A Generative Model of Exploration and Exploitation

A first implementation of predictive processing in abstract decision making processes.

Bachelor’s Thesis Artificial Intelligence

By Jesse Fenneman, 3036154

Under supervision of Johan H.P. Kwisthout & Maria E. Otworowska
Abstract
Known under different guises such as the multi-armed bandit problem, the trade-off between exploration and exploitation plays an influential role in human decision making processes, and consequentially, in the social and decision sciences. Predictions from reinforcement learning, the dominant approach for modelling such decision making processes, do not match observed human behaviours. Specifically, reinforcement learning cannot account for the fact that human deciders formulate generative world models, make inferences based on these models and show a developmental U curve in learning. The predictive processing account might provide for a better explanatory fit, but has not yet been implemented for such decisions. The present study investigates if – or to what extent – the predictive processing account is a candidate descriptive account for human decision making by constructing a novel and explorative predictive processing model. Although unable to provide a full test of the hypotheses, this model highlights several observations and design considerations, thereby informing future research.

*Keywords:* predictive processing, exploration, exploitation, generative world model
In our daily lives we often decide between trying new things to learn about our environment and relying on tried and true options that we know will be beneficial immediately. This trade-off between gathering new information, called *exploring*, and acting upon all the information that we have to select the best outcome, called *exploiting*, is a fundamental trade-off in the decision sciences. Consider for instance an animal that has to gather food in an environment that consists of several patches that vary in food availability: some reliably provide sustenance, while others often provide no food at all. Suppose that the animal has no initial knowledge of which patch will most likely provide food. In absence of this world knowledge, it faces a trade-off: it can inspect a large number of patches to get a good understanding of “what is out there”, or it inspect a few and settle for the most suitable one. This first option, which is high on *exploration*, increases long-term fitness by giving the animal a good understanding of its environment, guiding future actions. The second option, which is high on *exploitation* of known information, increases immediate payoff by forgoing on the investment cost of exploration.

How should this animal balance exploration and exploitation to maximize its food intake? Should it go through a long period of exploration and spend little time exploiting, should it spend a short time on exploration and a longer time on exploitation? Although under different guises (e.g., as the *multi-armed bandit* problem; Baron, 2008; Racey, Young, Garlick, Pham, & Blaisdell, 2011; Speekenbrink, & Konstantinidis, 2014; Sutton & Barto, 2014), the trade-off between exploration and exploitation is ubiquitous day-to-day life, and has consequentially received attention from many diverse fields, including psychology (Baron, 2008), economics (Shanks, Tunney, & McCarthy, 2002; Speekenbrink & Konstantinidis, 2014), biology (Cohen, McClure, & Angela, 2007; Daw, O’Doherty, Dayan, Seymour, & Dolan, 2006) and mathematics (Sutton, & Barto, 2014). Increasing our understanding of how this trade-off should be resolved from a normative point of view, and
how it happens to be resolved by human and non-human actors is therefore pivotal for a broad range of future investigations.

**Learning by trial and error: reinforcement learning**

A highly popular approach to modelling how human actors solve this trade-off is reinforcement learning (Sutton & Barto, 2015). Based on the influential work of Thorndike and his *law of reinforcement* models implementing reinforcement learning feature agents that learn by repeatedly sampling a possible action in a current state, observing the outcome of that action, and updating their beliefs about the world accordingly. This process of sampling is repeated many times, allowing the agent to derive correct estimates about the expected outcome of each action in each state. In the foraging problem described above, a reinforcement learning agent would repeatedly sample different patches, observe the outcome, and note the average food quality. In a typical implementation of reinforcement learning, agents initially focus on exploration, but place increasing focus on exploitation as its estimates about the consequences of its becomes more accurate (e.g. by implementing decaying $\varepsilon$-greedy action or Softmax selection strategies, see Sutton, & Barto, 2015 for further discussion).

