as,

$$Q^\pi(s, a) = E_\pi\{R_t|s_t = s, a_t = a\} = E_\pi\{\sum_{k=0}^{\infty} \gamma^k r_{t+k+1}|s_t = s, a_t = a\} \tag{2.4}$$

where $Q^\pi(s, a)$ is the action-value function, often shortened to Q-value.

Knowing the functions $T(s, a, s')$ and $R(s, a, s')$, one can compute the state-value function of a given state s according to the Bellman equation (equation 2.5).

$$V^\pi(s) = \sum_{a \in A(s)} \pi(s, a) \sum_{s'} T(s, a, s')[R(s, a, s') + \gamma V^\pi(s')] \tag{2.5}$$

The Bellman equation simply states that the value of a state s is a weighted average of the total expected reinforcement when the agent is in state s, performs action a, and transitions to state s', over the possible set of actions and state transitions. The weights correspond to $T(s, a, s')$ and $\pi(s, a).\pi(s, a)$ consists in the probability of taking action a in state s under policy π.

$Q^\pi(s, a)$ depends on both state and action, and thus it is computed after an action is performed, which means it is independent of $\pi(s, a)$. Thereby, it can be expressed as:

$$Q^\pi(s, a) = \sum_{s'} T(s, a, s')[R(s, a, s') + \gamma V^\pi(s')] \tag{2.6}$$

If the MDP is known, i.e. $T(s,a,s')$ and $R(s,a,s')$, finding the state-value function $V^{\pi}(s)$ and the Q-value $Q^{\pi}(s,a)$ is straightforward. Often, however, the MDP is unknown. For instance, in decision-making tasks, besides some rules that are given in the instructions, the participants do not know the task (environment) structure. Therefore, the state-value function and the Q-value have to be estimated from experience. This is the main idea underpinning TD methods.

As in the Rescorla-Wagner model, the TD algorithms update recursively the state-value or the action-value functions using a PE, which signals the difference between the expected and the actual reinforcement. RL literature has proposed different versions of calculating PE, associated with different temporal difference algorithms: QL, SARSA and AC. In the QL [51] and SARSA [35] algorithms, the prediction errors are determined by the Q-value, whereas in the AC [3], they are determined by the state-value function.

2.3.2 Q-learning and SARSA

According to the QL framework, the estimate of the Q-value is updated using the learning rule expressed in equation 2.7.

$$\hat{Q}^{\pi}(s,a) \leftarrow \hat{Q}^{\pi}(s,a) + \alpha[r + \gamma \max_{a} \hat{Q}^{\pi}(s',a) - \hat{Q}^{\pi}(s,a)] \tag{2.7}$$

$\hat{Q}^{\pi}(s,a)$ is the estimate of $Q^{\pi}(s,a)$, $r + \gamma \max_a \hat{Q}^{\pi}(s',a) - \hat{Q}^{\pi}(s,a)$ is the prediction error and α ($0 \leq \alpha \leq 1$) is the learning rate. The learning rate dictates how fast the agent learns the Q-value under policy π. Rearranging the right side of the equation, $\alpha[r+\gamma \max_a \hat{Q}^{\pi}(s',a)] + (1-\alpha)\hat{Q}^{\pi}(s,a)$, we can find that an agent with a higher learning rate gives more importance to the immediate outcome rather than the previous outcomes, which is the reason why the agent learns faster. The max operator means that the prediction error is computed with respect to what is believed to be the best action at the subsequent state s'. This method is considered "off-police", as it takes into account the best future action, even if this will not be the action that is actually taken in s'. In an alternative "on-police" variant called SARSA, the prediction error take into account the next chosen action, rather than the best possible action, resulting in the learning rule described in equation 2.8.

$$\hat{Q}^{\pi}(s,a) \leftarrow \hat{Q}^{\pi}(s,a) + \alpha[r + \gamma \hat{Q}^{\pi}(s',a) - \hat{Q}^{\pi}(s,a)] \tag{2.8}$$

Given the Q-value estimates for all action candidates in a given state, choice might be as simple as "greedily" selecting the action expected to deliver the greatest future value. However, this policy might miss out on more valuable actions that have not yet been explored or have previously been unlucky. For this reason, RL models generally assume that there should be some degree of randomness in the choices. In RL, this is often accomplished by the Gibbs softmax method (equation 2.9). This method assigns a probability to each action according to the corresponding Q-value: the actions with the highest Q-values are associated with the highest probabilities.

$$P(a|s) = \pi(s,a) = \frac{e^{\beta Q(s,a)}}{\sum_{b \in A(s)} e^{\beta Q(s,b)}} \tag{2.9}$$

Parameter $\beta \geq 0$, known as inverse of temperature, allows the agent to deal with the trade-off between exploration and exploitation. For high values of β, the agent tends to exploit the action with the highest Q-value (greedy agent), whereas for $\beta = 0$ the agent choices are totally random (all actions have the same probability of being chosen), i.e. the agent explores all the actions.

The states of the task used in this work are independent and thus the transition probabilities do not depend on the action performed. Therefore, the agent does not need to take into account the future reinforcements when updating the Q-value, i.e. $\gamma = 0$. This way, both learning rules become equal (equation 2.10).

$$\hat{Q}^{\pi}(s,a) \leftarrow \hat{Q}^{\pi}(s,a) + \alpha[r - \hat{Q}^{\pi}(s,a)] \tag{2.10}$$

This task is thus not able to distinguish between QL and SARSA. However, as previously explained, we are not interested in this distinction, but instead in comparing AC with the class of models formed by the QL and SARSA. In other words, we aim to investigate whether prediction errors, in humans, are determined by the value of states (actor-critic) or by the value of actions.

2.3.3 Actor-Critic

The actor-critic model has separate memory structure in order to represent explicitly the policy independent of the value function. Unlike Q-learning and SARSA, the actor-critic model does not make use of the action-state function to compute PEs. Instead, they are determined by the state-value function. The structure which updates the state-value function and computes prediction errors is called critic (2.11).

$$\hat{V}^{\pi}(s) \leftarrow \hat{V}^{\pi}(s) + \alpha[r + \gamma\hat{V}^{\pi}(s') - \hat{V}^{\pi}(s)] \tag{2.11}$$

$\hat{V}^{\pi}(s)$ is the estimate of $V^{\pi}(s)$ and $r + \gamma\hat{V}^{\pi}(s', a) - \hat{V}^{\pi}(s, a)$ is the prediction error.

The prediction error is then provided by the critic to the actor to update the preferences (equation 2.12).

$$p(s,a) \leftarrow p(s,a) + \eta[r + \gamma\hat{V}^{\pi}(s') - \hat{V}^{\pi}(s)] \tag{2.12}$$

The parameter η is the actor's learning rate.

In sum, the actor selects actions and the critic, as it suggests its name, criticizes the policy currently followed by the actor using the prediction error. The actions are also selected according to the softmax decision rule, providing the preferences instead the Q-values.

2.3.4 Pavlovian effects on reinforcement learning

In decision-making, one must perform choices which maximize reward and minimize punishment. In animals, this optimization is governed by two mechanisms: pavlovian policies which links outcomes to valence-dependent stereotyped behavioral responses and a more instrumental policy which learns solely based on contingent consequences [19]. Regardless of the action validity, Pavlovian responses associated with reward entails vigor whilst responses linked to punishments are associated with action inhibition

[18, 19]. Usually, cognitive tasks explore this interdependence between action and valence, i.e. the coupling between reward and go choices and punishment and nogo choices. In the previous case, the pavlovian effect does not reveal itself so prominently, because it does not disrupt the interaction between action and valence. Some studies started to use tasks which allow to orthogonalize action and valence [19] similar to the one presented here. In other words, in this task both action and action inhibition are associated with reward and punishment which is translated into 4 different conditions: go to win, go to avoid losing, nogo to win, nogo to avoid losing. The pavlovian effect is more prominent in conditions where the action-valence interaction is disrupted, i.e., the condition where the subject must choose to go in order to avoid a punishment (go to avoid losing) and the condition where the correct choice is to inhibit the go action to receive a reward (nogo to win).

It has already been shown that a Q-learning model which accounts for the Pavlovian component effect fits better to behavioural data of subjects who performed tasks where action and valence were orthogonalized in the manner previously explained [19]. Since the task presented here also makes action and valence independent, one expects that behavioural data show some action by valence interaction which must be taken into account by our models. Thereby, we have included the Pavlovian parameter described in [19].

2.4 Motivation

Recently, neuroscience have begun to integrate temporal difference models directly into the design and analysis of experiments, quantitatively studying the models' fit to behavioral responses from individual subjects and trials. Therefore, the identification of the correct model may be helpful to understand the decision-making process used by animals and humans for selecting actions in the face of reward and punishment.

Understanding how such process can fail in certain conditions may shed light on the complex pathology of psychiatric disorders [12]. Particularly, disturbances of the dopaminergic system and basal ganglia circuits have a key role in several psychiatric and neurological disorders. Reinforcement learning models have recently started to be applied to these disorders and have been shown to have substantial explanatory and predictive power [27]. The approach builds on an understanding of the computations that these circuits perform in healthy individuals and investigates how pathophysiological processes alter these computations, producing symptoms. Thus, determining the correct class of algorithms employed by healthy humans is now, more than ever, of utmost importance.

However, there are different versions of temporal difference models associated to different ways of calculating PEs. In the actor-critic, the PEs are calculated solely based on the value of the states and thus with no knowledge of the actions ($\delta_t = r_{t+1} - V(s_t)$). On the other hand, in the QL framework, they are determined by the value of the actions ($\delta_t = r_{t+1} - Q(s_t, a_t)$).

Electrophysiological evidences in animals are in line with the QL framework [31, 40]. However, neuroanatomical findings suggest that the brain implements the actor-critic model [3]. As described in section 2.3, the AC model is divided into two different structures: the actor and the critic. The former is responsible for select actions and the latter estimates the value of states, which is then used to calculate

the PEs. It has been shown that such functions are also anatomically separated in the brain. Namely, the dorsolateral striatum is associated with the actor and the ventral striatum is related to the critic [3, 26].

Although animals can display complex decision-making behavior, we are interested in comprehend human-decision making and its relationship with the Reinforcement learning framework. Furthermore, the possibility of instructing subjects verbally allows for much more complex paradigms in human experiments. This way, we applied a task which highlights some differences between the actor-critic and QL framework.

Recent studies have used tasks that fully orthogonalize action and valence in a 2 (reward/punishment) $\times$ 2 (Go/NoGo) design [7, 19]. Typically, in this kind of task, there are four different conditions: respond to gain a reward (go to win); respond to avoid punishment (go to avoid losing); do not respond to gain a reward (nogo to win); do not respond to avoid punishment (nogo to avoid losing). Assuming a Q-learning approach, there are two possible actions: go and nogo. For each condition and action, it is calculated the corresponding Q-value using equation 2.10. In conditions whose outcome is positive, the PE is also positive and negative for conditions whose outcome is negative. Therefore, the nogo to win and go to avoid losing conditions exhibit positive and negative PEs, respectively. For this reason, nogo to win and go to avoid losing are linked to DA bursts and dips, respectively. The basal ganglia neuroanatomical model dictates thus that the nogo to win condition would strengthen the Go pathway instead of the NoGo pathway. Conversely, the go to avoid losing condition would weaken the Go pathway and strengthen the NoGo pathway. This incongruence reveals a gap between the QL algorithm and the neurobiological explanation, which can be overcame by the AC model.

In the AC model only one action is considered (go action), but subjects are free to choose between performing or not the action. The state-value is updated by the critic regardless if the action is performed or do not performed. Conversely, the preferences are only updated when an action is performed. Therefore, in the nogo to win condition, when the subject do not perform an action it will obtain a positive outcome, and thus the state-value becomes positive. When the agent stops doing the action, the prediction error turns negative, and, consequently the preference too. This is in accordance with the basal ganglia Go/NoGo neurocomputational model, because the negative PEs will weaken the Go pathway and strengthened the NoGo pathway. The previous explanation concerns the nogo to win condition, but the same logic could be employed to the go to avoid losing condition.

Recent findings [19], which used a model whose structure is identical to the Q-learning, showed that subjects perform worse in the nogo to win and go to avoid losing conditions, comparing to the nogo to avoid losing and go to win conditions, respectively. They explained these findings by assuming the existence of a Pavlovian effect, i.e. the subjects tend to perform the go action when the sate-value is positive (nogo to win and go to win) and the opposite when the it is negative (go to avoid losing and nogo to avoid losing). However, we suggest that these observations might result from the fact that subjects execute the AC rather than the QL model. In the actor-critic, subjects must learn the state-value before making a decision. In the nogo to win and go to avoid losing conditions, in order to learn the state-value subjects must perform the action which does not update the preferences, and thus it slows the learning process. Put differently, in these two conditions it is not possible to learn the state-value while updating

the preferences.

This thesis is committed to determine which model, QL or AC, best describes the subjects behaviour in a modified probabilistic Go/NoGo task which orthogonalizes action and valence and contains the four conditions previously described: go to win; go to avoid losing: nogo to win; nogo to avoid losing.

3

Methods

Contents

This chapter describes the details of the modified Go/NoGo learning paradigm used in this study. Furthermore, a description of a flexible algorithm which permits the design of different type of paradigms within the Go/NoGo family, including the task employed in this study, is provided. The QL and AC models adopted to explain the behavioral data are also fully defined in this chapter. The statistical approach used to fit temporal difference algorithms to the behavioral data, as well as a description of the current statistical methods of model comparison are also provided. In model comparison, we covered first and second level analysis. In the former, we conclude that Bayesian information criterion (BIC) would be the best measure of the goodness of fit, and, in the latter a full description of both classical and Bayesian approaches were addressed along with its pros and cons.

Finally, other statistical procedures employed in this work, namely principal component analysis and Gaussian mixture models, are fully covered.

3.1 Experimental Task

Our task consists of a modified version of a probabilistic reinforcement Go/NoGo task in which at each trial a stimulus is presented and the subject has to press a key (Go) or withhold from pressing it (NoGo) according to the stimulus valence (reward/punishment). There are five different types of trials represented by five fractal images. Four of the trials differ from each other in the action-valence interaction: press the key to gain a reward (go to win); do not press the key to gain a reward (nogo to win); press the key to avoid a punishment (go to avoid losing); do not press the key to avoid a punishment (nogo to avoid losing). In the fifth type of trial the subject always receives a neutral outcome regardless of the action performed (neutral). Each of the five conditions has a fixed expected outcome (reward/punishment), but it is presented stochastically, i.e. in the go to win condition, every time the participant presses there is 70% of chance of getting a reward, 20% of chance of getting a punishment and 10% of chance of getting a neutral outcome; in the nogo to win condition, every time the participant withholds from pressing there is 70% of chance of getting a reward, 20% of chance of getting a punishment and 10% of chance of getting a neutral outcome; in the nogo to avoid losing condition every time the participant presses there is 70% of chance of getting a punishment, 20% of chance of getting a reward and 10% of chance of getting a neutral outcome; in the go to avoid losing condition every time the participant withholds from pressing there is 70% of chance of getting a punishment, 20% of chance of getting a reward and 10% of chance of getting a neutral outcome. If the opposite action is performed in each condition, the outcome is neutral with 100% of chance (figure 3.1). The outcome is probabilistic so that the learning process could be more difficult. However, the expected outcome remains constant, being positive for the win conditions (go to win and nogo to win) and negative for the avoid losing conditions (go to avoid losing and nogo to avoid losing).

The task is divided into 3 blocks, each with 10 trials of each type (go to win, go to avoid losing, nogo to win, nogo to avoid losing and neutral) in a total of 50 trials within a block. All trials share the same structure (figure 3.3). At the begin of every trial, a fixation point is displayed on the screen for 1000 ms (fixation event). After this, a fractal image appears on the screen for 1500 ms (stimulus event). During this period, the participants have to choose whether to press or not the key. After the offset of the image,

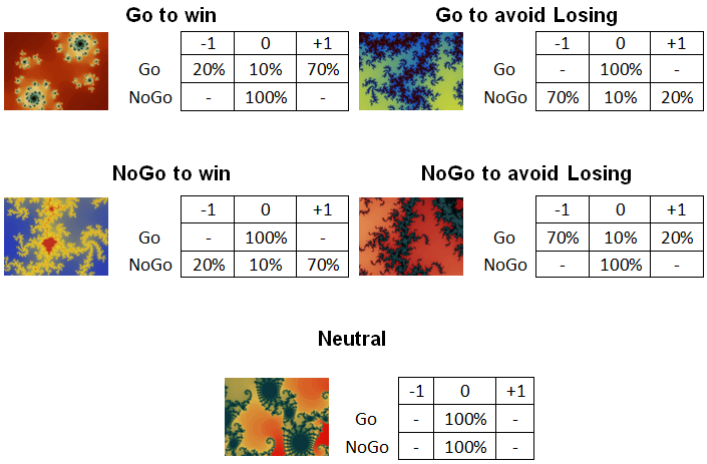

Figure 3.1: Probability distribution of the outcomes for the go to win, nogo to win, go to avoid losing and no-go to avoid losing conditions. The possible outcomes are $-1, 0, 1$, which corresponds to reward, neutral and punishment, respectively. The images correspond to the fractals shown throughout the task. Images and conditions were counterbalanced across subjects.

there is a variable interval (3000 ± 1000 ms) during which a progress indicator is displayed in order to give the participant a sense of time until he/she waits for the feedback (loading event). Furthermore, it prevents the participant to keep on pressing the key. Participants are presented with the feedback after this interval (feedback event). The feedback remains on the screen for 1000 ms: a green +1 indicates a win of 1 point (reward), a red -1 indicates a loss of 1 point (punishment) and a black 0 indicates no win or loss (neutral). The feedback is followed by a blank screen which takes 3000 ± 2000 ms (blank event). After this blank period, the next trial starts. The interval between the stimulus and the feedback and the interval between the feedback and the fixation point are both jittered, because this task is going to be used in an fMRI model-based analysis. Since the time onsets of the events of interest match both the stimulus and feedback onsets, it is extremely important to have jittered intervals between them in order to increase the variance of the signal[1].

Before the task begins, the subjects have to read some instructions which explains how the task works. Subjects were told that "images would appear on the screen throughout the task" and "each time the image appears [they] had to choose pressing the key or withhold from pressing it". It was also explained that "afterwards the result of [their] action would be presented on the screen". They were told "to notice that for the same image, the same action (press or not) could deliver different outcomes (to win points, to lose points, or nor win neither lose points)". "The aim of the game was to gather the maximum number of points" and "if [they] decided to press, [they] should do it as fast as they could".

Throughout the experiment, we have noticed subjects showed some difficulty to know whether the key pressing was valid or not, i.e. if their response was in time. Thereby, we decided to display the image with a transparency layer during 500 ms every time the key was pressed. This would signal that the response was valid. Since this alteration was performed later on, only 6 out of the 24 subjects presented here were affected. The original paradigm also included two more conditions: neutral plus delay go and neutral plus delay nogo. In the former condition, a neutral outcome was displayed whether the participant

[1] The fMRI is out of the scope of this thesis, thus it is not going to be covered here extensively.