**Unexplained observations**

The reinforcement learning procedure is similar to the behaviourist approach to psychology: rather than modelling an agent that relies on ‘cognitive’ processes such as induction and deduction of global rules (e.g., ‘patches that have a warmer climate have higher food availability’), reinforcement learning relies solely on observed input-output relations (e.g., ‘in the past, action A in state S has yielded an X amount of food’). Consequently, reinforcement learning is a brute-force approach: rather than understanding the causal mechanisms of its environment, an agents learn by performing each action in each state a large number of times. As such, these algorithms (Sutton, & Barto, 2014) rely on
bottom-up information streams, but neglect top-down abstract reasoning. Phrased differently, reinforcement learning agents learn input-output patterns, but hold no *generative model* of the world.

This seems to be at odds with how humans learn: typically, we form (implicit) generative models of the world from which we predict what the outcome of our actions will be (Clark, 2013). Thus, although this approach performs admirably as a normative model, its merit as a descriptive model of actual behaviour can be questioned. Specifically, there are three observations from real-life human behaviour in human decision making in general, and behaviour during exploration versus exploitation trade-off in specific, that are difficult to explain from a purely reinforcement learning perspective that does not incorporate a cognitive generative model.

**Observation 1: humans use induction to predict future outcomes.** The first observation is already hinted on above. Consider the following decision problem: you are in an environments consisting of five different patches where you can forage, such as restaurants in a city. Suppose that you have information on the food quality of each restaurant in the form of ratings on a 10 point scale, and that restaurants are ordered according to how distant they are from your current location. For one restaurant no ratings are available, resulting in the following five ratings: {5, ..., 7, 8, 9}. The question now is what the expected rating of the missing restaurant is, and subsequently which restaurant you should go to tonight.

Without any additional information, a reinforcement learning agent will have to sample the food at this missing restaurant and update its belief about the rating score before it can make a decision about what the best restaurant to go to is. Human actors might follow the same procedure, but additionally might be able to discern a rule in the data. For instance, the fact that restaurants that are further away receive higher ratings might indicate that you are currently in a poverty stricken neighbourhood. If this rule is true, the missing value is most
likely between a 5 and 7, and the best restaurant is found as far away as possible. Note that when one relies on the underlying (or *generative*) rule no trial and error learning is required. Instead, all current and future restaurant decisions can be made without any tedious trial-and-error learning. Phrased in a more abstract level, humans often perceive structure in their environment, allowing us to extrapolate results from previous actions to novel actions that have not been conducted yet. Typically, such extrapolation of known data to unknown states and actions is not possible in a reinforcement learning framework.

**Observation 2: human development shows an U pattern.** The second observation is that human cognitive development typically shows a U-shaped pattern (e.g. Baylor, 2001). Although this pattern has been observed across decision domains, it is still an active debate what the underlying mechanisms are that give rise to such a (broad ranged) developmental pattern. This shape emerges because humans typically go through several stages when learning new skills. When presented with a novel problem, we do not have any initial knowledge about the decision domain. As such, actions tend to be selected randomly, leading to a large amount of erroneous decisions. Over time these errors accrue, leading to a decrease in average performance (for instance, we might learn incorrect decision rules). However, as we progress and learn more about the decision problem, our generative model includes better decision rules that make more finely grained predictions, resulting in an increasingly better performance (for decisions in the clinical domain, see Witteman & Van den Bercken, 2007; Galanter, & Patel, 2005; for motion, see Gershkoff-Stowe & Thelen, 2004).

In contrast, the learning rate of reinforcement learning agents which determines how beliefs are updated in light of new information, is typically either held constant or is decreasing over time. As a result, learning in these algorithms tends to happen in a monotonous fashion: over time the algorithm’s estimates asymptotically approach the true estimates, resulting in continuously lowering errors and increasing performance.
**Observation 3: humans predict uncertainty.** A final consequence having a generative world model is the ability to differentiate between expected and unexpected uncertainty. The two types of uncertainty differ in whether or not they can predicted. Expected uncertainty is variance in the outcome on an action that is an irreducible aspect of the world, and thus cannot be predicted even if all information is known (e.g., sampling a patch that is known to have a 50 percent chance of a negative outcome and a 50 percent chance of a positive outcome). Errors resulting from this kind of uncertainty should not influence learning: after all, these errors cannot be learned. In contrast, unexpected uncertainty originates from erroneous predictions (e.g., observing a negative outcome when the patch was predicted to only provide positive outcomes), indicating that one’s hypotheses about the world are incorrect. In order to differentiate between these two uncertainties one requires a generative model of the world (Kwisthout, Bekkering, & Van Rooij, in press; see also Payzan-LeNesour & Bossaerts, 2011) that explains when outcome variance is expected. As stipulated above, the reinforcement learning framework typically does not implement a generative model, and hence cannot predict uncertainty. Consequentially, these models cannot determine when learning should not follow after receiving an outcome.