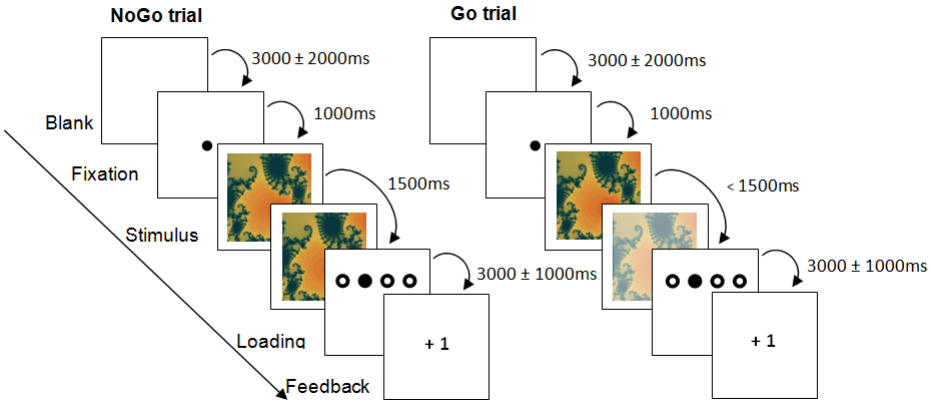

Figure 3.2: Schematics of a typical trial in the task composed of 5 events: fixation, stimulus, loading, feedback and blank. Temporal differences between trials in which the participant responds (go trial) and trials in which the participant does not respond (nogo trials) are also depicted. This scheme takes into account the image with transparency displayed after a key pressing (go trial). The suspension points-like structure corresponds to the progress indicator displayed on the screen while the participants were waiting for the feedback.

pressed or not the key, just like in the neutral condition. However, whenever the participant pressed, the duration of the interval between the stimulus offset and the feedback onset would be longer, i.e. there would be a delay. In the latter condition, a delay occurred whenever the participant did not respond. Since the delay was related to just one of the possible actions (go or nogo) and the outcome was always neutral, these conditions aimed to analyse the impact of the delay aversion in subject's decision making. We expected subjects to press more in the neutral plus delay nogo condition and less in the neutral plus delay go condition compared to the neutral condition. However, the time varying probability of making a go action of the neutral plus go delay showed an odd tendency in the last trials 3.3, starting to climb around the 18^{th} trial. At first glance, this could be explained by participants tiredness at the end of the task which makes them to start pressing in order to finish it earlier. Nevertheless, if this was the reason the same tendency should be presented in the neutral condition which was not the case. Since we could not find a reasonable explanation for this issue, we decided to discard both conditions with delay from this study and to came back to this issue later.

This paradigm is able to orthogonalize action and valence which allows us to study the effects of action and valence independently. However, it stands out from other similar previously used paradigms [19], because it contains a neutral condition capable of measuring the baseline behaviour, i.e. the natural tendency to perform an action without learning it. This is important not only to study vigour, which is out of the scope of this thesis, but also to determine whether participants have actually learned, by comparing the baseline responses (without learning) and the responses after learning.

Images and conditions were counterbalanced across subjects in order to avoid interactions between the image and condition and the order of appearance of the images was randomized within blocks. The task was run in a HP Pavilion dm4 Notebook PC with a 14-inch screen.

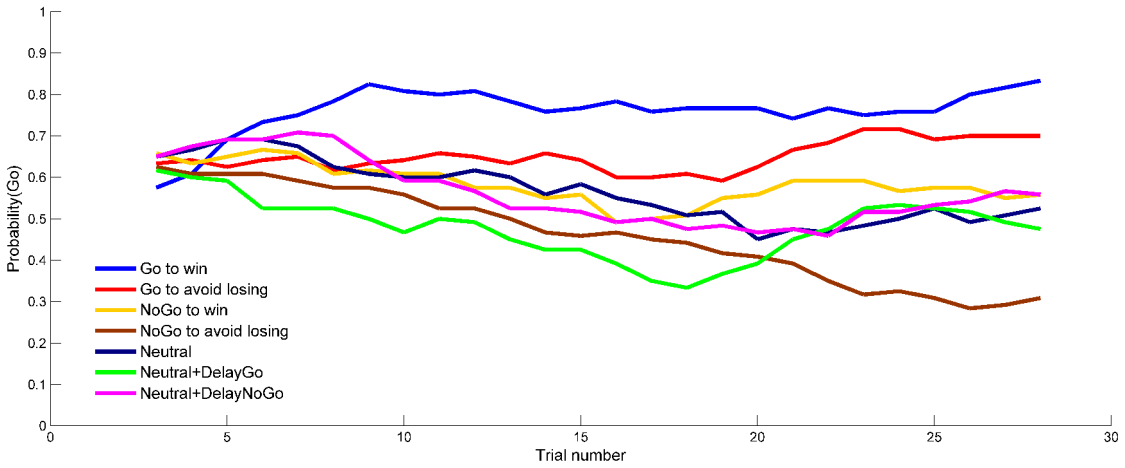

Figure 3.3: Time varying probabilities, across subjects, of making a go response of each condition convolved with a central moving average filter with length of 5.

3.2 Task programming

The paradigm described in section 3.1 was developed in MATLAB® version R2012b using Psychophysics Toolbox Version 3 (PTB-3). This toolbox consists on a free set of Matlab functions which allows us to achieve full control of the hardware for precise stimulus display, which could only be provided by low-level languages. PTB-3 allows us to work with interpreted languages (e.g. MATLAB) but provides the hardware control needed for precise stimulus display. The algorithm designed is very flexible, enabling us to design different tasks within the Go/NoGo family. The core of the algorithm is based on a Go/NoGo task with reinforcements. In each trial, the subject is presented with a stimulus (image) and have either to press a key or withhold from pressing the key. After that, a feedback is delivered. A feedback is delivered with a certain probability which depends on both the action and the image, (e.g. for image 1 the probability of getting a positive feedback when subject presses the button might be different from the one when he does not). Therefore, the user is free to match whatever condition to an image by setting the appropriate probabilities (e.g. the user can set whether an image is linked to positive or negative feedback when the subject presses or not the button). Each trial is composed by a **fixation** event followed by the **stimulus**. Whether the subject presses or not the button, after the stimulus there is a **loading** event until the feedback is provided. The feedback stays on the screen for a while and then the screen stays **blank** until the next trial starts. All these events are separated in time by a fixed or variable time intervals that the user is allowed to specify. The task is divided into a number of blocks and each block has a predefined number of times the same image can appear throughout the block.

In order to design a specific task, the user can set the aforementioned features, and many others, by filling out an excel file whose structure is already predefined. This file has 6 sheets: ConditionsGo, ConditionsNoGo, Groups, ImageSettings, TimeSettings and TaskSettings.

1. ConditionsGo sheet

   In this sheet, the user can set not only the outcomes the participant receives when he/she presses on a certain image but also its probability distribution. There is a table (Figure 3.4) where the

user is allowed to set all the reinforcement probabilities for each image/condition. Each column is a condition and the rows are the possible outcomes (e.g $+1$, -1, 0). Since the user is allowed to use as many images as he/she wants, it is possible to create new columns as long as the number of columns is the same as the number of images. For instance, if you want a simple Go/NoGo task you have just to set the probability of a positive feedback when pressing as 1 whilst the others as 0. Each condition is assigned to an image in a random fashion. In the output file one can check the correspondence between images and conditions.

Go to win		Go to avoid losing		NoGo to avoid losing		NoGo to win		Neutral	
Probability	Delay (seconds)	Probability	Delay (seconds)	Probability	Delay (seconds)	Probability	Delay (seconds)	Probability	Delay (seconds)
0,7	gamrnd(9,1/3)	0	gamrnd(9,1/3)	0,2	gamrnd(9,1/3)	0	gamrnd(9,1/3)	0	gamrnd(9,1/3)
0,2	gamrnd(9,1/3)	0	gamrnd(9,1/3)	0,7	gamrnd(9,1/3)	0	gamrnd(9,1/3)	0	gamrnd(9,1/3)
0,1	gamrnd(9,1/3)	1	gamrnd(9,1/3)	0,1	gamrnd(9,1/3)	1	gamrnd(9,1/3)	1	gamrnd(9,1/3)

Figure 3.4: Example of how a table should be set up for the present task.

There are some other columns that must be filled and whose aim is described below.

- **Feedback Text** - In this column the user have to match the outcomes (rows) and a certain image by tipping the name of the image file.
- **Upper left corner x** - In this column the user have to set the x coordinate of the upper left corner of the feedback image on the screen.
- **Upper left corner y** - In this column the user have to set the y coordinate of the upper left corner of the feedback image on the screen.
- **Down right corner x** - In this column the user have to set the x coordinate of the down right corner of the feedback image on the screen.
- **Down right corner y** - In this column the user have to set the y coordinate of the down right corner of the feedback image on the screen.
- **Delay** - Here the user can set the time between the stimulus and feedback by specifying its distribution through a MATLAB® function (e.g. normrnd(0,1) for a normal distribution with zero mean and unitary variance). This must be a valid MATLAB® function with the correct inputs and the output must be a scalar.
- **Feedback Points** - Here the user must specify how many points the participant receives for each outcome. This information will then be recorded in the output file for a posterior easier analysis of the data.

Feedback points	Feedback Text	Upper left corner x	Upper left corner y	Down right corner x	Down right corner y
1	arrowG.png	0,333333333	0,333333333	0,666666667	0,666666667
-1	arrowR.png	0,333333333	0,333333333	0,666666667	0,666666667
0	bar.png	0,333333333	0,333333333	0,666666667	0,666666667

Figure 3.5: Table used for setting up the present task.

2. ConditionsNoGo sheet

In this sheet, the user can specify the outcomes and its probability distribution when subjects do not press the key. The layout is exactly the same as described in the ConditionsGo sheet.

3. Groups sheet

   In the Groups sheet the user have to specify all the group conditions (e.g patient, control) to match each subject to his own group. This list will appear as a pop up menu in the dialog box displayed at the beginning of the task.

4. ImageSettings

   In this section, the user must set some features of the images displayed as stimuli. The user has to specify the coordinates of both upper left and down right images corners. By doing this the user is specifying at the same time the size and image position. In other words, instead of specifying the usual four degrees of freedom (height, width, center coordinates), the corners coordinates are specified. The coordinates are expressed in pixel units. The images are saved under the folder "FeedbackImages".

	x	y
Upper Left Corner	0,333333333	0,333333333
Down Right Corner	0,666666667	0,666666667

Figure 3.6: Table where the images corners coordinates are specified.

5. TimeSettings

   As mentioned above, all the events in a single trial are separated in time. This is of utmost importance not only to distinguish them during the task, but also if the task is used to perform an fMRI analysis. In this sheet there is a column called "Time Distributions" (figure 3.7) where the user has to define the probability distributions of the duration of the different events in each trial (Fixation, Stimulus, Loading, Feedback and Blank). These distributions are MATLAB® functions which generate a random time vector out of a distribution (e.g. normrnd(2,0,1,40) generates a 40 length row vector based on a normal distribution)[2]. The function output must be a vector containing a time interval for each trial. The loading event distribution is the same as the delay distribution, thus its cell should be left in blank, because its duration is set up in table 3.4.

Trial phase	Time Distributions
Fixation	normrnd(1,0)
Stimulus	normrnd(1.5,0)
Loading	
Feedback	normrnd(1,0)
Blank	gamrnd(9/2,2/3)

Figure 3.7: Table to set the time distributions.

6. TaskSettings

   In this sheet the user has to define the number of blocks and the number of time an image is displayed within a block.

[2]If the user wants a fixed time interval he still must use a probability distribution by setting the variance as 0.

3.2.1 Output Data

The output data is recorded in a file saved under the folder "OutputData" as "output#subjectID.xlsx". This excel file has two sheets: "Data" and "Observations".

- Data

  The data is divided into 14 categories organized as columns in a table whose rows correspond to each trial. A full description of each category is given below.

  1. **Subject ID** - The subject identification number provided at the beginning of the task.
  2. **Subject Group** - The group condition which the subject belongs to (e.g. ADHD, OCD, Control, etc).
  3. **Block Number** - The block number.
  4. **Trial Number** - The trial number.
  5. **Condition** - Name of the condition presented in a trial (e.g Go to win).
  6. **Image** - Name of the images presented in a trial.
  7. **User Press** - Categorical variable which codes whether the subject pressed (1) or not (0) in each trial.
  8. **Reaction Time** - Time between the onset stimulus and the subject's response. It is measured in seconds.
  9. **Feedback** - Number of points gained in a trial.
  10. **Fixation Onset** - Time onset of the fixation event. It is measured in seconds.
  11. **Stimulus Onset** - Time onset of the stimulus. It is measured in seconds.
  12. **Load Onset** - Time onset of the loading event. It is measured in seconds.
  13. **Feedback Onset** - Time onset of the feedback. It is measured in seconds.
  14. **Blank Onset** - Time onset of the blank event. It is measured in seconds.

3.2.2 Running the experiment

When the user runs the task, a dialogue box is displayed on the screen with a blank box and a pop-up menu (figure 3.8). The subject ID (may be a number or a string) must be inserted in the blank box and the group condition chosen from the conditions given by the pop-up menu.

After the task ends, a dialogue box is displayed on the screen where the user can take any note (e.g. something unusual that happened during the task).

As it was previously said, this task is also going to be performed in an MRI scanner. The MRI setup only runs tasks built in E-prime® which consists in a suite of applications to design computerized paradigms. Despite all the efforts, we were not capable of managing to run the task programmed in MATLAB®. For this reason, we reprogrammed it in E-prime® version 1.2. Five of our subjects performed the task in an E-prime® layout. Since the task characteristics are the same we decided to include this set of subjects in our analysis.

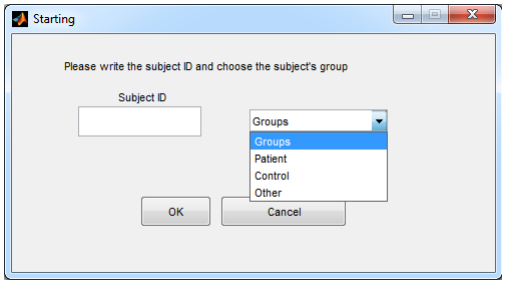

Figure 3.8: Dialogue box displayed at the beginning of the task. In the blank box the user must insert the subject's identification and from the pop-up menu choose the group.

3.3 Q-learning models

We built 4 parametrized QL learning models to fit to the behavior of the subjects. All models assigned a probability to each action a_t on trial t according to the action weight $W(s_t, a_t)$. This was based on the softmax method:

$$P(a_t|s_t) = \pi(s_t, a_t) = \frac{e^{\beta W(s_t,a_t)}}{\sum_{b\in A(s)} e^{\beta W(s_t,b)}} \tag{3.1}$$

where β was constrained to be always positive. The states s_t corresponded to the five conditions (go to win, go to avoid losing, nogo to win, nogo to avoid losing, neutral) and, in each state, there were two possible actions a_t, respond (go) or not respond (nogo).

The models further differed in terms of how action weight was constructed. For the standard QL, $W(s_t, a_t) = Q(s_t, a_t)$, which was updated recursively according to the equation 3.2.

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t - Q(a_t, s_t)] \tag{3.2}$$

The parameter α was the learning rate and it was free to vary between 0 and 1. The reinforcements entered the equation through $r_t \in \{-1, 1, 0\}$.

The standard model assumed that the initial Q-value was zero, and thus both actions (go and nogo) were equally probable. However, subjects might exhibited a natural bias towards the go or nogo action. The model QL+Q_0 (equation 3.3) captured this initial bias by allowing the Q-value on the first trial to vary freely, while for all other models this was set to zero.

$$Q(s_{t=0}, go) = Q_0 \tag{3.3}$$

The parameter Q_0 was constrained to be always between -1 and 1.

This tendency to perform an action could also be mathematically translated into a bias parameter which was kept constant across the experiment. This parameter represented the general tendency to perform an action, whereas the Q_0 was the *a priori* bias which could change according to new outcomes. Therefore, for the QL+*bias* model, the action weight was modified to include a static bias parameter b according to the equation 3.4.

$$W(s_t, go) = Q(s_t, go) + b \tag{3.4}$$

In order to capture the action by valence interaction, a Pavlovian parameter was included in the action weight (equation 3.10).

$$W(s_t, go) = Q(s_t, go) + \pi V(s_t) \tag{3.5}$$

The parameter π was constrained to always be positive. Thus, for conditions in which the outcomes were mostly negative ($V(s_t) < 0$), the Pavlovian factor decreased the tendency to go, while it promoted the tendency to go in conditions where the outcomes are mostly positive ($V(s_t) > 0$).

The state-value function was evaluated according to the equation 3.6.

$$V(s_t) \leftarrow V(s_t) + \alpha[r_t - V(s_t)] \tag{3.6}$$

3.4 Actor-Critic models

As for the AC framework, we built 4 parameterized learning models. Similarly to the previous case, all models assigned a probability to each action a_t on trial t according to the action weight $W(s_t, a_t)$. However, only the weight that depends on the go action is considered (equation 3.7).

$$\begin{array}{rcl} P(go|s_t) = \pi(s_t, go) & = & \frac{e^{\beta W(go,a_t)}}{e^{\beta W(go,a_t)}+1} \\ P(nogo|s_t) = \pi(s_t, nogo) & = & \frac{1}{e^{\beta W(go,a_t)}+1} \end{array} \tag{3.7}$$

For the standard AC model, $W(s_t, go) = p(s_t, go)$, which was updated recursively according to equation 3.8.

$$p(s_t, go) \leftarrow p(s_t, go) + \eta[r_t - V(s_t)] \tag{3.8}$$

The critic updated the state-value function according to the equation 3.6.

However, the standard AC model did not take into account that subjects might exhibit a go bias. Consequently, we tested the model AC+p_0 (equation 3.9) which captured this bias by allowing the preferences on the first trial to vary freely. Unlike Q_0 in the QL framework, prediction errors were not determined by the preferences, and, thus, p_0 was not erased as the subjects learned, remaining static throughout time.

$$p(s_{t=0}, go) = p_0 \tag{3.9}$$

The p_0 is a free parameter and it is constrained to be between -1 and 1.

For the model including the Pavlovian factor (AC+*pav*), the action weight was modified in the same way as in the QL+*pav* model,

$$W(s_t, go) = p(s_t, go) + \pi V(s_t) \tag{3.10}$$

3.5 Maximum likelihood estimation for RL

In order to determine which model best describes the behavioral data, we fitted the aforementioned models to individual behavioural data. The fitting procedure consisted of estimating the individual model parameters using the maximum likelihood estimation method.

As described in the previous section, the models have in common a basic structure composed by a learning and a decision model. Additionally, they share the same decision model which is defined as a Gibbs softmax distribution. This probability function, or likelihood function, describes how likely a given set of individual choices D_s is, given the model M and the set of parameters θ_s, i.e. $P(D_s|\theta_s, M)$. However, we are interested in finding the probability of a given set of parameters, given the individual data and the model, $P(\theta_s|D_s, M)$, known as the posterior probability.

According to the Baye's rule, the posterior probability distribution over the parameters for one subject s is given by equation 3.11.

$$P(\theta_s|D_s, M) = \frac{P(D_s|\theta_s, M)P(\theta_s|M)}{P(D_s|M)} \tag{3.11}$$

D_s denotes the set of choices made by subject s. This is a categorical variable which indicates whether the subject pressed or withheld from pressing. θ_s denotes the set of parameters of subject s in model m.

Ignoring the normalization constant, $P(D_s|M)$, the posterior probability is proportional to the product of the likelihood function and the prior probability over the parameters, $P(\theta_s|M)$. For the purposes of parameters' calculation, we assumed a non-informative prior. Therefore, the likelihood function will be proportional to the posterior probability (equation 3.11).

$$P(\theta_s|D_s, M) \propto P(D_s|\theta_s, M) \tag{3.12}$$

Given the model M and the sequence of choices made by subject s along T trials ($D_s = c_{s_1}, ..., c_{s_T}$), it is straightforward to determine the likelihood function for each subject by applying the chain rule[3] (equation 3.13).

$$P(D_s|\theta_s, M) = \prod_{i=1}^{T} P(c_{s_i}|\bigcap_{j=1}^{i-1} c_{s_j}, \theta_s, M) \tag{3.13}$$

The maximum likelihood estimation (MLE) method adopts a classical approach by seeking the set of parameters which maximizes the likelihood function (equation 3.14). This way, it takes the derivative of the likelihood function with respect to θ_s in order to find an estimate of the true set of parameters, $\hat{\theta_s}$. Note that the terms in equation 3.13 are determined by the softmax function.

$$\hat{\theta_s} = max_{\theta_s} P(D_s|\theta_s, M) \tag{3.14}$$

The result of equation 3.13 is often a very small number, thus it is numerically more stable to compute its logarithm. Since the logarithm is a monotonic function, the optimum of this quantity remains the

[3] In probability theory, the chain rule permits the calculation of the joint distribution of a set of random variables using only conditional probabilities, e.g. given 3 random variables A, B, C, the joint probability is given by $P(A, B, C) = P(A|B, C)P(B|C)P(C)$

same but is less likely to underflow the minimum floating point value representable by a computer [10]. Furthermore, the logarithm transforms a product of factors into a sum which makes easier to compute the derivative of equation 3.13. For all of these reasons, the quantity which will be used to determine the parameters which best fits the data will be the log-Likelihood (LLH).

The optimization was performed using the Matlab routine *fmincon*. This algorithm allows us to find parameters that minimize a function while satisfying constraints. Consequently, we minimize -LLH which is the same as maximize the LLH function. Furthermore, as a gradient-based method, it also computes an approximation of the Hessian matrix evaluated at the minimum point, which is extremely important to estimate the variances for each parameter. However, this method is not very accurate in finding minimums in non-smooth surfaces where there are several local minimums. Generally, it finds the minimum closest to the initial point. In order to overcome this issue, we run *fmincon* for a set of different starting points widely dispersed over the search domain.

3.6 Model Comparison

3.6.1 1st level inference

According to Neymann-Pearson lemma, the best statistic to compare models is the probability of observing the data under one model divided by the probability under another model [47]. This is known as the likelihood ratio. This quantity allows us to asses how likely a particular level of improvement is in a model fit's to data, i.e. if the improvement is due to adding an unnecessary parameter and fitting noise.

This is accomplished by performing a likelihood ratio test. This test consists of computing the probability of the observed likelihood ratio under the null hypothesis that the simpler model is correct, and thus (if this p-value is low) reject the simpler model with confidence [10]. However, one can only use this approach to compare nested models[4] [10, 47]. Since we are interested in comparing non-nested models (Q-learning and actor-critic), we turned to the Bayesian methods which can deal with both type of models. The Bayesian framework uses the ratio of model evidences, also known as Bayes factor. The evidence is the probability of obtaining observed data (D_s) given a particular model (M), i.e. $p(D_s|M)$. This quantity is computed according to the equation 3.15.

$$p(D_s|M) = \int p(D_s|\theta, M)p(\theta, M)d\theta \tag{3.15}$$

As one can notice, the model parameters are averaged out according to their prior probability ($p(\theta, M)$). This fact makes model evidence not depending on the parameters and thus it avoids over-fitting. In other words, the model evidence incorporates automatically the Occam's razor by imposing penalty on more complex and flexible models [10]. Since $p(D_s|M)$ is a probability distribution, it must sum to 1 over all possible data sets, thus in more flexible models, which achieve good fit to many data sets, the evidence assigned to each data set will be lower.

[4] Models are nested when the simpler model can be obtained by restricting a parameter in the complex model by setting it to zero

Generally, computing the integral in equation 3.15 is intractable and numerically difficult. Therefore, it is necessary to use computationally tractable approximations to the model evidence, or equivalently, to the logarithm of the model evidence (log-evidence). One of the approximations of this integral is the Laplace approximation (equation 3.16) [22].

$$\log P(D_s|M) \sim \log P(D_s|M,\theta) + \log P(\theta|M) + \frac{n}{2}\log 2\pi - \frac{1}{2}\log \frac{1}{|H|^{-1}} \tag{3.16}$$

This method makes a Gaussian like approximation around a maximum *a posterior* estimate. This assumption is based on the large data limit, thus the integral will be poorly approximated for small data sets. It has been shown that samples of size greater than $20n$ (with n being the number of parameters) are large enough for the method to work well [22]. In our case, this criterion is fully accomplished, and it is, actually, surpassed. However, this Gaussian approximation is also poorly suited to constrained parameters, since it allows non-zero probability outside the parameter domain. There is a worse problem called degeneracy which arises when parameters are redundant. In this case, the posterior will contain an infinity of different configurations with the same likelihood. Therefore, if a non-informative prior is used, the volume element $|H^{-1}|$ will be infinite [4]. We have tested for this method, using Jeffrey's non-informative prior for the parameters (θ) [22], and, for the lion share of the subjects, we obtained very high log-evidences which biased our results. This effect was caused by the strong coupling between the parameters which makes $|H^{-1}|$ very high. This way, we turned to the BIC (equation 3.17) [44] which can be obtained from equation 3.16 in the limit of large data.

$$\log P(D_s|M) \sim \log P(D_s|M,\theta) - \frac{n}{2}\log m \tag{3.17}$$

Where n is the number of parameters and m is the number of datapoints. The BIC approximation is appealing in that it can be applied even if one does not know which prior to set [22]. Furthermore, it is proven that BIC is a reasonable good approximation to the Laplace equation when using the Jeffrey's non-informative prior [22]. Nevertheless, BIC does not take into account the real contribution of a parameter to explain the observable data, it overpenalizes the model accuracy and, thus it tends to favour the simplest model [10]. However, weighing the pros and cons, we decided to use the BIC approximation.

3.6.2 2nd level inference

Given the model evidence approximations discussed in the previous section, we now consider model comparison at the group level. One can address this issue adopting a fixed or random effects approach [10, 47]. The fixed effects analysis assumes that subjects' data is generated by the same model, and thus, one can can aggregate the individual log-evidences to compute the model log-evidence of the full dataset (equation 3.18). These aggregates are known as the group Bayes factor and they can then be used to decide which model best describes the subjects' data.

$$\log P(D_1, D_2, ..., D_N|M) = \sum_{s=1}^{N} P(D_s|M) \tag{3.18}$$

N is the number of subjects. This approach assumes that the model M is a fixed feature of the population, i.e different subjects cannot follow different models. Therefore, this approach is discarding the inter-subject variability, which makes it extremely sensitive to the presence of outliers [47]. We address this issue by adopting a random effect analysis which takes into account inter-subject heterogeneity. This procedure can be done according to a classical or Bayesian approach.

3.6.3 Classical inference

In the classical setting one uses the log-evidences across subjects, testing the null hypothesis that one model is no better than the other. This is accomplished by performing a simple two-sample t-test on the log-evidence differences. The t-test assumes that log-evidence differences are normally distributed, thus a test of normality must be performed. Here, we adopted the Kolmogorov-Smirnov test to test for this parametric assumption. Whenever this test rejects the null hypothesis of a normally distributed data, we use a Wilcoxon signed rank test which does not make any distributional assumptions [47]. This classical random effects approach can generate incorrect results when the inter-subject variability is highly due to outliers, i.e. one might fail to reject the null hypothesis that one model is no better than the other due to the presence of outliers [47]. A more robust method to outliers was recently presented which consists in a Bayesian framework.

3.6.4 Bayesian inference

The classical inference described in the previous section uses the sample variance to compute the t-statistics and thus it is sensitive to the influence of outliers, i.e. the sample variance increases monotonically with the magnitude of the outlier. A hierarchical Bayesian method recently described [47] shows more robustness to the presence of outliers. This approach is based on a hierarchical Bayesian model, where each subject model is sampled from a multinomial distribution and then individual data is generated under that subject-specific model. This hierarchical model is illustrated graphically in figure 3.9.

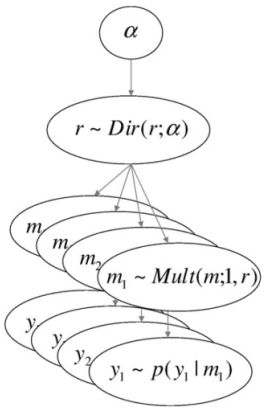

Figure 3.9: Hierarchical Bayesian model used in the Bayesian model selection approach. Individual data is generated according to a model sampled from a multinomial distribution where each model has a probability r of being sampled. The probability of a each model in the population (r) follows a Dirichlet distribution with parameter α. α = parameters of the Dirichlet Distribution;r = probabilities of the model; m = model labels; y = observed data. Adapted from [47]

Thereby, the multinomial distribution parameters (r) are the probability of each model in the population. The model probabilities follow in turn a Dirichlet distribution with parameter α. In order to compare models, we are interested in the density from which models are sampled to generate subject-specific data. In other words, we seek the probability of the multinomial parameters given the data, i.e. $P(r|D)$, where D is the subject's data. This can be achieved by inverting the hierarchical Bayesian model using an variational Bayes approach [47], which only needs to be feed with the log-model evidences of each subject. This method was implemented using the spm_BMS function from the SPM8 toolbox in MATLAB. The density over r can be used to quantify our belief that a particular model $k1$ is more likely than other model $k2$, given the observed data, denominated as exceedance probability (ϕ) (equation 3.19)

$$\phi_{k1} = P(r_{k_1} > r_{k_2}|y) \tag{3.19}$$

Since $r_{k_1} + r_{k_2} = 1$, equation 3.19 can be re-written as $\phi_{k1} = P(r_{k_1} > 0.5|y)$.

The exceedance probability might thus be used as a quantitative measure to compare models. However, it differs from the conventional posterior probability of a model in Bayesian model comparison. The exceedance probability is a statement of belief about the posterior probability, not the posterior itself. So, for example, when the exceedance probability is 98%, it means that we can be 98% confident that the favoured model has a greater posterior probability than any other model tested. This measure has two main advantages: 1) sensitivity to the confidence in the posterior probability and 2) easy interpretability. There is no defined threshold on the exceedance probability. In the article [47] where this method is firstly described, it is not defined from which value of exceedance probability one might consider having enough confidence on the posterior probability of a given model. For instance, in the same article, an exceedance probability of $\pi = 92.8\%$ was considered an evidence that the favoured model was superior. Therefore, during model comparison we will not advocate that a certain model is the best, but that it is more likely to be superior with a probability given by the exceedance probability.

3.7 Other machine learning methods

Apart from the model fitting, in this study other machine learning methods were also applied to the behavioral raw data in order to extract some relationships among conditions. Since these associations depend on whether the subjects follow a Q-learning or an actor-critic model, they can provide evidences towards one of the models. The methods covered in the present study were the principal component analysis and the Gaussian mixture model.

3.7.1 Gaussian mixture model

We consider the problem of finding meaningful subgroups in a set of data points when no information other than the observed values is provided. This problem is known as cluster analysis [14]. Probability models have been proposed as a basis for cluster analysis. In this approach, the data is assumed as coming from a mixture of probability distributions, each representing a different cluster.

In this study, we used a clustering methodology based on a simple linear superposition of Gaussian distributions. This method is known as Gaussian mixture model.

A Gaussian mixture distribution can be written as a linear superposition of N Gaussians as the form described in equation 3.20.

$$p(x) = \sum_{k=1}^{N} \pi_k N(x|\mu_k, \Sigma_k) \tag{3.20}$$

Each Gaussian density $N(x|\mu_k, \Sigma_k)$ represents a subgroup and has its own mean μ_k and covariance Σ_k. The parameters π_k correspond to the the mixing coefficients. Equation 3.20 is equivalent to equation 3.21.

$$p(x) = \sum_{k=1}^{N} p(k)p(x|k) \tag{3.21}$$

in which we can view $\pi_k = p(k)$ as the prior probability of picking the k^{th} subgroup. $p(x|k) = N(x|\mu_k, \Sigma_k)$ is the likelihood of picking up the data point x in the k^{th} subgroup. In order to determine to which subgroup a particular data point belongs, we are interested in the posterior probabilities, $\gamma_k = p(k|x)$, which are also know as responsibilities. From Baye's theorem these are given by equation 3.22.

$$\gamma_k(x) = p(k|x) = \frac{p(k)p(x|k)}{\sum_l p(l)p(x|l)} \tag{3.22}$$

In order to determine which subgroup a data point belongs to, we simply compare the responsibilities of different subgroups and retain the subgroup with higher probability.

This analysis was performed in MATLAB® version R2012b using the toolbox SPM version 8. The algorithm used applies a Gaussian mixture model to a dataset using a variational Bayesian framework [2].

3.7.2 Principal component analysis

This technique reduces the dimensionality of the data while retaining most of the variance. It accomplishes this reduction by identifying directions, called principal components, along which the variation in the data is maximal [39]. When mining a dataset comprised of numerous variables, it is likely that subsets of variables are highly correlated with each other. Therefore these variables are redundant and thus share the same driving principle in defining the outcome of interest. The principal components can thus reveal relationships between variables and identify new meaningful underlying factors.

These components simply corresponds to the eigenvectors of the data covariance matrix. Instead of the covariance matrix, we can use the correlation matrix which is like a covariance matrix but first the variables have been standardized by setting all variances equal to one. This standardization gives the values in terms of standard deviation units from the variable's mean, and thus makes them comparable between different variables. Additionally, the results obtained using the covariance matrix poorly revealed the correlations measured by the Pearson's correlation coefficient. This way, it seemed that in this situation the correlation matrix captures better the associations among variables. For these reasons, we chosen the correlation matrix.

The eigenvector associated with the largest eigenvalue has the same direction as the first principal component. On the other hand, the eigenvector associated with the second largest eigenvalue determines

the direction of the second principal component, and so on and so forth. Since both covariance and correlation matrices are square symmetric, the eigenvectors are all orthogonal to each other.

The fraction of an eigenvalue out of the sum of all eigenvalues represents the amount of variation accounted by the corresponding eigenvector. The squared components of the eigenvectors determine the contribution of each variable to explain the variance accounted for the principal component.

After calculating the principal components, we need to determine the number of factors to retain, i.e. include the factors with relevant information and exclude the factors which might represent noise. Unfortunately, there is not an indisputable approach for the determination of the number of factors and thus it is suggested to rely on multiple criteria to make this decision [13].

The best know and most utilized in practice is the K1 method [21, 25]. According to this rule, only the factors that have eigenvalues greater than one are retained [25]. We will combine the former method with another popular approach: the Cattell's scree test [5]. In this method, the variances explained by each component are presented in descending order and linked with a line. Afterwards, the graph is examined to determine the point at which the last significant drop or break takes place. The Cattell's scree test says that this point divides the important or major factors from the minor or trivial factors. Therefore, we must discard all the components located to the right of this point.

4

Behavioural analysis

Contents

This chapter describes the statistical analysis of the behavioral data. Two different approaches were employed. The first approach did not consider the tendency to respond whereas the second approach took into consideration such a possible go bias when determining whether subjects have learned each condition. Both approaches revealed that subjects learned correctly the go to win condition. However, they disagree in the learning performance of the go to avoid losing and nogo to avoid losing. When the tendency to respond is included, the nogo to avoid losing does not show a proper learning whereas the go to avoid losing does. The opposite occurs when the go bias is not considered. However, both measures agree in the existence of a strong action by valence interaction. Particularly, the subjects tend to learn better the go to win comparing to the go to avoid losing condition and nogo to avoid losing comparing to the nogo to win condition. This finding is in line with the Pavlovian effect previously described.

4.1 Subjects

24 adults participated in this experiment (10 females and 14 males) whose age ranges from 19 to 50 years, mean=25.38, SD=7.82 years. All the participants are friends or relatives of the people who work in the laboratory of Professor Tiago Vaz Maia in Instituto de Medicina Molecular. All the participants performed the task voluntarily.

4.2 Behavioural statistical analysis

4.2.1 Behavioural analysis without the neutral condition

To characterize the group performance in the task, we have done a repeated measures three-way ANOVA for the number of correct responses, with factors of block (3 levels), valence (win/avoid losing: 2 levels) and action(Go/NoGo: 2 levels). We have gone for a repeated measures approach, because the same subjects were used between different levels of a factor which disrupts the assumption of data independence made by a simple ANOVA. The number of correct responses corresponds to the number of times the participants pressed in the go to win and go to avoid losing conditions whereas in the nogo to win and nogo to avoid losing conditions it was the number of times the participants did not press. Correct responses were collapsed into bins of 10 trials per condition. This statistical analysis was performed in SPSS® version 16.

In figure 4.1 is depicted the time varying probability, across subjects, of making the go action for each of the five conditions.

The temporal dynamics of the curves shows that subjects seem to have learned correctly at most three conditions, namely, go to win, go to avoid losing and nogo to avoid losing. The three way repeated measures ANOVA showed a main effect of block ($F(2, 46) = 10.742, p < 0.001$), which indicates that the number of correct responses increased across blocks. This tendency is depicted in figure 4.2.

A post hoc paired t-test revealed a significant difference in the number of correct responses between the first and the second blocks ($t(23) = -2.720, p = 0.012$), the second and the third blocks ($t(23) = -2.465, p = 0.022$) and the first and the third blocks ($t(23) = -3.884, p = 0.001$). This evidence could suggest an overall correct learning of the task. However, the presence of a significant block by valence

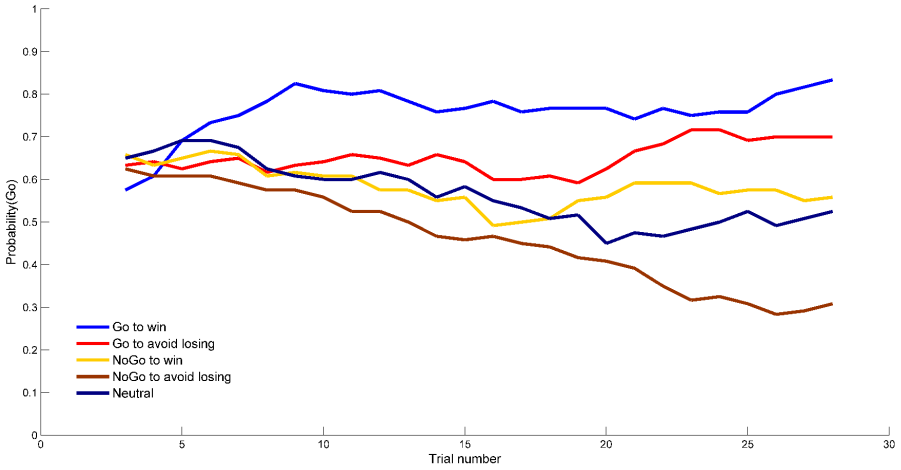

Figure 4.1: Average time varying probability, across subjects, of making the go action for the five conditions convolved with a central moving average with length 5.

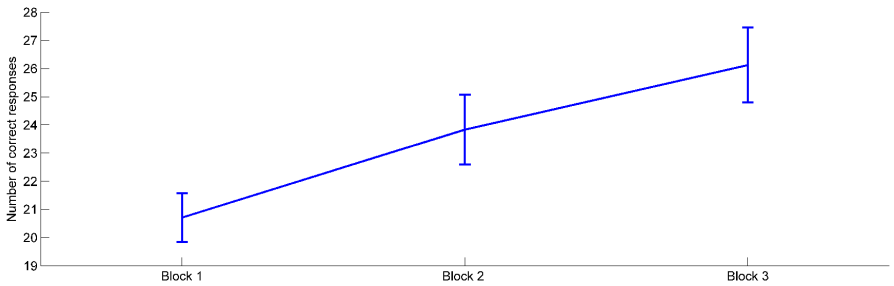

Figure 4.2: Main effect of number of correct responses in blocks. The error bars depict standard error of the mean (SEM).

interaction ($F(2,46) = 4.643, p = 0.015$) does not permit to make such conclusion. Therefore, in order to determine whether all conditions were effectively learned, we performed a repeated measures one-way ANOVA for the number of correct responses with factors of block for each condition. The go to win ($F(2,46) = 4.031, p = 0.024$) and nogo to avoid losing ($F(2,46) = 14.506, p < 0.001$) conditions exhibited a main effect of block, whereas the go to avoid losing ($F(2,46) = 2.793, p = 0.072$) and nogo to win ($F(2,46) = 0.934, p = 0.4$) did not. These tendencies are graphically depicted in figure 4.3. The results suggest that subjects learned well the conditions where action-valence interaction is not disrupted, whereas in conditions where this interaction is disrupted it seems that subjects did not learn. This effect could be explained by the Pavlovian effect. Since outcome is always zero, we expect no learning in the neutral condition which is in agreement with the non significant main effect of block ($F(2,26) = 10.889, p = 0.06$).