Human deciders, in contrast, do rely on these generative models. Empirical investigation find that the human brain does indeed differentiate between expected and unexpected uncertainty. For instance, when placed in an fMRI scanner, participants playing a multi-armed bandit task showed differential PFC activity when exploring or exploiting (Cohen, McClure, & Angela, 2007; Daw et al., 2006). Similarly, Yu and Dayan (2005) found that the two were associated with different neurotransmitter activity (acetylcholine and norepinephrine, respectively).

**Learning generative models: the predictive processing framework**
If reinforcement learning does not provide an adequate descriptive model of human decision making, then what other model do we have that can explain these observations? Or more specific, how do incorporate a generative model that seems to underlie human decision making? One prospective candidate is the predictive processing account (Clark, 2013; Kwisthout, Bekkering, & Van Rooij, in press).

Stated briefly, the predictive processing (or predictive-coding) theory states that the brain continuously makes predictions of what will happen according to an internal generative model. Rather than processing all available input stimuli, the brain processes only those stimuli that are observed but not predicted. Central in this account of cortical processes are three pillars: (1) the brain is organized in a hierarchical manner, (2) the brain uses top-down predictions from a generative model to augment bottom-up sensory information, and (3) rather than processing all inputs, the brain is proposed to processes only the input that is not explained by the generative models – i.e., the prediction error. In this account the brain is presumed to process information in an hierarchical and increasingly abstract set of hypotheses. That is, the predictions resulting from hypotheses on a superordinate level are used as the hypotheses on a subordinate level. On each level these predictions are compared to the prediction error, and the generating hypotheses are – if necessary – adjusted to reduce the discrepancy with the prediction error. The prediction error can be reduced through one of several ways, including revising existing hypotheses or gathering further information (Kwisthout, Bekkering, & Van Rooij, in press).

Although heralded at times as the unifying theory of cognition, the predictive processing account has been instantiated only for lower cognitive processes such as visual processing (Kwisthout, Bekkering, & Van Rooij, in press). For higher order processes such as decision making in general, and the exploration versus exploitation trade-off in specific, no concrete models have been developed. As such, it is unknown whether the predictive
processing account is a better descriptive model than reinforcement learning approach for the three observations that are listed above.

Present investigation

The present research investigates if – or to what extend – the predictive processing account is a candidate descriptive account for human decision making in tasks involving an exploration versus exploitation trade-off. Note that this question implies two different sub-questions, reflecting the duality of the present investigation.

Since the predictive processing account has not been investigated yet for (human) decision making processes, and no comparable instantiations of this theory exists – at least, as far as is known to the author – the present investigation will follow an explorative approach. Thus, the first objective is to investigate if any model can be constructed that implements the predictive processing account in a task featuring an exploration versus exploitation trade-off. To achieve this sub-goal we create a computational model of an predictive processing agent in a decision task where the exploration versus exploitation trade-off is influential, operationalize this model in a simulated environment, and explore agent’s behaviour. In this model an agent lives in a world with a fixed number of states to sample in, which are grouped together into patches. The objective of this agent is to create a generative world model that correctly specifies the expected outcomes of sampling each patch. Over time the number of patches increases, capturing the idea that over time deciders are able to specify more complex generative models that make more fine-grained predictions about the world. In the extreme case, when each state belongs to an unique patch, the agent will be able to perfectly predict the environment. Creating this initial model will allow us to make observations about the approach of modelling and investigate the necessary ‘design consideration’, informing future investigations.
A GENERATIVE MODEL OF EXPLORATION AND EXPLOITATION

Conditional on the success of this first goal, the second goal of the present investigation is to determine if modelled agent’s behaviour matches the first two observations listed above (note that by completion of the first sub-goal the third observation is automatically met: our model will include a generative model of the world). As such, we hypothesize that the accuracy of the agents when determining which patch to sample required comparatively small amount of learning and shows a developmental U curve.