According to the ANOVA test, there is a strong action by valence interaction ($F(1,23) = 11.581, p = 0.002$). Figure 4.4 shows this interaction and suggests that subjects perform better when they have to press to win and when they have to not press to avoid punishment. A post hoc paired t-test revealed that there is no significant difference between the number of correct choices in the go to win and go to avoid losing conditions ($t(23) = 1.491, p = 0.15$). Likewise, the number of correct responses did not differ

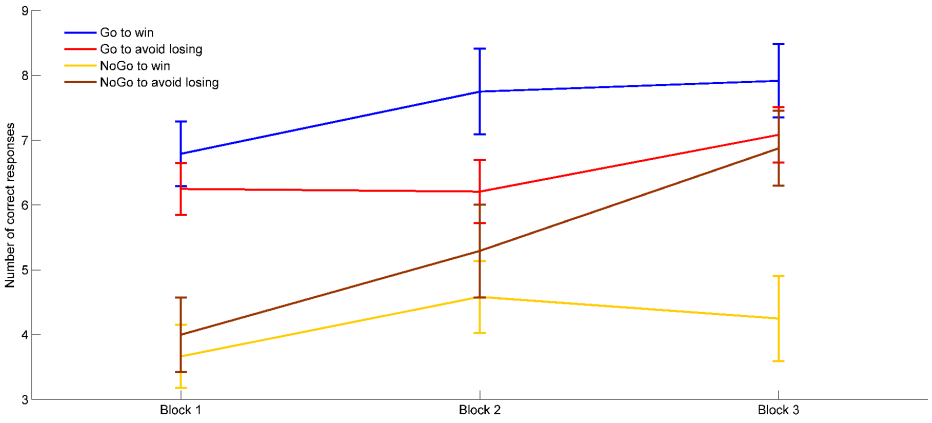

Figure 4.3: Number of correct response per block and condition. The error bars depict the standard error mean.

in the nogo to win and nogo to avoid losing conditions ($t(23) = -1.984, p = 0.059$). These observations might result from the aggregation of the number of correct responses across time which might mask some differences between conditions. Actually, the three way ANOVA revealed a significant action by valence by block interaction ($F(2, 46) = 5.089, p = 0.01$), which does not allow us to directly interpret the action by valence interaction results.

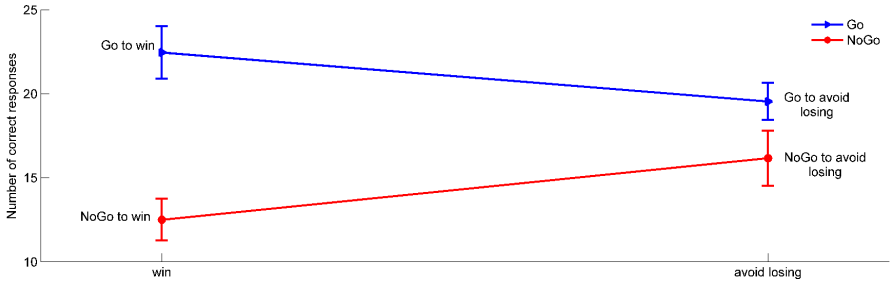

Figure 4.4: Interaction between action (go and nogo) and valence (win and avoid losing). The error bars depict standard error of the mean (SEM).

According to figure 4.3, in the go conditions, the first and third blocks show a similar percentage of correct responses which suggests that, at the beginning of the task, the subjects showed the same tendency to go and, at the end of the task, they were capable of performing equally in both conditions. However a more prominent difference is depicted in the second block, which suggests that, despite of the similar performance at third block, it is more difficult to learn the go to avoid losing condition. This difference was clearly hidden due to the aggregation of all responses across time. A post hoc paired t-test revealed a trend towards statistical significance in the difference between both conditions in the second block ($t(23) = 2.057, p = 0.051$). Since this result showed a p-value very near the threshold, possibly because of insufficient data, it is worth to also take it into account.

Regarding the nogo conditions, figure 4.3 exhibit the existence of a significant difference in the percentage of correct responses in the third block ($t(23) = -3.784, p = 0.001$). The significant action by valence by block interaction is likely to reflect this difference. We also tested for the difference in the rest of the blocks, but they were were not significant ($t(23) = 0.460, p = 0.650$, $t(23) = -0.893, p = 0.381$,

first and second block, respectively).

Thus the behavioural data indicates that subjects were better at learning to go in the win condition (compared to go in the avoid losing condition), and were better at learning to withhold a response (nogo) in the avoid losing condition (compared to a similar response in the win condition). These evidences suggest the existence of an interdependence of action and valence where positive outcomes support learning of go choices and negative outcomes supports learning of nogo choices. As described in section 2.3.4, this effect, known as the Pavlovian effect, has already been reported in literature [19], i.e. the go to avoid losing condition is linked with punishments promoting the action inhibition (nogo) whereas the nogo to win is linked with reward, entailing the go action.

These findings suggest that subjects have correctly learned the go to win and nogo to avoid losing conditions, but not the nogo to win and go to avoid losing conditions. However, this might be interpreted as an effect of classical conditioning rather than bad learning. A strong action by valence interaction supports this hypothesis. The subjects seem to have learned worse the nogo to win than the nogo to avoid losing condition and exhibited more difficulties to correctly learn the the go to avoid losing than the go to win condition. These results might be explained by the Pavlovian effect which hypothesises that classical conditioning also comes into play in this task and, thus, when a stimulus has a positive value (win condition) subjects exhibit a higher tendency to respond whereas when a stimulus has a negative value (avoid losing conditions) subjects tend not to respond.

4.2.2 Behavioural analysis with the neutral condition

Due to the non-existence of outcome, the neutral condition simply measures the tendency to respond (go) or withhold from responding (nogo). This condition will be used as a baseline to determine whether the subjects did learn or not the other four conditions. The tendency to respond might be quantitatively translated as the number of times the participants pressed in the neutral condition whereas the tendency to not respond might be translated as the number of times the participants did not press in the neutral condition. To characterize the group performance in the task, we have done a repeated measures three-way ANOVA for the number of adjusted correct responses, with factors of block (3 levels), valence (win/avoid losing: 2 levels) and action(Go/NoGo: 2 levels). The number of adjusted correct responses corresponds to the number of times the participants pressed in the go to win and go to avoid losing conditions subtracted by their tendency to press whereas in the nogo to win and nogo to avoid losing conditions it corresponds to the number of times the participants did not press subtracted by their tendency to not respond. These measures were collapsed into bins of 10 trials per condition. This statistical analysis was performed in SPSS® version 16.

In figure 4.1, it is depicted the time varying probability, across subjects, of making the go action for the five conditions. Conditions whose correct response is to press, when correctly learned, will be above the neutral line, whereas conditions whose correct response is to withhold from pressing will be below the neutral line. The temporal dynamics of the curves shows roughly that subjects seems to have learned at most three conditions, namely, go to win, go to avoid losing and nogo to avoid losing. As expected, a three way repeated measures ANOVA showed a main effect of block ($F(2, 46) = 10.742, p < 0.001$),

which indicates that the number of rectified correct responses exhibit a positive trend. This tendency is depicted in figure 4.5. A post hoc paired t-test revealed a significant difference in the number of correct responses between the first and the second blocks ($t(23) = -2.720, p = 0.012$), the second and the third blocks ($t(23) = -2.465, p = 0.022$) and the first and the third blocks ($t(23) = -3.884, p = 0.001$). However, a significant block by valence interaction does not allow us to determine whether each condition was effectively learned. Therefore, we opted for performing a repeated measures one way ANOVA for the number of adjusted correct responses with factors of block for each condition.

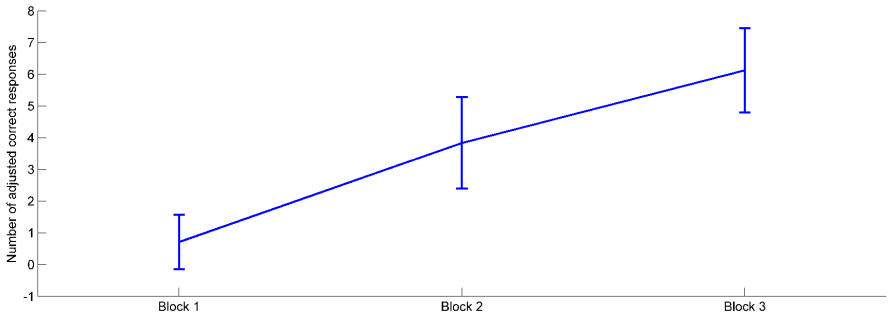

Figure 4.5: Main effect of block. The error bars depict standard error of the mean (SEM).

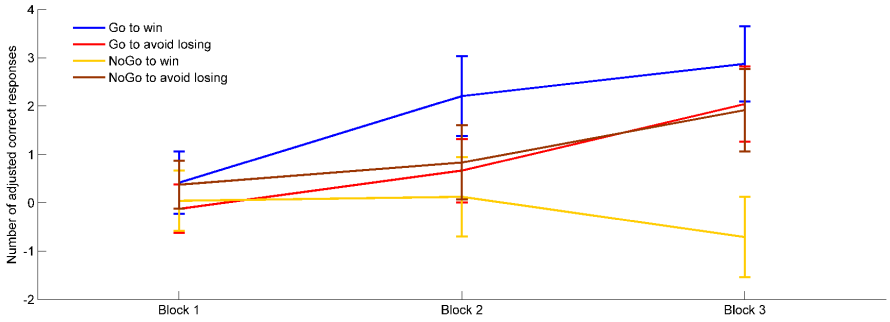

Figure 4.6: Number of adjusted correct responses per block and condition. The error bars depict the standard error mean (SEM).

The go to win ($F(2, 46) = 5.5, p = 0.007$) and go to avoid losing ($F(2, 46) = 6.154, p = 0.004$) conditions exhibited a main effect of block, whereas the nogo to avoid losing ($F(2, 46) = 2.158, p = 0.127$) and nogo to win ($F(2, 46) = 0.618, p = 0.544$) did not. Although, these results might indicate that subjects did not learn the nogo to avoid losing condition, it is clear from figure 4.6 that this condition exhibits a positive trend which it is likely to be masked by the high data variance (post-hoc paired t-test in the nogo to avoid losing condition between the first and third block, $t(23) = -1.776, p = 0089$). The higher variance presented in data comparing to the previous one results from the subtraction of the neutral condition which adds the respective variance to the variance already present in the other conditions. This lead us to believe that a higher number of subjects would unravel a positive main effect. Conversely, the nogo to win condition seems to show a less strong but negative trend (post hoc paired t-test in the nogo to win condition between the first and third block, $t(23) = 0.878, p = 0.389$) which, even with a lower variance, would indicate learning disability in this condition.

Figure 4.1 reveals that, in the first trials, all conditions exhibit the same behaviour, i.e. a high tendency to press. Therefore, the number of correct responses is high and low in the go and nogo conditions, respectively. This leads us to conclude that at the beginning, subjects have better learned the go conditions than the nogo conditions, which is an erroneous inference since all conditions exhibit the same behaviour as the neutral condition. When one uses the number of adjusted correct responses this effect vanishes. In figure 4.6, no difference is depicted in the first block between conditions whereas when simply using the correct responses a very significant difference appears (figure 4.3). This is confirmed by performing a post hoc paired t-test between go and nogo conditions in the first block (table A.2 in the appendix A). These differences explain why considering the tendency to respond shows that the subjects learn better the go to avoid losing condition and worse the nogo to avoid losing condition compared to the situation when this tendency is not taken into account.

This ANOVA test also showed the presence of a strong interaction between action and valence ($F(1,23) = 11.581, p = 0.002$). Figure 4.7 shows this interaction and suggests that subjects perform better when they have to press to win (compared to go in the avoid losing condition) and when they have to not press to avoid punishment (compared to withhold from responding in the win condition).

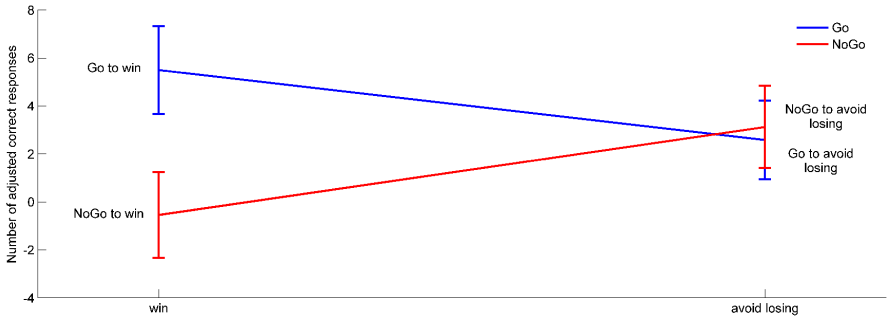

Figure 4.7: Interaction between action (go and nogo) and valence (win and avoid losing). The error bars depict standard error of the mean (SEM).

A post hoc paired t-test revealed that there is no significant difference between the number of correct choices in the go to win and go to avoid losing conditions ($t(23) = 1.491, p = 0.15$). Likewise, the number of correct responses did not differ in the nogo to win and nogo to avoid losing conditions ($t(23) = -1.984, p = 0.059$). Since there is a significant action by valence by block interaction ($F(2,46) = 5.089, p = 0.01$), it is not possible to draw conclusions about the action by valence interaction without take this 3-way interaction into consideration. This way, we had to analyse conditions individually.

Figure 4.3 reveals that, in the go conditions, the first and third blocks show a similar percentage of correct responses which suggests that, initially, the subjects show the same tendency to go and, at the end of the task, they had successfully learned both conditions. However a more prominent difference is depicted in the second block ($t(23) = 2.057, p = 0.051$), which, just like in the previous analysis, suggests that, although subjects learned correctly both conditions, it is more difficult to learn the go to avoid losing condition. Since this result showed a p-value very near the threshold, possibly because of insufficient data, and this is a pilot study we considered it statistically significant.

Additionally, figure 4.3 reveals the existence of a significant difference in the percentage of correct

responses in the third block between the nogo conditions ($t(23) = -3.784, p = 0.001$). The significant action by valence by block interaction ($F(2, 46) = 5.089, p = 0.01$) is likely to reflect this difference. The differences in the rest of the blocks were not statistically significant ($t(23) = 0.460, p = 0.650$, $t(23) = -0.893, p = 0.381$, first block and second block, respectively).

These evidences suggest that subjects learn worse in the nogo to win (comparing to the nogo to avoid losing) and go to avoid losing (comparing to the go to win) conditions which could be explained by the Pavlovian effect.

5

Comparing Q-learning and Actor-Critic

Contents

In this chapter the results from the model fitting to the behavioral data are described. The four versions of the QL and AC algorithms previously described are compared in order to determine the approach that best describes the raw data. Additionally, principal component analysis and Gaussian mixture models were also applied to the behavioral raw data in order to extract some relationships among conditions. Since these associations depend on whether the subjects follow a QL or an AC model, they can provide evidence towards one of the models. Our findings suggests that the QL framework is the most appropriate to describe the learning process in humans, albeit the AC approach seems to capture better the action by valence interaction revealed in the raw data.

5.1 Model fitting analysis

TD learning models can parametrized subjects' behavior. We adapted the parameters of a nested collection of QL and AC models (see section **??**) to the observed behavioral data, and compared the fit of their explanations using Bayesian information criterion that takes into account the data likelihood and the model complexity (see section 3.6 for further details). The models were compared at the group level using both classical and Bayesian approaches. Since the Bayesian comparison is a more recent approach than the classical analysis, and, thus, it has been less employed in similar studies, we decided to supplement it with the classical analysis. Furthermore, the Bayesian approach is more robust to outliers. We also generated surrogate data from the models by using the subjects' model parameters and simulating behavior in the task.

5.1.1 Q-learning

In order to find the model which best fits to the behavioural data, we firstly used the standard QL model which was purely instrumental with two parameters: the learning rate (α) and the inverse of temperature (β). The data sampled from the standard QL model with the parameters emerged from the model fitting are depicted in figure 5.1.

This model assumed that the Q-value at the first trial was zero, and thus, initially, both actions were equally probable. However, the subjects exhibited some initial tendency to respond which was impossible to capture using a standard Q-learning model (in figure 5.1, the initial probability of the go action was above 0.5 in most of the conditions, namely go to avoid losing, neutral, nogo to win and nogo to avoid losing). Consequently, we tested the model QL+Q_0 which accounted for this effect. The learning time course generated from the QL+Q_0 model is depicted in figure 5.2.

The t-test assumed that data were normally distributed, thus we performed the Kolmogorov-Smirnov test which was unable to reject the null hypothesis of normally distributed log-evidences ($p = 0.0971$). Applying the t-test, we verified that the QL+Q_0 model was better ($t(23) = -2.4581, p = 0.0219$) than the standard QL model. The Bayesian model comparison (BMS) analysis was also in accordance with the previous results, giving an exceedance probability of $\phi = 0.9962$ in favour of the QL+Q_0 model.

This tendency to perform the go action could also be taken into account by a bias parameter (b). Unlike Q_0, which gradually vanishes as new outcomes are integrated, the bias parameter is constant across the experiment. We thus compared the QL+*bias* model to the QL+Q_0 model.

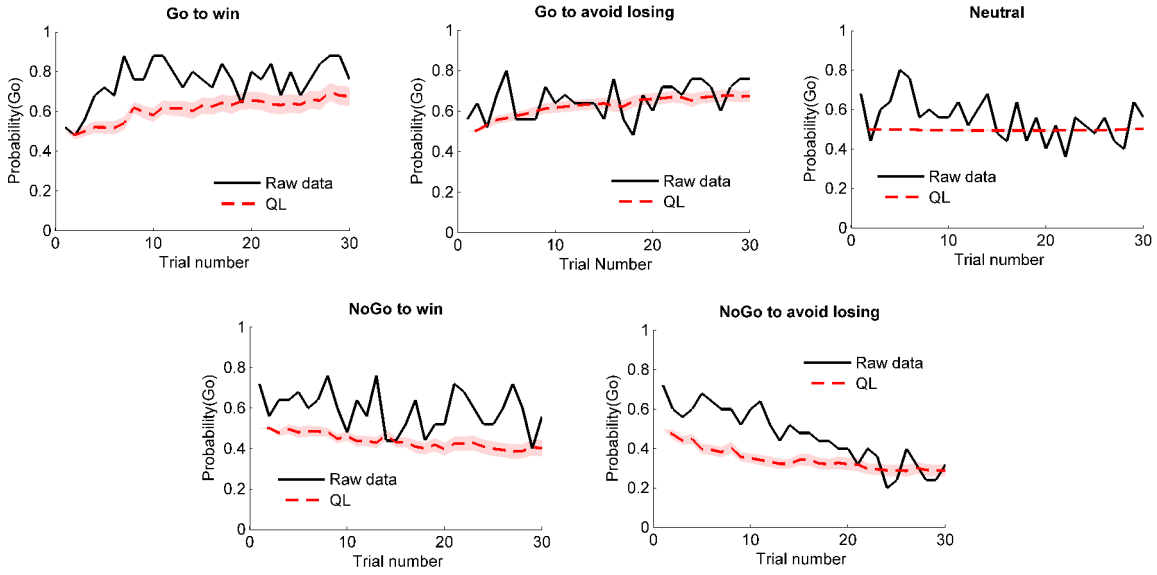

Figure 5.1: Learning time courses for all five conditions. The black lines depict the time varying probabilities, across subjects, of making a go response. The red lines represent the same time-varying probabilities, across subjects, but sampled from the standard Q-learning model.

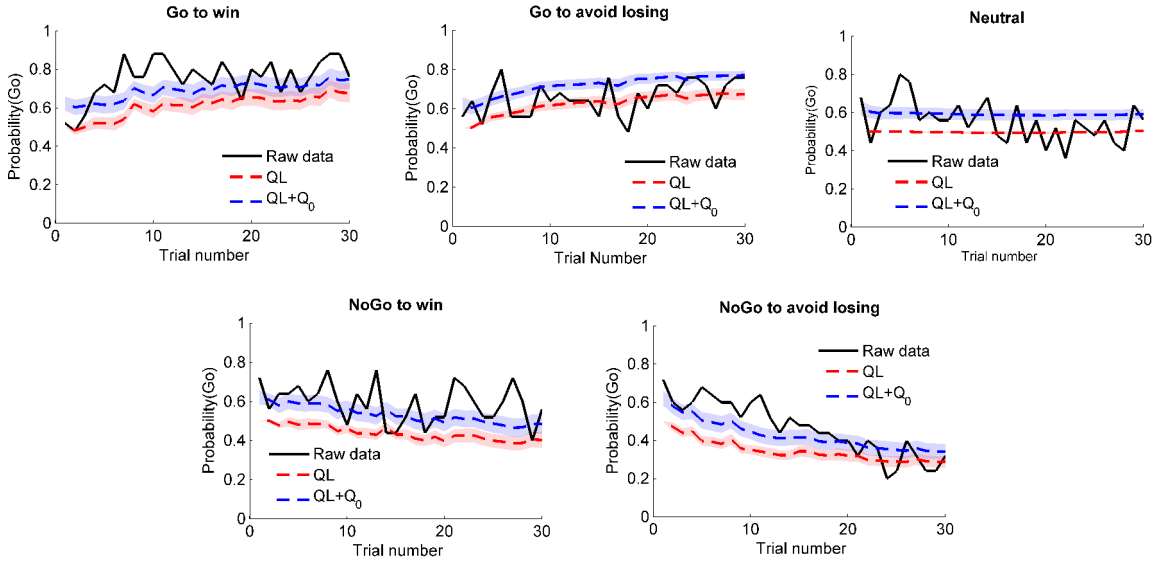

Figure 5.2: Learning time courses for all five conditions. The black lines depict the time varying probabilities, across subjects, of making a go response. The red lines represent the same average time varying probabilities, across subjects, but sampled from the standard QL model and the blue lines were sampled from the QL+Q_0 model.

The log-evidence differences were not normally distributed ($p = 0.0346$). Given this derivation from normality, we applied a non-parametric Wilcoxon signed rank test which makes no distributional assumptions. However, this test was not able to reject the null hypothesis of no difference between the models ($p = 0.4929$). Using the Bayesian approach, we obtained an exceedance probability of $\phi = 0.8160$, meaning that the QL+Q_0 model was more likely than the model QL+*bias*. Although the exceedance probability was not particularly strong, it favoured the QL+Q_0 model. Indeed the model's simulated behavior were practically the same in both models (figure 5.3).

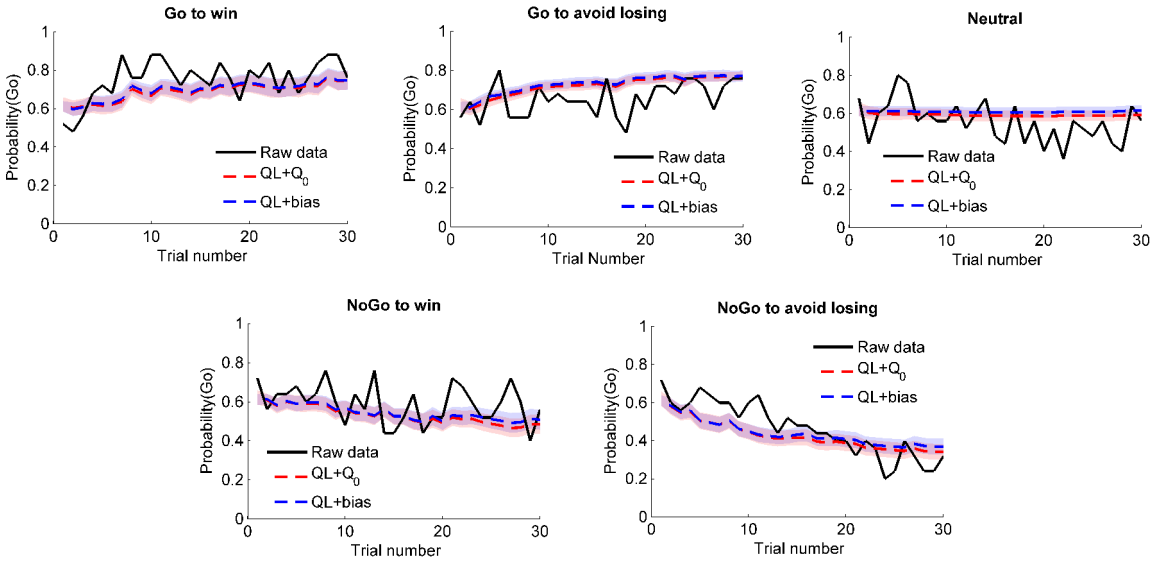

Figure 5.3: Learning time courses for all five conditions. The black lines depict the average time varying probabilities, across subjects, of making a go response. The red lines represent the same average time varying probabilities across subjects but sampled from the QL+Q_0 model and the blue lines were sampled from the QL+*bias* model.

However, the QL+Q_0 model slightly failed to capture the tendency to respond in the win conditions and the tendency to not respond in the avoid losing conditions. Particularly, this failure was more pronounced in the go to avoid losing condition (blue line in figure 5.2). Thus, we tested the QL+*pav* model which accounted for the action by valence interaction. This model added a Pavlovian factor to the other instrumental components [19]. In this model, the value of the go action was incremented proportionally to the state-value of each condition. This way, the go probability increased when the reward expectancy also increased, and the nogo probability increased when the punishment expectancy increased. For example, in the go to avoid losing condition, the stimulus embodies negative expectancy (the possible outcomes were zero (go action) or negative (nogo action)), and thus, this model promoted the inhibition of the go action. Conversely, this model promoted the go action in the nogo to win (this condition embodies a positive expectancy).

According to the Kolmogorov-Smirnov test, the log model evidences were normally distributed ($p = 0.1945$), and thus the t-test could be used to compare the QL+*pav* model to the QL+Q_0 model. The t-test did not reject the null hypothesis of no difference in model goodness ($t(23) = -1.4433, df = 23, p = 0.1624$). However, the BMS method supported strongly the model without the Pavlovian factor with an exceedance probability of $\phi = 0.9964$.

Although the QL+*pav* model was rejected, it seems that it captured better the Pavlovian effect present in the raw data (blue line in figure 5.4). This inconsistency could be explained by the the overpenalization imposed by the BIC in the model accuracy.

Thus, our computational analysis suggested that the behavioral data were determined by an instrumental controller with a strong, but not persistent bias towards emitting a go choice.

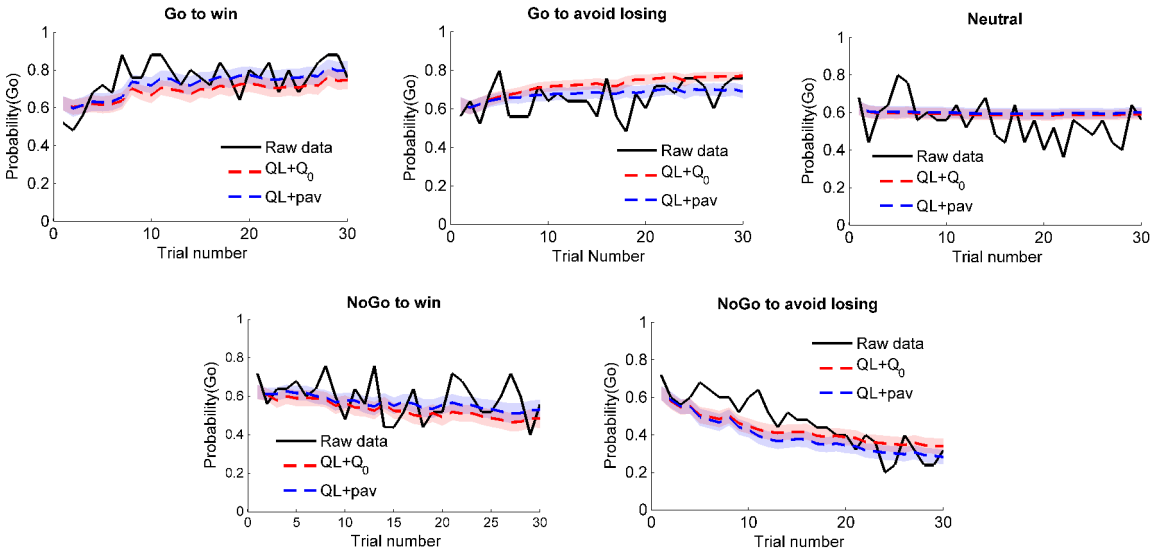

Figure 5.4: Learning time courses for all five conditions. The black lines depict the average time varying probabilities, across subjects, of making a go response. The red lines represent the same average time varying probabilities across subjects but sampled from the QL+Q_0 model and the blue lines were sampled from the QL+*pav* model.

5.1.2 Actor-Critic

Firstly, we fitted the standard AC model to the subjects' behavioral data. This model consisted of three parameters: the critic's learning rate (α), the actor's learning rate (η) and the inverse of temperature (β). The data generated from the model after the fitting procedure is depicted in figure 5.5.

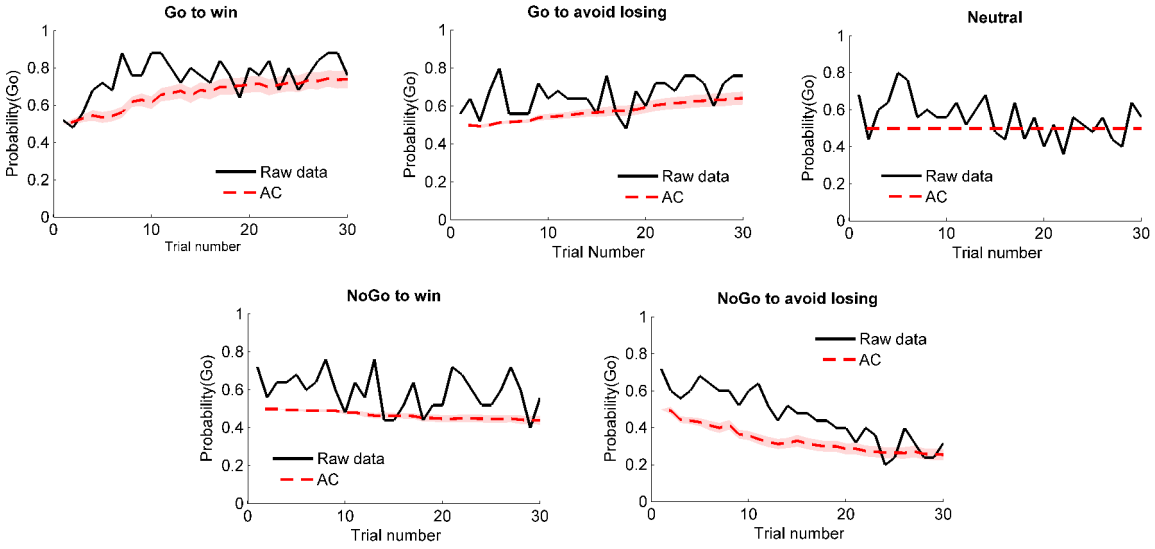

Figure 5.5: Learning time courses for all five conditions. The black lines depict the average time varying probabilities, across subjects, of making a go response. The red lines represent the same time varying probabilities, across subjects, but sampled from the standard AC model.

Similarly to the QL framework, the standard AC model failed to capture the initial tendency to perform the go action (red lines in figure 5.5). Therefore, we tested an alternative model which included an initial preference (AC+p_0), which was able to account for this effect.

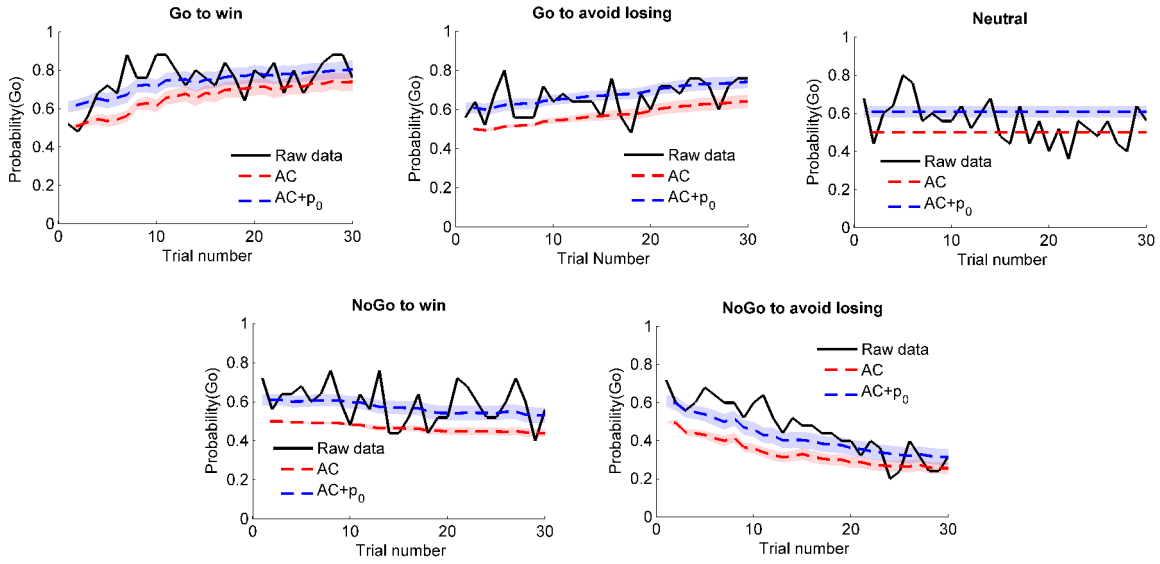

Figure 5.6: Learning time courses for all five conditions. The black lines depict the average time varying probabilities,across subjects, of making a go response. The red lines represent the same average time varying probabilities, across subjects, but sampled from the standard AC model and the blue lines from the AC+p_0 model.

The Kolmogorov-Smirnov test rejected the null hypothesis of a normally distributed log model evidence differences ($p = 0.0379$), and thus a t-test was not appropriate. Consequently, we applied a non-parametric Wilcoxon signed rank test which was indeed able to find that the AC+q_0 model explained data better than the standard AC model ($p = 0.0240$). The BMS method also strongly supported the AC+p_0 model with an exceedance probability of $\phi = 0.9957$. Indeed, the model's simulated behavior matched better the true behavior (blue lines in figure 5.6).

As is clearly evident in figure 5.6, the AC+p_0 model seemed to capture the action by valence interaction. For example in the go to avoid losing condition, the AC+p_0 model captured better the tendency to not respond than the QL+Q_0 model (blue line in figure 5.2). However, we still fitted the AC+pav model to the behavioral data and compared it to the AC+p_0 model, so that we could be sure that the Pavlovian factor was useless. Indeed, the non-parametric Wilcoxon signed rank test (the Kolmogorov-Smirnov test rejected the null hypothesis of normality ($p < 0.001$)) claimed that the AC+p_0 model was better than the AC+pav model ($p < 0.001$) which corroborated the validity of our findings. The BMS also showed a strong exceedance probability ($\phi = 1$) in favour of the AC+p_0.

Now that we found the best QL and AC models, we were able to determine which model best explained the behavioural data. The AC+p_0 model was then compared to the QL+Q_0. According to the Kolmogorov-Smirnovv test, the log model evidence differences were normally distributed ($p = 0.4520$), and thus one could perform the one tailed t-test. The former test was unable to reject the null hypothesis of no difference in the log model evidence of both models ($t(23) = 1.1742, p = 0.1262$). The BMS showed that the QL model had a higher probability of occurring in the population with an exceedance probability of $\phi = 0.8780$.

In the AC framework, subjects must learn $V(s_t)$ to compute the prediction error $\delta_t = r_{t+1} - V_t$ and then update the preferences $p(s_{t+1}, go) = p(s_t, go) + \alpha\delta$. Since the preferences are just updated when

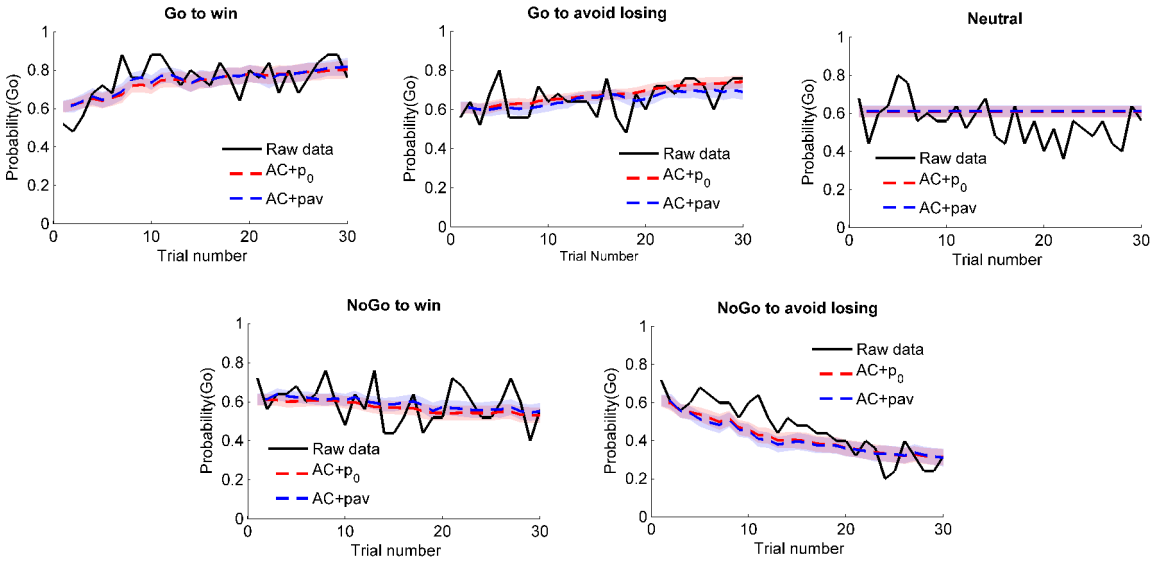

Figure 5.7: Learning time courses for all five conditions. The black lines depict the average time varying probabilities, across subjects, of making a go response. The red lines represent the same average time varying probabilities, across subjects, but sampled from the standard AC+p_0 model and the blue lines were sampled from the AC+pav model.

the subject performs the go action and, in the nogo to win and go to avoid losing conditions subjects must not respond in order to learn $V(s_t)$, the learning process tends to be slower. Put differently, in these two conditions, it is not possible to learn the state-value function while updating the preferences. The Pavlovian effect is indeed characterized by a slower learning process, therefore it could be captured better by the AC model.