Unfortunately, the (temporal) resources available for this project necessitate a limited scope. As such, the overarching objective of implementing the predictive processing account was divided into four subprojects, including the present investigation. The present subprojects focusses on investigation how the behavioural outcomes when the structure of the underlying model is manually changed. How such changes might be effected without outside influence is the purview of other individual projects in the overarching project. The structure of a network refers to the nodes and the connections between these nodes in a Bayesian network. Although this structure is provided manually and remains fixed, the probability distribution of these nodes is subject to change and are to be learned by the agent. The structure of this model is manually adjusted over time to represent more complex models.

In the present model the agent’s beliefs are captured in a Bayesian belief network. The agent has to derive correct estimates for the hypothesis nodes in this network (i.e., the most fundamental nodes that do not have incoming connections). In turn, these estimates influence three other layers of intermediate nodes in the network, finally resulting in a prediction node (i.e., a node that has no outgoing but no incoming arcs) that signifies the predicted outcome (for a further discussion on Bayesian modelling and the predictive processing account, see Kwisthout, Bekkering, & Van Rooij, in press). These outcomes could be either a positive or negative reward if the agent sampled a patch, or a neutral reward if the agent decides to move to a different patch. After making this prediction and observing the actual outcome the
difference between the two is calculated. Subsequently, the probability distributions of the hypothesis nodes are adjusted to minimize this prediction error.

**The model**

The full model of the present investigation is implemented in Python version 2.7 and relies on the predictive processing toolbox discussed in (Otworowska, Riemens, Kamhuis, Wolfert, Vuurpijl, & Kwisthout, in press). The implementation is provided with extensive in-code documentation and is available upon request.

---

**Figure 1.**

The environment of the agent, consisting of sixteen states which are grouped into three patches. Over time the number of patches are manually increased. Each state has a probability that sampling there will result in a positive outcome. When sampling a patch, the agent randomly samples in one of the states in this patch. Thus, the probability of a positive outcome when sampling a state is equal to the average probability of all states in that patch.
In the present model an agent resides in a 4 by 4 grid world consisting of 16 different states (Figure 1). Each state in this model has a unique probability $p$ that determines if sampling results in a positive (with probability $p$) or negative outcome (with probability $1-p$). States are grouped together into one or more patches. At every discrete moment the agent is in precisely one state and in one patch, and can decide to either move north, south, east or west (but has to stay within the grid world), or can sample the patch it is currently in. If the current patch is sampled, the agent randomly selects a state in that patch to sample in. As such, the expected probability of receiving a positive outcome when sampling a patch is equal to the average probability of receiving a positive outcome of all states in the patch. The agent holds no prior knowledge about the probabilities of each state, nor does it know the probability of a positive outcome when sampling any patch. Rather, it holds a belief about the outcome of sampling a patch, and updates this belief over time.

Over time, we manually increase the number of patches. This is done in the following procedure: the patch with the highest expected probability is selected, and is evenly divided into two new patches. If the original patch is wide rather than tall (i.e., spans more states on the x-axis than on the y-axis), the patch is divided into two new patches along the x-axis (e.g., the precursor to patches 2 and 3 in Figure 1). Otherwise, the patch is divided along the y-axis. After dividing the patches, the structure belief network (see below) is amended to represent the world with one additional patch. The expected outcomes of sampling in these novel patches are identical to the original patch’s expected outcome.
A GENERATIVE MODEL OF EXPLORATION AND EXPLOITATION

The belief network

What beliefs the agent holds about the probability of receiving a positive outcome when sampling a patch, and whether the agent will sample or move to a different state is given by the belief network of the agent. This Bayesian network consists of 4 layers of nodes, and implements the predictive processing account for the present model.

Figure 2. The initial belief network of the present model. This Bayesian network consists of 4 layers. Layer one consists of the nodes representing the beliefs of the agent that sampling in a patch will result in a positive outcome. The second layer consists of belief nodes concerning which patch is best to sampling in, which state the agent is currently in, and which patch the agent is currently in. The third layer represents the agent’s belief about which action {sample, go north, go east, go south, go west} is best. Finally, the fourth layer represents the agents belief about the outcome of the best action. After this action is taken, the actual outcome is observed and the hypothesis nodes from layer one are adjusted to minimize the prediction error between the observed outcome and the expected outcome (red arrows).
When constructing this model, several design considerations were encountered. Here, we first discuss the initial network that was implemented. Hereafter we make observations about the results of this implementation, discuss the design considerations that follow from these observations and implement these considerations to derive new observations.

**The initial belief network.** This network is graphically represented in Figure 2, and consists of four different layers. The first layer of this network contains the hypothesis nodes that represent the positive outcome probabilities for each patch. The nodes in this layer, named \{Patch1, Patch2, ..., Patch N\}, represent a probability distribution over the expected outcome of sampling in that patch, which can either result in a positive or a negative outcome.

The second layer consists of three nodes. The first node, named *current state*, represent the state the agent currently inhabits. This node consists of a probability distribution over all 16 states. Because the agent knows its location with certainty, this probability distribution always has a probability of 1 for the current state, and a probability of 0 for all other nodes (i.e., it is a deterministic node). The *current state* node has one outgoing connection, with the *current patch* node. The *current patch* node holds a probability distribution over all patch nodes and represents the patch the agent is currently in. Because this node is influenced solely by the *current state* node, its conditional probability distribution is likewise a 1 for the current patch, and 0’s otherwise. The third and final node in this layer is the *best patch* node, which holds a probability distribution over all patches and represents in what patch the agent is most likely to encounter a positive outcome. This node is conditioned on the patch nodes from the first layer. The conditional probability distribution of this node is determined as follows: if all patch nodes are hypothesized to result in a negative outcome, the probability distribution is uniform over all patch nodes. If there are \(N\) (with \(N\) larger or equal to 1) patches hypothesized to result in a positive outcome, the probability distribution for the *best patch* node has a value of \(1/N\) for the patches that are hypothesized to result in a
positive outcome, and a value of 0 for all patches hypothesized to result in a negative outcome.

The third layer consists of one node, named *best action*, which has three input nodes: *current state, current patch* and *best patch*. This node holds a probability distribution over each five actions that the agent can take: sample the current patch, go north, go east, go west and go south. This conditional probability distribution is determined as follows. Using a iterative depth-first search procedure, the agent finds all quickest routes from the current state and patch to the best patch. If this procedure results in no paths (i.e., a patch of length 0), the agent concludes that it already is in the best patch and the best action is to sample. Otherwise, the first action of each path (i.e., a singular move to go north, east, south or west) is placed into an array. For each action besides sampling the value in the conditional probability distribution in the *best action* node is the proportion of that action in this array. As an example, suppose that the agent is in state 6 and patch 1, and it believes the best patch to be patch 3. In this scenario there are two patch of length 2 that the agent can take to patch 3: {go south, go west} and {go west, go south}. In this case the conditional probability distribution of the *best action* node is {0, 0, 0, 0.5, 0.5} for the actions {sample, go north, go east, go south, go west}, respectively.

The fourth layer of the belief network consists of a single *prediction* node named *expected outcome*. This node holds a probability distribution over three possible outcomes: {negative outcome, neutral outcome, positive outcome}. This distribution is conditional on only the *best action* node: if the best action is to sample a positive outcome is expected with a probability of 1, if the best action is to move states, a neutral outcome is expected with a probability of 1.

*Selecting actions, observing outcomes and minimizing the prediction error.* The actual action that the agent takes is determined by the *best action node*: from the conditional
probability distribution of this node an action is selected and executed. Hereafter the actual outcome of this action is observed, denoted by the red node in Figure 2. The difference between the actual and expected outcome is determined by calculated the Kullback-Leibler divergence (also known as the relative entropy, Kwisthout et al, in press) between the two probability distributions. This prediction error is subsequently ‘pushed upstream’ (red arrows in Figure 2) to the patch nodes. The probability distributions of these hypothesis nodes is subsequently adjusted to minimize the prediction error.

**Decision accuracy and the complexity of the world model.** How well the agents beliefs match the true state of the world can be determined by calculating the distance between the hypothesized probability of receiving a positive outcome when sampling each patch, and the true probability of this outcome occurring. Initially, the generative world model captured in the belief network consists of two patch nodes in layer one. Such a generative model can specify only very coarse prediction: i.e., the agent can only hold beliefs that its better so sample the northern or southern patch. Over time, the number of patches is manually increased to reflect more complex generative models that are able to specify more fine-grained predictions.