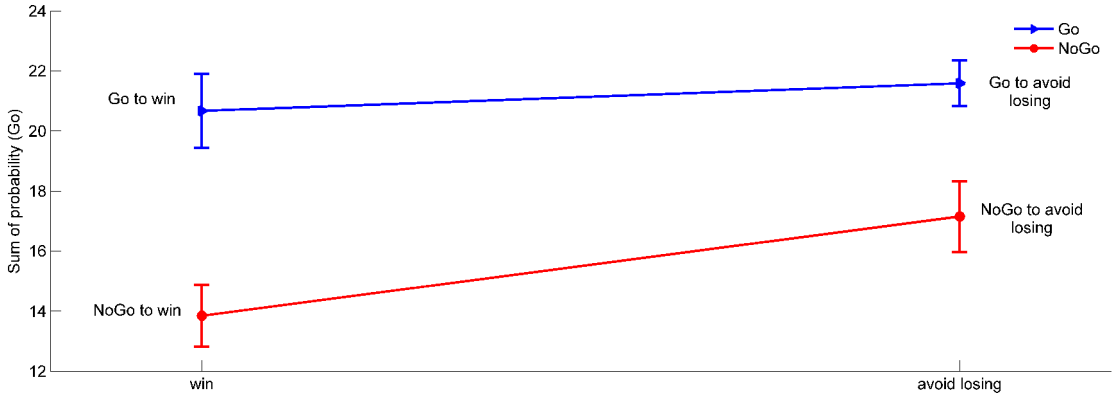

Figure 5.8: Action-valence interaction revealed in the $QL+Q_0$ model . The error bars depict the standard error of the mean (SEM).

In order to verify this hypothesis, we performed a two-way ANOVA for the probability of correct responses sampled from the fitting of QL+Q_0 and the AC+p_0 models to the raw data, with factors of action (2 levels: go and nogo) and valence (2 levels: win and avoid losing). This analysis showed a stronger action by valence interaction in the AC model ($F(1,23) = 19.595, p < 0.001$) than in the QL model ($F(1,23) = 4.590, p = 0.043$). Although, the Q-learning framework seems to better explain the

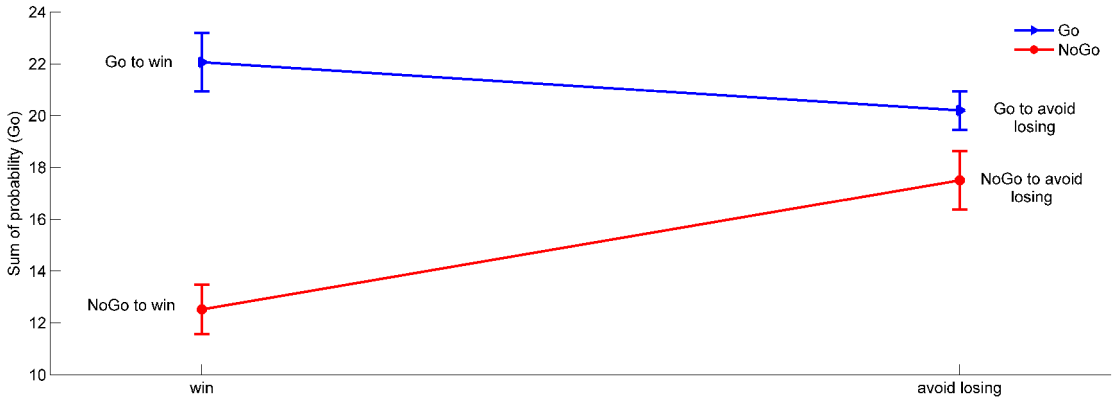

Figure 5.9: Action-valence interaction revealed in the $AC + p_0$ model. The error bars depict the standard error of the mean (SEM).

raw data, this analysis showed that the interaction action-valence was better captured by the actor-critic model.

5.2 Principal component analysis of behavioral data

The actor-critic and the Q-learning models differ in the sign of the PEs in the nogo to win and go to avoid losing conditions. For example, consider the nogo to win condition (respond: positive outcome, not respond: negative outcome): in a Q-learning framework, as subjects learned the action-values during the task, PEs become positive ($0 \leq Q(s, nogo) \leq 1, \delta = 1 - Q(s, nogo) > 0$) when they withhold their responses. On the other hand, if the subjects' performance is based on an actor-critic framework, the prediction errors are determined by the state-value function and the preferences are only updated when the subjects respond. Therefore, as subjects learned the state-values during the task, PEs became negative ($V(s) > 0, \delta = 0 - V(s) < 0$) every time they respond. In sum, the nogo to win condition is related to positive and negative PEs in the QL and AC models, respectively. Similarly, PEs differ in the go to avoid losing condition: the QL model embodies negative PEs, whereas the AC model embodies positive PEs. The dissimilarities in prediction errors according to each model are summarized in table 5.1.

	Go to win	NoGo to win	Go to avoid losing	NoGo to avoid losing
QL	$\delta \geq 0$	$\delta \geq 0$	$\delta \leq 0$	$\delta \leq 0$
AC	$\delta \geq 0$	$\delta \leq 0$	$\delta \geq 0$	$\delta \leq 0$

Table 5.1: Prediction errors underlying each condition in the Q-learning and actor-critic models.

As previously described in section 2.2.2, positive PEs are conveyed by dopamine bursts in the striatum, whereas negative PEs are conveyed by dopamine dips. The basal ganglia Go/NoGo model along with several findings in individuals with Parkinson's disease[1] [16, 41, 54] suggest that dopamine affects the subjects' learning performance, namely augmented DA enhances learning from positive PEs (DA bursts),

[1]Parkinson's disease is characterized by loss of dopaminergic neurons innervating the striatum in the basal ganglia.

but would impair learning from negative PEs (DA dips). Conversely, reduced DA would enhance learning from negative PEs, but would impair learning from positive PEs.

Therefore, we hypothesized that conditions might be associated with each other in terms of performance (correct or incorrect learning). In a Q-learning framework, the win conditions, which are determined by positive PEs (first and second entries of the top row in table 5.1) (DA bursts), would be positively correlated. Similarly, the avoid losing conditions, which are determined by negative PEs (third and fourth entries of the top row in table 5.1) (DA dips), would exhibit a positive correlation. Additionally, the win and the avoid losing conditions would be negatively correlated.

On the other hand, in an AC framework, the go conditions, which are governed by positive PEs (first and third entries of the bottom row in table 5.1) (DA bursts), would show a positive association. The performance in the nogo conditions, which are controlled by negative PEs (second and fourth entries of the bottom row in table 5.1) (DA dips) would also exhibit a positive association. Additionally, the go and nogo conditions would be negatively correlated.

Thereby, in order to investigate which model, QL or AC, is in line with the subjects' behavior, we tried to find associations among conditions in the behavioral raw data. This was achieved by performing a principal component analysis (PCA) on the subjects' performance in each condition. The subjects' performance was mathematically formulated as the fraction of correct responses in the third block. The data set was standardized such that each subject is centered to zero (A.1 in the appendix A). This analysis was performed in SPSS version 16.

Firstly, to verify the presence of correlations among conditions, we computed the Pearson's correlation coefficients, which measured the strength of the association among conditions. This preliminary analysis (in table 5.2) showed that only the go to avoid losing and the nogo to avoid losing conditions exhibited a positive strong and significant association ($r = 0.664, p = 0.001$)[2]. This finding was in line with the Q-learning framework. Although the correlations in the other pairs of conditions were weak or moderate and were not statistically significant, most of them exhibited a direction (positive or negative) in favour of the QL model. Namely, the coefficient between the go to win and nogo to win showed a positive direction and the coefficient between the go to win and the avoid losing conditions presented a negative direction. However, the coefficient between the nogo to win condition and the avoid losing conditions exhibited a positive direction, instead negative, as it would be expected.

The principal component analysis was performed on the the same dataset. The principal components are depicted in the figure 5.10. Before proceeding with the analyses of the results, we determined the more meaningful components.

In order to find the meaningful components, we combined the K1 method and the Cattel's scree test (see section subsection 3.7.2). According to the former, all the components whose eigenvalues were above 1 should be retained whereas the others should be discarded. The eigenvalues of each component are given in table 5.3. The first three components exhibited eigenvalues above 1 and, thus, they were considered significant. Additionally, the scree plot (figure 5.11) showed a significant last break at the third principal component, and, thus, according to the Cattel's scree test, only the first three components

[2]The strength of the Pearson correlation coefficient was categorized according to the criteria introduced in [6] as strong ($r = \pm.10$ to $\pm.29$), moderate ($r = \pm.30$ to $\pm.49$) and weak ($r = \pm.50 to \pm 1.0$).

		Go to win	Go to avoid losing	NoGo to win	NoGo to avoid losing	Neutral
Go to win	Pearson Correlation	1	-.156	.159	-.345	.174
	p-value		.468	.459	.098	.416
Go to avoid losing	Pearson Correlation	-.156	1	.151	.644	.006
	p-value	.468		.482	.001	.978
NoGo to win	Pearson Correlation	.159	.151	1	.375	-.189
	p-value	.459	.482		.071	.378
NoGo to avoid losing	Pearson Correlation	-.345	**.644**	.375	1	-.038
	p-value	.098	**.001**	.071		.860
Neutral	Pearson Correlation	.174	.006	-.189	-.038	1
	p-value	.416	.978	.378	.860	

Table 5.2: This table gives the Pearson's correlation coefficients among the 5 conditions. The p-value indicates how significantly the correlation coefficient is different from zero. The coefficients that are statistically significant are marked in blue.

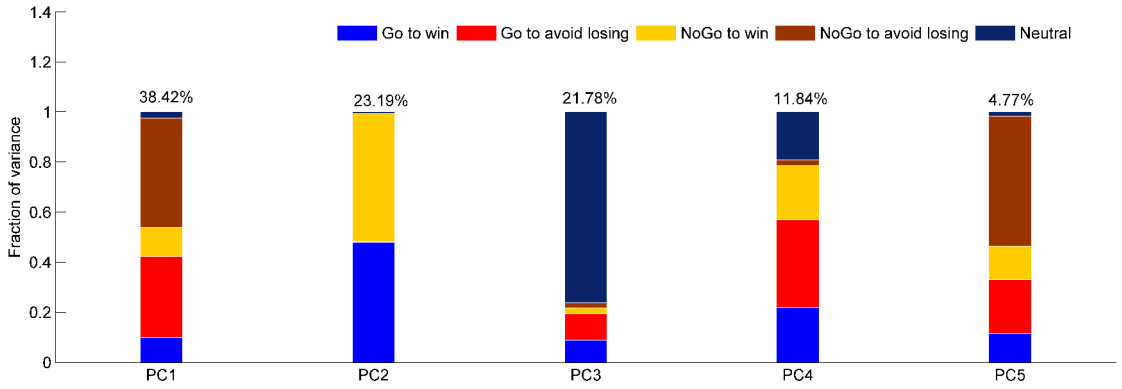

Figure 5.10: Results of the PCA analysis on the fraction of correct responses per subject and condition. Each set of stacked bars represents the fraction of the variance of a component explained by the original variables (go to win, go to avoid losing, nogo to win and nogo to avoid losing and neutral). Component 1: go to win (0.0919), go to avoid losing (0.3249), nogo to win (0.1155), nogo to avoid losing (0.4372), neutral (0.0233); Component 2: go to win (0.4797), go to avoid losing (0.0028), nogo to win (0.5098), nogo to avoid losing (0.0001), neutral (0.0076); Component 3: go to win (0.0888), go to avoid losing (0.1043), nogo to win (0.0245), nogo to avoid losing (0.0211), neutral (0.7612); Component 4: go to win (0.2191), go to avoid losing (0.3497), nogo to win (0.2175), nogo to avoid losing (0.0220), neutral (0.1916); Component 5: go to win (0.1133), go to avoid losing (0.2183), nogo to win (0.1328), nogo to avoid losing (0.5195), neutral (0.0162). The percentage on the top of each stacked bar corresponds to the contribution of each component to explain the total variance of the original data.

should be retained. This means that both tests agreed in retaining components 1,2 and 3. Our analysis, hence, only focused on these three principal components.

	PC1	PC2	PC3	PC4	PC5
Eigenvalue	1.9209	1.1595	1.0891	0.5919	0.2385

Table 5.3: Eigenvalues of each principal component.

The results depicted in figure 5.10 revealed that the first component was mostly explained by the go to avoid losing and nogo to avoid losing conditions (76.21%), the second component was almost totally explained by the go to win and nogo to win conditions (98.95%) and finally the third component only captured the variance imposed by the neutral condition (76.12%). These findings clearly showed that win, avoid losing and neutral conditions are orthogonal to each other. A geometrical visualization of this evidence is depicted in figure 5.12. This finding suggested that the win, avoid losing and neutral conditions have different driving forces. The principal components grouped together the conditions which

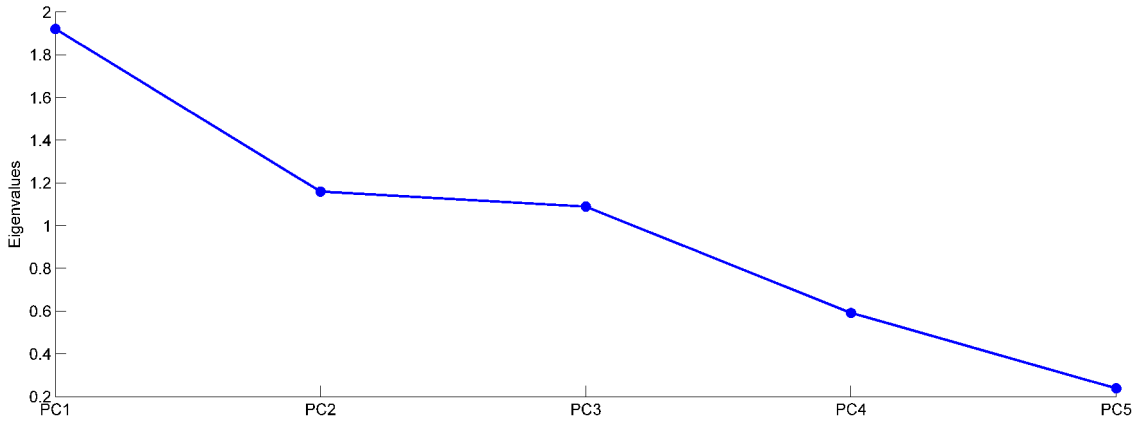

Figure 5.11: Scree plot.

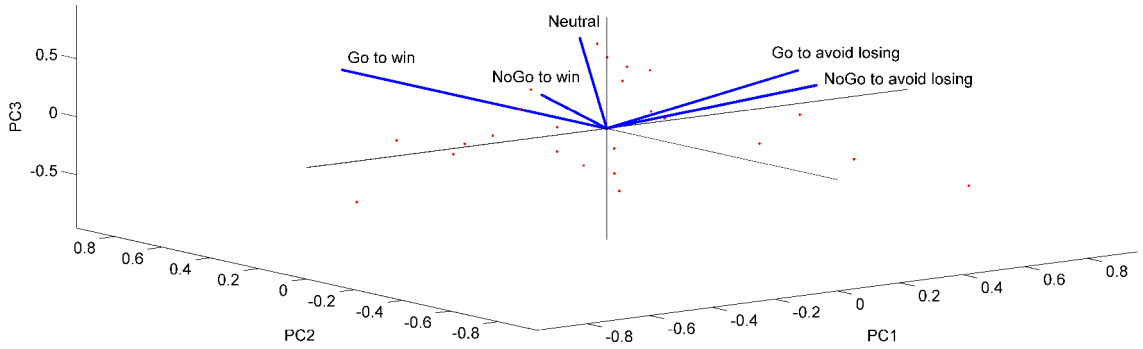

Figure 5.12: The red dots are the fraction of correct responses represented in the 3-dimensional space spanned by the first and second principal components, PC1,PC2 and PC3, respectively. The blue lines depict each condition in the new coordinate system.

share the same sign of the PEs in the Q-learning framework: the win conditions are linked to positive PEs, the avoid losing conditions are connected to negative PEs and the neutral are connected to zero PEs. Therefore, the driving forces could be related to the sign of the PEs.

Additionally, the QL framework would require that the win conditions were positively associated, as well as the avoid losing conditions. Indeed, the eigenvectors (in table 5.4) showed a positive trend in the win and avoid losing conditions. The first principal component (mostly explained by the avoid losing conditions) was geometrically described by an eigenvector whose avoid losing components (second and fourth entries of the top row in table 5.4) had the same sign, and thus, are positively correlated. Similarly, the second principal component (mostly explained by the win conditions) was associated with an eigenvector whose win components showed the same sign, and thus, are positively correlated. Regarding the third component, it only captured the variation of the neutral condition and thus did not exhibit any relevant relationship between other conditions.

According to what was explained above about the relationship between dopamine and prediction errors, it would be plausible to expect that the avoid losing and win conditions shared the same driving

	Go to win	Go to avoid losing	NoGo to win	NoGo to avoid losing	Neutral
PC1	0.3147	-0.5700	-0.3398	-0.6612	0.1528
PC2	0.6926	-0.0527	0.7140	-0.0120	-0.0871
PC3	0.2980	0.3230	-0.1564	0.1454	0.8725

Table 5.4: Eigenvectors of each principal component.

force, i.e. level of dopamine in the striatum. Thereby, they should be part of the same principal component with a negative association between them. However, they are clearly explained by two independent components.

As explained in section 2.2.2, there was evidence which suggested that negative prediction errors might be conveyed by the dip duration instead of its magnitude. This could be the reason why win and avoid losing conditions showed a different driving force, and, thus, belonged to orthogonal components.

The neutral condition measures the natural tendency of subjects to respond without undergoing any learning process. There were evidences that this natural tendency to respond depended on tonic dopamine instead on the phasic dopamine. Tonic dopamine corresponds to the basal level of dopamine in the striatum, and thus, when it is increased it is likely to activate the direct pathway, leading to a higher tendency to respond, whereas when it is decreased it activates the indirect pathway leading to a lower tendency to respond. It is still unknown, if there is a relationship between phasic and tonic dopamine. However, these results suggest that they are uncorrelated.

The PCA showed an association between go to win and nogo to win conditions (second component), but the Pearson's correlation analysis exhibited a weak and non-significant correlation ($r = .159, p = .459$). Both analyses assumed that all the subjects performed solely according to one policy. As described in section 2.3.4, subjects could exhibit a more Pavlovian or more instrumental policy. Pavlovian policy linked outcomes to valence-dependent stereotyped behavioral. In other words, regardless of the action validity, Pavlovian responses associated with reward entailed vigor whilst responses linked to punishments were associated with action inhibition. On the other hand, subjects exhibiting a more instrumental policy learned solely based on contingent consequences. Thereby, we should have accounted for this difference in policy, in order to capture correctly the associations among conditions.

To identify the presence of these sub-groups in the overall population, we carried out a Gaussian mixture model on the 3-dimensional space spanned by the first, second and third principal components. We were interested in finding two sub-populations, and thus, we applied this method by defining the number of clusters as two. This analysis was performed in MATLAB version 2012b using a function provided by the toolbox SPM version 8.

The Gaussian mixture method found two sub-populations which were mainly determined by the first principal component (in the first and second charts in the figure 5.13, from left to right, the two clusters are located either in the positive or in negative side of the PC1 axis.). In order to identify to which sub-population each data point belonged, we compared the responsibilities of each subject in each sub-group. The responsibilities are given in table 5.5.

The learning time courses of each sub-population were determined so that we could have a clearer

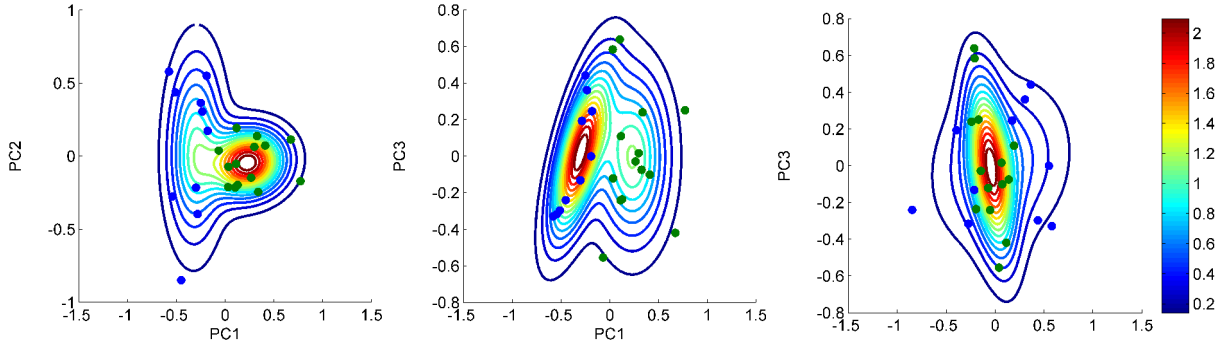

Figure 5.13: Projection of the probability distribution of observations in the overall population onto the spaces spanned by each pair of the principal components. The probability density lays on a 3-dimensional space spanned by the first three principal components. Hence, to visualize its shape we should be able to plot it in a 4-dimensional space, which is geometrically impossible. Therefore, we projected the probability distribution onto the 2-dimensional space spanned by each set of two components. To achieve this, we integrated out a component from the distribution assuming that they are statistically independent. Since the covariances were very low, the error induced by this assumption was also small. The blue data points represent the data with higher probability of belonging to sub-population 1 and the green data points represent the data with higher probability of belonging to sub-population 2.