**Results**

**Observation 1.** The conditional probability distribution for the *best patch* node is exponential in the number of patch nodes. Specifically, because each patch node can have two values, *positive outcome* and *negative outcome*, the number of values that the conditional probability distribution of the *best patch* node has to specify is $2^n$, where $n$ is the number of patch nodes in the network. Consequentially, the specified model becomes unworkable for all non-small numbers of patch nodes. Although this is not an issue for the present model, it seems unlikely that the human brain similarly requires a small number of ‘nodes’ to represent actual decision making processes.
**Observation 2.** A strictly hierarchical information processing will not result in a valid generative world model. To illustrate this point, consider the case that the agent has a belief network consisting of 3 patch nodes and currently inhabits patch 1. Suppose that the probabilities of a possible outcomes for each patch are \{0.1, 0.05, 0.01\}. In this situation the agent is already in the best patch and accordingly will sample. However, this action will result in the belief that a positive outcome is to be expected, even though this outcome is highly unlikely. This problem arises because the *expected outcome* node has no direct information about the patch probabilities and can be circumvented by directly conditioning the *expected outcome* node on the patches nodes. Note, however, that it cannot be ruled out that this problem can be resolved by different implementations of the generative world model. That is, it might be possible that a different *structure* of the belief network can be created such that nodes on a lower level need no direct access to information in nodes more than one level higher, thus maintaining a completely hierarchical network. As such, this problem might be idiosyncratic to the present model.
Design consideration 1. To base the expected outcome on additional information that is present in the patch nodes of layer 1, we added connections between layer 1 and layer 4 (green arrows in Figure 3). Note that this direct connection violates the strictly hierarchical processing hypothesized by predictive processing account. Additionally, the problem of complexity raised about the best patch node raised in observation 1 now also applies to the expected outcome node.

Observation 3. The predictive error can always be minimized by setting the probability distributions of the patch nodes such that they solely contain 0’s and 1’s. For instance, if the agent samples patch 1 and observes a negative outcome, the prediction error is minimized when the probability distribution for the node Patch 1 is set to \{0, 1\} for the
outcomes \{positive\ reward, negative\ reward\}, respectively, and the probability distributions of all other patch nodes are set to \{1, 0\}. Under these new values the agent will be certain that patch 1 will result in a negative outcome and sample patch 1 with a probability of 0. Although this minimizes the prediction error, this approach does not result in a valid generative world model.

**Design consideration 2.** One potential reason for observation 3 is that the network is memoryless: past observations do not influence how the prediction error should be minimized, and consequently have no influence on revising the probability distribution of the hypothesis nodes. To remedy this, we added a memory to the belief network that influenced the expected outcome node. This memory kept track of all positive, negative and neutral outcomes encountered in each patch. Rather than deriving a prediction on the basis of only the
A GENERATIVE MODEL OF EXPLORATION AND EXPLOITATION

*best action* node and the patch nodes, the *expected outcome* node adds the memory values of
the current patch to the expected outcomes. The resulting array is then normalized into a
probability distribution, and is used as the expected outcome. In essence, rather than
predicting the result of the present action, the belief network now makes a prediction over all
past and current outcomes. Similarly, the observed outcome is a probability distribution over
all outcomes observed thus far for the patch the agent is currently in.

An example might be insightful. Suppose that the agent decides to sample
patch 1 and
holds a conditional probability distribution of \(\{0.2, 0.3, 0.5\}\) for the actions \{negative
outcome, neutral outcome, positive outcome\}. Furthermore, previously the agent has
observed 3 negative outcomes, 7 neutral outcomes and 11 positive outcomes when it was in
patch 1. Rather than have a probability distribution of \(\{0.2, 0.3, 0.5\}\) as the expected outcome,
these values are added to the memory (i.e., \(\{3.2, 7.3, 11.5\}\)) and normalized into a new
probability distribution (i.e., \(\{0.15, 0.33, 0.52\}\)). If the agent observes a novel negative outcome,
the observed outcome is then the normalized probability distribution over all observed
outcomes ( \(\{3 + 1, 7, 11\}\), which when normalized becomes \(\{0.18, 0.32, 0.50\}\)).