	1	2
1	0	**1**
2	**1**	0
3	**1**	0
4	0.1090	**0.8910**
5	**1**	0
6	**0.9984**	0.0016
7	**1**	0
8	0.0055	**0.9945**
9	0.0002	**0.9998**
10	0	**1**
11	0	**1**
12	0.0004	**0.9996**
13	**0.9682**	0.0318
14	**0.9212**	0.0788
15	0.0001	**0.9999**
16	**0.9765**	0.0235
17	0.1892	**0.8108**
18	0.0692	**0.9308**
19	0	**1**
20	**0.9997**	0.0003
21	**1**	0
22	0	**1**
23	0	**1**
24	0.0001	**0.9999**

Table 5.5: Probability of belonging to the sub-population 1 and 2 of each subject. The subjects marked in blue belong to sub-population 2 and the subject marked in green belong to sub-population 1.

idea of what was the source of this clustering (figure 5.16).

Subjects from the sub-population 1 (figure 5.15) learned to respond in the nogo to win condition and performed worse in the go to avoid losing condition. These findings could suggest that the subjects from sub-population 1 followed a Pavlovian policy. However, they performed worse in the nogo to avoid losing

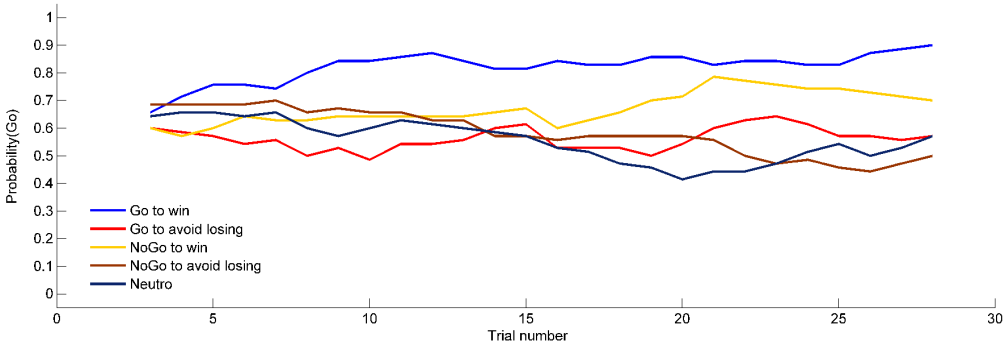

Figure 5.14: sub-population 1

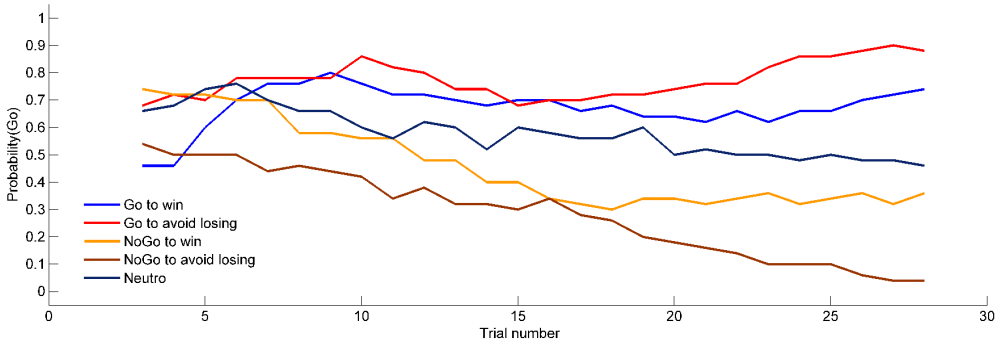

Figure 5.15: sub-population 2

Figure 5.16: Average time varying probability, across subjects, of making the go action for the five conditions convolved with a central moving average filter with length 5.

condition compared to the sub-population 2, which was not in accordance with a Pavlovian behavior. In order to further investigate whether the behavioral differences between the two groups were instrumental or Pavlovian, we used the parametric estimates of individual fits of the QL+*pav* model. A complication with using these model fits to compare groups was that the parametric estimates were not independent but instead tended to covary over subjects. In particular, because the Pavlovian parameter (π) was multiplied by the softmax inverse of the temperature to compute action probability, these two parameters tended to be inversely coupled. Therefore, the parameters viewed separately could have improbable means and large estimation errors [10]. However, their product $\beta \times \pi$ tends to be more reliable estimate. $\beta \times \pi$ can be viewed as the impact of the Pavlovian policy in the subjects' behavior. Since this parameter was normally distributed in both sub-populations (Kolmogorov-Smirnov test, $p = 0.3730$, $p = 0.1852$ for sub-population 2 and 1, respectively), the comparison was performed according to a two-sample test. The test revealed that there was no significant difference between both groups ($p = 0.3321$). This finding suggested that the difference between sub-population 1 and sub-population 2 might not came from the difference in policy. A second hypothesis was that sub-population 1 gathered all the subjects that learned worse in all conditions.

Although, we could not find the source of such a difference, it was clear that the nogo to win condition was inverted in the sub-population 1 compared to the sub-population 2 (yellow lines in figure 5.16). This

inversion could be the reason why win conditions did not exhibit a significant positive Pearson's correlation in our first analysis. In order to test this hypothesis, we re-computed the Pearson's correlation coefficients, separately, for each sub-population.

		Go to win	Go to avoid losing	NoGo to win	NoGo to avoid losing	Neutral
Go to win	Pearson correlation	1	0	.947	-.180	.072
	p-value		1	<.001	.619	.843
Go to avoid losing	Pearson correlation	0	1	.060	.133	-.610
	p-value	1		.869	.713	.061
NoGo to win	Pearson correlation	**.947**	.060	1	-.274	.008
	p-value	**<.001**	.869		.443	.983
NoGo to avoid losing	Pearson correlation	-.180	.133	-.274	1	-.425
	p-value	.619	.713	.443		.221
Neutral	Pearson correlation	.072	-.610	.008	-.425	1
	p-value	.843	.061	.983	.221	

Table 5.6: Pearson's correlation coefficients between the 5 conditions in the sub-population 2. The p-value indicates how significantly the correlation coefficient is different from zero. The coefficients whose p-value is below the significance level of .05, and thus are statistically significant, are highlighted in blue.

		Go to win	Go to avoid losing	NoGo to win	NoGo to avoid losing	Neutral
Go to win	Pearson correlation	1	.158	.-578	-.204	.377
	p-value		.590	.030	.483	.184
Go to avoid losing	Pearson correlation	.158	1	-.492	.438	.373
	p-value	.590		.074	.117	.190
NoGo to win	Pearson correlation	**-.578**	-.492	1	-.034	-.398
	p-value	**.030**	.074		.909	.159
NoGo to avoid losing	Pearson correlation	-.204	.438	-.034	1	.179
	p-value	.483	.117	.909		.541
Neutral	Pearson correlation	.377	.373	-.398	.179	1
	p-value	.184	.190	.159	.541	

Table 5.7: Pearson's correlation coefficients between the 5 conditions in the sub-population 1. The p-value indicates how significantly the correlation coefficient is different from zero. The coefficients whose p-value is below the significance level of .05, and thus are statistically significant, are highlighted in blue.

The results showed a strong and statistically significant correlation between the win conditions in both sub-populations. These associations were positive and negative in the sub-populations 2 and 1, respectively. This finding clearly suggests that there is a correlation between the win conditions that was likely to be masked by the abrupt inversion of the behavior in the nogo to win condition. The positive direction of the coefficient in sub-population 2 matched expectations, however the negative direction exhibited by sub-population 1 did not. Although, the results were not in in line with a change of policy as the cause of the differences between the sub-populations, it was clear that subjects changed their behavior in the nogo to win condition, which might explain that the correlation between the win conditions is negative in the sub-population 1.

In sum, the results obtained suggested that the win and avoid losing conditions are governed by an independent driving force. This evidence could be justified by the different dopamine mechanisms behind the positive and negative prediction errors. Namely, positive prediction errors would be conveyed by the dopamine burst magnitude whereas negative prediction errors would be governed by the duration of the dopamine dip.

These results also strongly supported the idea of an independent driving force between the neutral and the rest of the conditions. The neutral condition is related to the tonic dopamine in striatum and the

other conditions are associated with the phasic dopamine. This suggests that they might be independent.

Finally, these results were clearly in favour of the Q-learning framework which make us believe that subjects' learning is mechanistically more similar to the Q-learning comparing to the actor-critic. Particularly, we found positive strong correlations between the win and the avoid conditions, and the PCA analysis grouped together the conditions which had in common the prediction errors' sign in the QL framework. Indeed, our findings are in accordance with previous studies in animals which indicated that prediction errors were determined by the action-value rather than the state-value.

6

Conclusions and Future Work

Contents

6.1 Conclusions

This study aimed to determine which reinforcement learning framework explained better decision-making in healthy subjects: Q-learning or actor-critic. Particularly, whether the prediction errors used to update old predictions were determined by the value of the action or by the value of the state. This was achieved by conducting a modified probabilistic Go/NoGo task which orthogonalizes action and valence in 24 healthy subjects.

Before proceeding to the analysis which could answer our main question, we started by verifying if the subjects have correctly learned the task. The neutral condition measures the natural tendency to respond and thus should be used as a baseline behavior when analysing the subjects' performance. Since a neutral condition has never been used in this type of task, we wish to analyse its impact on the behavioral analysis. This was achieved by performing an analysis with and without taking into account the natural tendency to respond.

Our findings suggested that subjects had correctly learned the go to win and nogo to avoid losing conditions, but not the nogo to win and go to avoid losing conditions. However, this might be interpreted as an effect of the strong action by valence interaction rather than bad learning. This interaction showed that there is a coupling between reward and go choices and punishment and nogo choices. In other words, subjects learn better when they have to respond to receive a reward and when they have to not respond to avoid a punishment. This is in line with previous findings which suggested that the learning process was not purely instrumental but it was also affected by a Pavlovian mechanism. We can thus conclude that subjects seemed to have learned worse the nogo to win (compared to the nogo to avoid losing condition) and the go to avoid losing conditions (compared to the go to win condition). However, the performance is not disrupted in the same manner in both conditions. In the go to avoid losing condition, subjects tend to perform correctly but they are slower to achieve the correct policy. On the other hand, subjects do not really learn to not respond in the nogo to win condition.

When taking into consideration the tendency to respond, the results were similar. However, they indicated that subjects did not learn correctly the nogo to avoid losing condition, but the go to avoid losing condition was effectively learned. Despite this result, the nogo to avoid losing exhibited a positive trend which supports the hypothesis that subjects might have learned.

Therefore, the analysis with the neutral condition was able to detect a correct performance in the go to avoid losing condition which was masked by the high tendency to respond at the beginning of the task. The baseline behavior play thus an important role in order to extract the precise conclusions from data. After this first analysis on the behavioral data, we turned to our main goal.

Firstly, several Q-learning and actor-critic models were fitted to the subjects' behavioral data and then compared in order to select the best model. This analysis suggested that the best model was based on a QL framework and it took into account an initial bias towards responding which was naturally erased as the subjects learned. However, according to the learning time courses, this model slightly failed to capture the action by valence interaction. Adding the Pavlovian factor to the model was useless. This could be explained by the tendency of the BIC to overpenalize the model accuracy by increasing the

model complexity.

Despite of explaining better the subjects' behavior, the QL was not able to naturally capture the action by valence interaction as well as the actor-critic model. This was in line with our initial expectations that hypothesized that the learning process tended to be naturally slower in the nogo to win and go to avoid losing conditions when based on an AC framework.

These findings clearly suggested that the subjects' choices came out from a computational mechanism more similar to the QL than to the actor-critic. In order to further test this hypothesis, we applied a principal component analysis to determine associations among conditions.

According to the sign of the prediction errors (positive or negative), the Q-learning and actor-critic exhibited different associations among conditions. Hence, finding the relationships among conditions accommodated in the raw data could corroborate the results obtained previously in the model fitting approach.

The PCA revealed two meaningful components which grouped together the go to win and nogo to win conditions, go to avoid losing and nogo to avoid losing conditions and it also exhibited a third meaningful component which was mainly determined by the neutral condition. Furthermore, both win and avoid losing associations showed a positive direction. Thereby, the principal components grouped together the conditions which share the same sign of the PEs in the Q-learning framework: the win conditions are linked to positive PEs, the avoid losing conditions are determined by negative PEs and the neutral condition are connected to zero PEs.

PEs are coded by phasic DA, and thus, ultimately, DA is the main driving force of the win and the avoid conditions. Particularly, positive PEs are conveyed by dopamine bursts in the striatum, whereas negative PEs are conveyed by dopamine dips. The basal ganglia Go/NoGo model along with several findings in individuals with Parkinson's disease [16, 41, 54] suggested that dopamine affects the subjects' learning performance, namely augmented DA enhances learning from positive PEs (DA bursts), but would impair learning from negative PEs (DA dips). Conversely, reduced DA would enhance learning from negative PEs, but would impair learning from positive PEs.

According to this relationship between DA and PEs, it would be plausible to expect that the avoid losing and win conditions shared the same driving force, i.e. level of dopamine in the striatum. Thereby, they should be grouped together in the same principal component with a negative association between them. However, they were grouped in two independent components.

These findings do not make sense if we assume that dips of dopamine are coded by its magnitude. However, there was evidence which suggested that negative prediction errors might be conveyed by the dip duration instead of its magnitude. This might explain why win and avoid losing conditions showed different driving forces, and, thus, belonged to orthogonal components.

These results also strongly support the idea of an independent driving force between the neutral and the rest of the conditions. Since the neutral condition is related to the tonic dopamine in striatum whereas the other conditions are associated with the phasic dopamine, this suggests that they might be independent.

Although these findings are clearly in favour of the QL framework, a preliminary Pearson's correlation

analysis showed a a non-significant positive correlation between the go to win and nogo to win conditions. A clustering method, namely Gaussian mixture models, revealed the presence of two sub-groups which clearly differed in the subjects' performance in the nogo to win condition. Although, a posterior analysis showed that the sub-groups did not differ in the policy followed by the subjects, Pavlovian or instrumental, it was clear that in sub-group 1 subjects learned to respond whereas in sub-group 2 they learned to not respond. This fact could be masking the true correlation between the win conditions. Indeed, the sub-group where subjects have correctly learned to respond exhibited a strong and positive correlation.

In sum, our findings suggests that healthy humans use the action-values (QL) instead of the state-values (AC) to determine the prediction errors when learning to make a choice. This conclusion corroborates electrophysiological findings in animals, which demonstrated that prediction errors were determined by the action value.

6.2 Future work

Although our findings and previous studies are in favour of the QL model, neuroanatomical findings support the AC framework. Namely, the dorsolateral striatum is associated with the actor and the ventral striatum is related to the critic. Therefore, determining whether the learning rules of the QL can be incorporated within the AC framework in the basal ganglia will necessitate further experiments and computational investigation.

There are different versions of calculating the PE, associated with different temporal difference algorithms. In the actor-critic framework, the PEs are determined by the action-value. In the other two classes of algorithms, Q-learning and SARSA, PEs are determined by the action-value. However, in the QL the PEs are determined by the action-value of the better option rather than the one actually chosen. On the other hand, in SARSA algorithms, the PEs use the action-value of the chosen option. In this work, we were only concerned about distinguishing whether the PEs are governed by the state-value or by the action-value. Therefore, in a future study, it might be worth to also take into consideration the difference between the QL and SARSA models. In order to achieve this, one has to use a task whose transition probabilities between stimuli are dependent on the action.

This paradigm was designed to investigate decision-making in humans in a fully orthogonalize action and valence context. Despite standing out some differences between the Q-learning and the actor-critic, in the future, paradigms should be used specially designed to identify the best model by taking more advantage of their differences. Moreover, functional magnetic resonance imaging could play an important role in finding the true mechanism, by providing a measure of signals that are more directly related to the mechanism than the behavioral data.

Additionally, our approach just took into consideration habit learning by neglecting the other end of the instrumental learning spectrum, goal-directed behavior. However, behavioral phenomena have already demonstrated the insufficiency of temporal difference-like mechanisms alone, making it necessary to use model-free Reinforcement learning models which provide a quantitative framework for the goal-directed behavior.

Although it is not possible to perform invasive electrophysiological recordings in humans as it is

possible in animals, other indirect measurements, such as BOLD signal, could be used to corroborate the behavioral findings. For example, one could perform an fMRI model-based analysis to determine which PE signal explained better the BOLD activity in striatum.

Finally, we would like to note that BIC, albeit simple, is not the best criterion for model comparison. It can bee seen that BIC approximates the complexity with the number of parameters scaled by the log of the number of observations. However, this does not account for the true model complexity. For example, true complexity should not change when we add a parameter whose effect is identical to another parameter, however BIC would indicate that the model complexity has increased. Thereby, BIC overpenalizes a model when a parameter is added. A criterion which take this redundancy into account should be used in future work, such as the negative free-energy [47].

Bibliography

[1] Albin, R., Young, A., and Penney, J. (1989). The functional anatomy of basal ganglia disorders. *Trends Neuroscience*, 12:366–375.

[2] Attias, H. (2000). A variational bayesian framework for graphical models. In *In Advances in Neural Information Processing Systems 12*, pages 209–215. MIT Press.

[3] Barto, A. G. (1994). *Models of information processing in the basal ganglia*, chapter Adaptive critic and the basal ganglia. MIT Press.

[4] Beal, M. J. (2003). *Variational algorithms for approximate bayesian inference*. PhD thesis, University College London.

[5] Cattell, R. B. (1966). The scree test for the number of factors. *Multivariate Behavioral Research*, 1:245–276.

[6] Cohen, J. (1988). *Statistical Power Analysis for the Behavioral Sciences*. Lawrence Erlbaum Associates.

[7] Crockett, M. J., Clark, L., and Robbins, T. W. (2009). Reconciling the role of serotonin in behavioral inhibition and aversion: acute tryptophan depletion abolishes punishment-induced inhibition in humans. *The Journal of Neuroscience*, 29:11993–11999.

[8] Daw, N. D. and Doya, K. (2006). The computational neurobiology of learning and reward. *Current opinion in Neurobiology*, 16:199–204.

[9] Daw, N. D., Niv, Y., and Dayan, P. (2005). *Recent breakthroughs in basal ganglia research*, chapter Actions, Policies, Values, and the Basal Ganglia. Nova Science Publishers.

[10] Delgado, M., Phelps, E., and Robbins, T. (2011). *Decision Making, Affect, and Learning: Attention and Performance XXIII*. Oxford University Press.

[11] Domjan, M. (2010). *The principles of learning and behavior*. Wadsworth, Cengage Learning, 6 edition.

[12] Doya, K. (2007). Reinforcement learning: Computational theory and biological mechanisms. *HFSP Journal*, 1:30–40.

[13] Frabrigar, L. R., Wegener, D. T., MacCallum, R. C., and Strahan, E. J. (1999). Evaluating the use of exploratory factor analysis in psychological research. *Psychological methods*, 4:272–299.