**Observation 4.** Adding a memory to the system results in more valid belief for the
patch nodes, but only when the memory contains a small number of occurrences. Initially,
beliefs about the consequences of sampling in a patch node take on values in the full range
between 0 and 1, and over time approach the true probability distribution of those patches.
However, as the memory contains more and more information, the prediction error
asymptotically approaches 0, *regardless of the prediction made for the current action*. The
cause of this asymptote is that, as the total number of occurrences increases over time, the
influence of the predicted outcome for the current action on the final expected outcome
becomes less influential. Simultaneously, the influence of one additional observation shapes
the observed outcomes less and less. In other words, because the observed outcome and the
expected outcome are both determined by past outcomes, these two distributions becomes more similar when the memory plays a more influential role.

**Observation 5.** The problem of diminished prediction errors regardless of decision accuracy cannot be resolved by relying on the simplified belief model described during the method section (i.e., by dropping the green arrows in Figure 4). At the time of writing, we know of no manner to solve this problem. Consequentially, the model is unable to process a large number of observations, inhibiting the model’s ability to compare decision accuracy between more complex models. Despite this inability, the present study is still able to provide tentative observations on how this belief model influences agent’s behaviour.

**Observation 6 (tentative).** One of such observations is that the agent does not participate in any exploration of its environment. Rather, the agent always samples the patch that has the highest hypothesized probability of resulting in a positive outcome. Consequentially, if due to chance events the hypothesized probability of receiving a positive outcome be estimated as too low, this patch is infrequently or never sampled afterwards. However, it should be noted that this observation is based on a small amount of observations made by the agent, and hence it cannot be ruled out that longer runs of the algorithm resolve this issue.

**Discussion**

The present model is, to the extent of our knowledge, the first to implement the predictive processing account in a decision task that features a trade-off between exploration and exploitation. Previously this decision has been modelled using reinforcement learning. However, this approach results in predictions that do not match observed human behaviour, and hence is unlikely to describe human decision making processes. Specifically, in contrast to reinforcement learning models, human actors hold generative models about their environment, can make inferences about unobserved actions and learn in a U-shaped
developmental pattern. The present study consisted of two sub-goals. The first goal was to investigate if such a model can be constructed. The second goal was to investigate if the predictive processing account provides a more suited descriptive fit to human decision making.

Unfortunately, the present study was unable to construct a model that implements the predictive processing account to the extent that the second goal could be investigated. The principle reason for this failure is that a memoryless system results in the predictive error being minimized by setting dichotomizing the probability distributions of the hypothesis nodes, whereas a system with a memory results in a disappearing prediction error. This failure can inform future research: by itself, a reducing in the prediction error alone does not necessarily entail that the generative world model is correct. Rather, the prediction error can be minimized by a – ecologically invalid – model. However, this result is tentative in nature: we cannot rule out that different implementations of the belief network (e.g., a network with a different structure) will result in prediction error minimization that does not rely on dichotomizing probability distributions.

A second, albeit tentative, result from the present that can inform future research is that by itself the predictive processing framework does not result in explorative behaviour. Rather, the agent will consistently sample the patch with the highest expected payoff – even if other patches have not or infrequently been visited. This issue can be resolved by either providing additional incentives gathering information (e.g., by incorporating not just the expected outcome of each patch, but also the confidence intervals of these outcomes) or by incorporating strategies such as the famous epsilon-greedy that occasionally result in random actions. To the extent of our knowledge, which method – if any – is a better descriptive account of human decision making is thus far an open question.
A third and final contribution from this study is that it provides future research with a framework of how to specify an exploration versus exploitation decision, how to implement the predictive processing account, and how to test whether this account provides a better descriptive fit. We feel confident that if the limitations of the present model can be resolved, this framework can provide a valid and valuable test of the predictive processing account.
References


Kwisthout, Johan, Bekkering, Harold, & Van Rooij, Iris (in press). To be precise, the details don't matter: On predictive processing, precision, and level of detail of predictions.

*Brain and Cognition*