[14] Fraley, C. and Raftery, A. E. (1998). How many clusters? which clustering method? answers via mode-based cluster analysis. *The Computer Journal*, 41:578–588.

[15] Frank, M. J. (2005). Dynamic dopamine modulation in the basal ganglia: A neurocomputational account of cognitive deficits in medicated and nonmedicated parkinsonism. *Journal of Cognitive Neuroscience*, 17:51–72.

[16] Frank, M. J., Seeberger, L. C., and O'Reilly, R. C. (2004). By carrot or by stick: Cognitive reinforcement learning in parkinsonism. *Science*, 306:1940–1943.

[17] Gerfen, C. R. (2000). Molecular effects of dopamine on striatal-projection pathways. *Trends Neuroscience*, 23:64–70.

[18] Gray, J. A. and McNaughton, N. (2000). *The Neuropsychology of Anxiety: An Inquiry into the Function of the Septo-hippocampal System*. Oxford University Press.

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

[20] Hirosaka, O., Nakamura, K., and Nakahara, H. (2006). Basal ganglia orient eyes to reward. *Journal of Neuroscience*, 95:567–584.

[21] Kaiser, H. F. (1960). The application of electronic computers to factor analysis. *Educational and Psychological Measurement*, 20:141–151.

[22] Kass, R. E. and Raftery, A. E. (1995). Bayes factors. *Journal of American Statistical Association*, 90:773–795.

[23] Kawagoe, R., Takikawa, Y., and Hikosaka, O. (1998). Expectation of reward modulates cognitive signals in the basal ganglia. *Nature Neuroscience*, 1:411–416.

[24] Kawagoe, R., Takikawa, Y., and Hikosaka, O. (2004). Reward-predicting activity of dopamine and caudate neurons-a possible mechanism of motivational control od saccadic eye movement. *journal of Neurophysiology*, 91:1013–1024.

[25] Ledesma, R. D. and Valero-Mora, P. (2007). Determining the number of factors to retain in efa: an easy-to-use computer program for carrying out parallel analysis. *Practical Assesment & evaluation*, 12:1–11.

[26] Maia, T. V. (2009). Reinforcement learning, conditioning, and the brain: Successes and challenges. *Cognitive, Affective, Behavioral Neuroscience*, 9:343–364.

[27] Maia, T. V. and Frank, M. J. (2011). From reinforcement learning models to psychiatric and neurological disorders. *Nature Neuroscience*, 14:154–162.

[28] Mink, J. W. (1996). The basal ganglia: focused selection and inhibition of competing motor programs. *Progress in Neurobiology*, 50:381–425.

[29] Montague, P. R., Dayan, P., Person, C., and Sejnowski, T. J. (1995). Bee foraging in uncertain environments using predictive hebbian learning. *Nature*, 377:725–728.

[30] Montague, P. R., Dayan, P., and Sejnowski, T. J. (1996). A framework for mesencephalic dopamine systems based on predictive hebbian learning. *The Journal of Neuroscience*, 16:1936–1947.

[31] Morris, G., Nevet, A., Arkadir, D., Vaadia, E., and Bergman, H. (2006). Midbrain dopamine neurons encode decisions for future action. *Nature Neuroscience*, 9:1057–1063.

[32] Nambu, A., Tokuno, H., and Takada, M. (2002). Functional significance of the cortico-subthalamo-pallidal 'hyperdirect' pathway. *Neuroscience Research*.

[33] Niv, Y. and Montague, R. (2009). *Neuroeconomics: Decision making and the brain*, chapter 22. Academic Press, 1 edition.

[34] O'Doherty, J. P., Dayan, P., Friston, K., Critchley, H., and Dolan, R. J. (2003). Temporal difference models and reward-related learning in the human brain. *Neuron*, 28.

[35] Rammery, G. A. (1994). On-line q-learning using connectionist systems. Technical report, Cambridge University Engineering Department.

[36] Reynolds, J. N. J., Hyland, B. I., and Wickens, J. R. (2001). A cellular mechanism of reward-related learning. *Nature*, 413:67–70.

[37] Reynolds, J. N. J. and Wickens, J. R. (2000). Substantia nigra dopmaine regulates synaptic plasticity and membrane potential fluctuations in the rat neostriatum, *in vivo*. *Neuroscience*, 99:199–203.

[38] Reynolds, J. N. J. and Wickens, J. R. (2002). Dopamine-dependent plasticity of corticostriatal synapses. *Neural Networks*, 15:507–521.

[39] Ringnér, M. (2008). What is principal component analysis? *Nature Biothechnology*, 26:303–304.

[40] Roesch, M. R., Calu, D. J., and Schoenbaum, G. (2007). Dopamine neurons encode the better option in rats deciding between differently delayed or sized rewards. *Nature Neuroscience*, 10:1615–1624.

[41] Rutledge, R. B., Lazzaro, S. C., Lau, B., Myers, C. E., Gluck, M. A., and Glimcher, P. W. (2009). Dopaminergic drugs modulate learning rates and perseveration in parkinson's patients in a dynamic foraging task. *The Journal of Neuroscience*, 29:15104–15114.

[42] Samejima, K., Ueda, Y., Doya, K., and Kimura, M. (2005). Representation of action-specific reward values in the striatum. *Science*, 310:1337–1340.

[43] Schultz, W., Dayan, P., and Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, 275:1593–1599.

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

[45] Shen, W., Flajolet, M., Greengard, P., and Surmeier, D. J. (2008). Dichotomous dopaminergic control of striatal synaptic plasticity. *Science*.

[46] Skinner, B. F. (1935). Two types of conditioned reflex and pseudo type. *Journal of General Psychology*, 12:66–77.

[47] Stephan, K. E., Penny, W. D., Daunizeau, J., Moran, R. J., and Friston, K. J. (2009). Bayesian model selection for group studies. *NeuroImage*, 46:1004–1017.

[48] Sutton, R. S. and Barto, A. G. (1998). *Reinforcement Learning: an Introduction*. MIT Press.

[49] Tanaka, S., Doya, K., Okada, G., Ueda, K., Okamoto, Y., and Yamawaki, S. (2004). Prediction of immediate and future rewards differentially recruits cortico-basal ganglia loops. *Nature neuroscience*, 7:887–893.

[50] Thorndike, E. L. (1911). *Animal intelligence: experimental studies*. The Macmillan Company.

[51] Watkins, C. J. C. H. (1989). *Learning from delayed rewards*. PhD thesis, University of Cambridge.

[52] Wickens, J. and Kötter, R. (1994). *Models of information processing in the basal ganglia*, chapter Cellular models of reinforcement, pages 187–214. MIT Press.

[53] Wickens, J. R., Begg, A. J., and Arbuthnott, G. W. (1996). Dopamine reverses the depression of rat corticostriatal synapses which normally follows high-frequency stimulation of cortex *In vitro*. *Neuroscience*, 70:1–5.

[54] Wiecki, T. V. and Frank, M. J. (2010). Neurocomputational models of motor and cognitive deficits in parkinson's disease. *Progress in Brain Research*, 183:275–297.

[55] Yerkes, R. M. and Morgulis, S. (1909). The method of pawlow in animal psychology. *Psychological Bulletin*, 6:257–273.

A

Tables and figures

	Go to win	Go to avoid losing	NoGo to win	NoGo to avoid losing	Neutral
1	0.8	0.6	0.2	0.5	0.5
2	1	0.9	0.8	0.9	0.9
3	1	1	1	1	0
4	1	0.9	0	0.8	1
5	1	0.8	1	0.7	0.5
6	0.3	0.9	0.5	1	0.2
7	1	1	0.8	1	0
8	0.6	0.6	0.5	0.6	0.5
9	0.7	0.5	0.2	0.7	0.3
10	0.8	0.3	0.5	0.6	0.7
11	1	0.6	0	0	0.8
12	0.6	0.7	0.6	0.5	0
13	0.3	0.9	0.4	0.8	0.8
14	0.5	0.7	0.4	1	0.4
15	1	0.7	0.1	0.7	0.1
16	0.8	0.7	0.7	0.9	0.8
17	1	1	0	0.8	0.9
18	0.9	0.5	0.6	0.7	0.7
19	1	0.6	0.3	0.3	0.4
20	0.9	0.7	0.8	1	0.9
21	0	0.9	0	1	0.3
22	1	0.2	0.3	0.2	0.2
23	0.9	0.6	0.5	0.3	0.5
24	0.9	0.7	0	0.5	0.7

Table A.1: Fraction of correct responses in the third block in each condition for each subject.

	Block 1	Block 2	Block 3
Action (Go/NoGo)	$t(23) = -0.076$ $p = 0.940$	$t(23) = 0.814$ $p = 0.424$	$t(23) = 1.444$ $p = 0.162$

Table A.2: Post hoc paired t-test on the number of rectified correct responses between go and nogo conditions for each block.

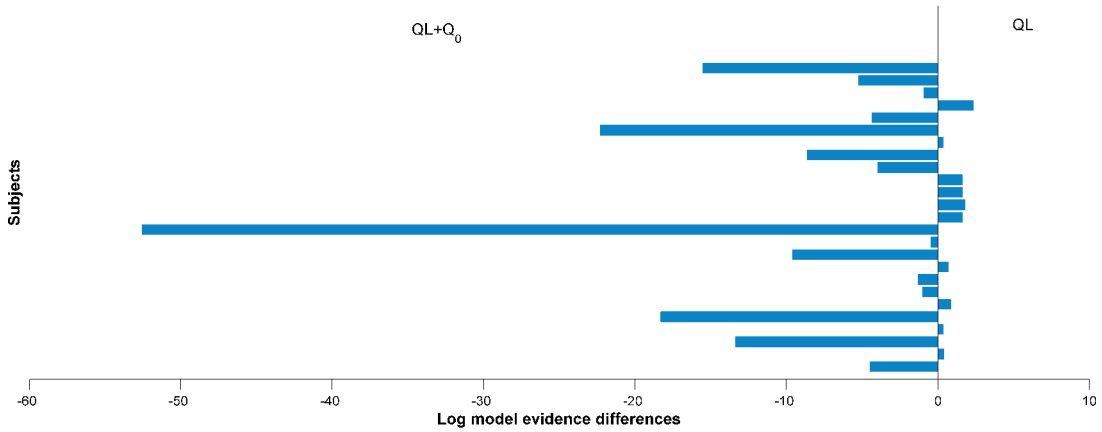

Figure A.1: Two nested models were fitted to the behavioural data and compared: (1) standard QL model with two parameters: the learning rate and the inverse of temperature; (2) equal to the model 1 with Q_0 added. The bar chart shows the difference in log-evidences for all twenty-four subjects.

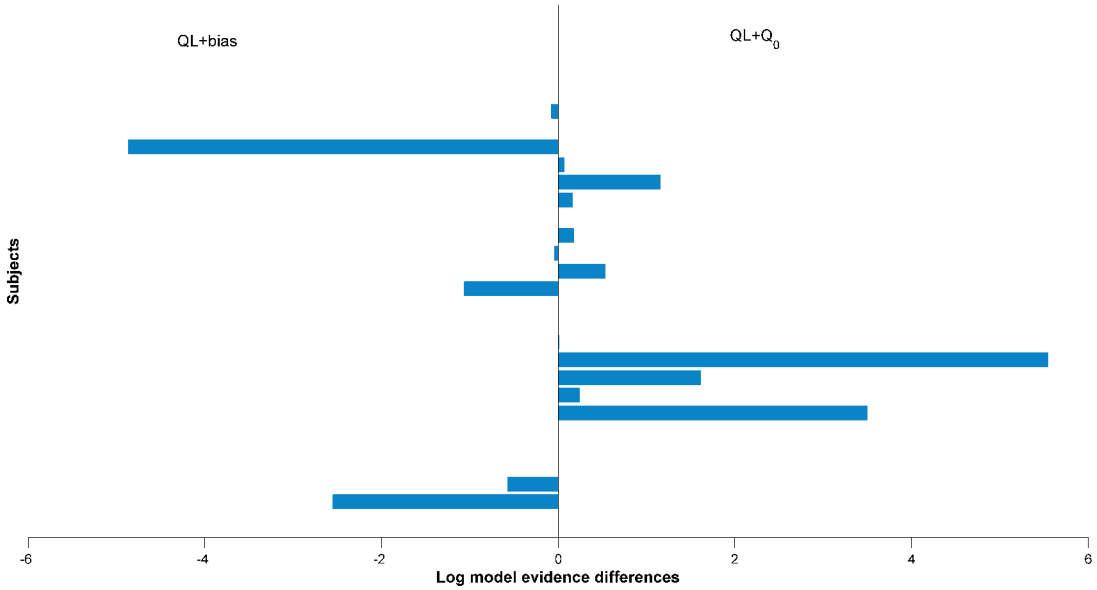

Figure A.2: Two variants of the standard QL model were fitted and compared: (1) with a bias parameter; (2) with Q_0 parameter. The bar chart shows the difference in log-evidences for all twenty-four subjects.

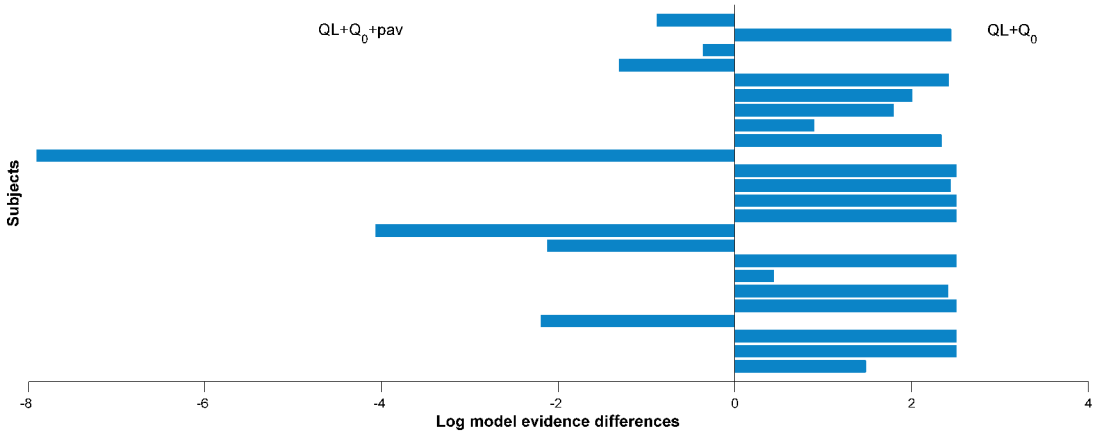

Figure A.3: Two variants of the standard Q-learning model were fitted and compared: (1) with a Q_0 parameter; (2) with Pavlovian parameter ($\pi V(s_t)$). The bar chart shows the difference in log-evidences for all twenty-four subjects.

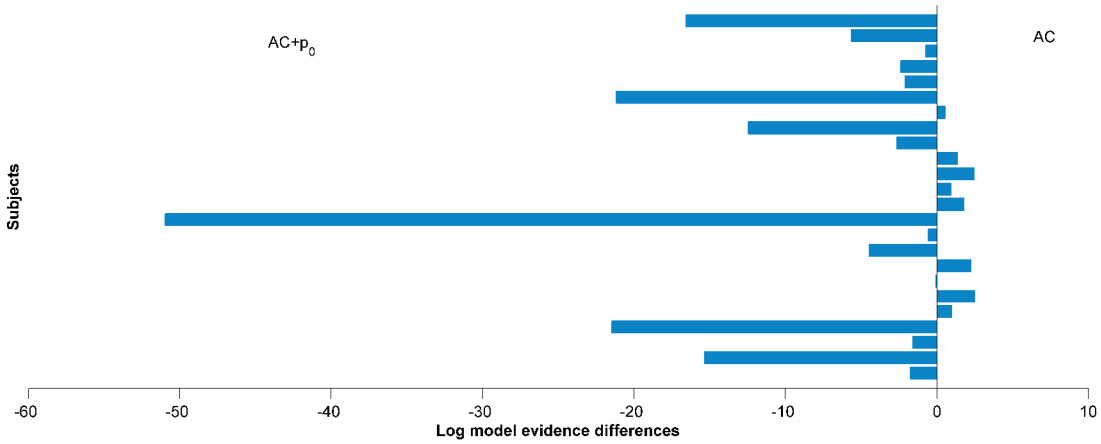

Figure A.4: Two variants of the AC models were fitted and compared: (1) AC; (2) AC+pav. The horizontal bars show the difference in log-evidences for all twenty-four subjects.

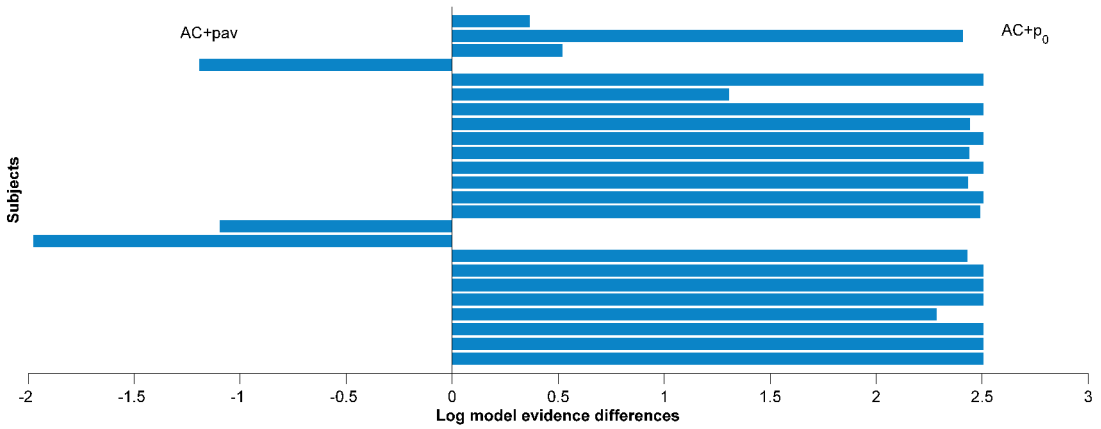

Figure A.5: Two variants of the AC models were fitted and compared: (1) AC+p_0; (2) AC+pav. The horizontal bars show the difference in log-evidences for all twenty-four subjects.

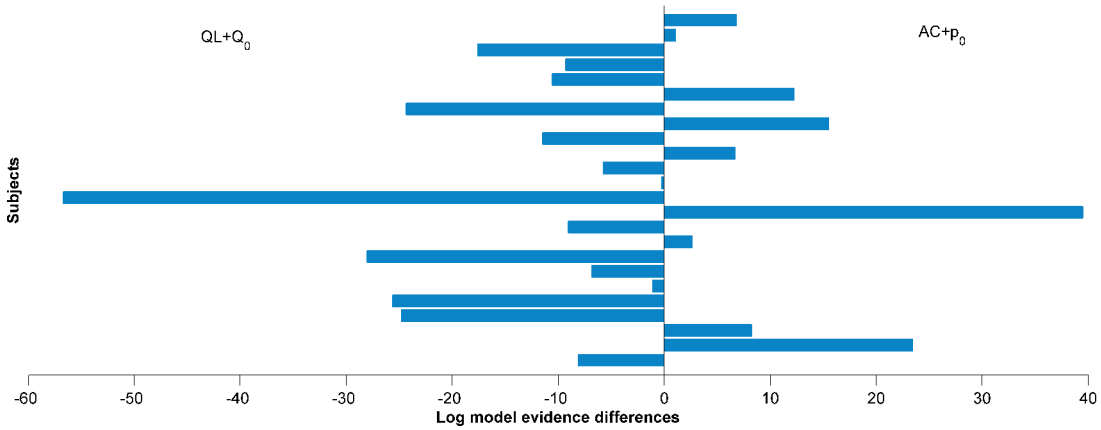

Figure A.6: Two variants of the QL and AC models were fitted and compared: (1) QL+Q_0; (2) AC+p_0. The horizontal bars show the difference in log-evidences for all twenty-four subjects.