First draft simulation

1 Outline simulation environment

Given our individual research questions we have decided to make a general environment which is applicable for all research questions and can be further evolved to fit our individual experiments.

We will implement a grid of tiles with distinct features per tile (e.g., blue/red/green). When a tile is flipped it reveals an object (e.g., apple banana, orange). The object revealed is connected to the feature on the tile. For now we will keep the grid at size 3x1, with 3 features and 3 objects (i.e., no tile has the same feature/object as another tile). After completing this we might increase the size of the grid (e.g., 3x3) to also take spatial information into account as now tiles can have the same feature and object. The will also allow for adaptations to be made regarding for example volatility and noise, however these adaptations depend on the requirements for the personal projects so for now they will not be implemented.

The idea is that the agent should learn the relation between which feature it sees on the tile and the object which is revealed once the tile has been turned. Initially we will have a passive agent, thus the agent will be told which of the tiles is going to be flipped. Based on on the agent's model it will predict which object is under the tile, given the feature of the tile. Or as explained step-wise: 1) "This tile is gonna flip", 2) agent makes prediction, 3) observe & compare to prediction. At a later point we will see if we can make it possible for the agent to choose which tile to flip according to entropy, however this is not a priority.

On the following two pages you will find the graphical and generative models for the environment and the agent that we constructed based on our implementation ideas. There is also corresponding explanation of the reasoning behind the models. These models have been created in a Bayesian Statistics manner.

2 Environment

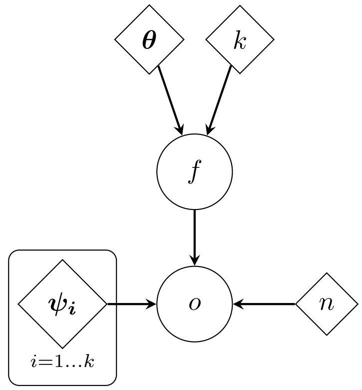

Figure 1: Graphical model of the environment

World model:

$$f \mid \theta, k \sim Categorical(\theta, k) \tag{1}$$
$$o \mid \psi_i, f, n \sim Categorical(\psi_f, n) \tag{2}$$

Figure 1 shows the graphical model of the environment. Here the node o represents the object which can be found underneath a tile. Which object is shown under the tile is dependent on the feature of the tile f and the corresponding probability distribution. We decided to make the initial environment deterministic. Thus, each probability distribution $\boldsymbol{\psi_i}$, for feature i, puts all its probability mass on one object i.e. one object shows with 100% probability and the other objects have 0% probability of appearing when the tile is turned. The feature of a tile is drawn from a categorical distribution with a uniform distribution over all features. This uniform distribution is represented by the vector of category probabilities $\boldsymbol{\theta}$. The hyperparameter k represents the number of possible features and the hyperparameter n the number of possible objects. Note that the number of possible objects and features does not have to be the same.

3 Agent

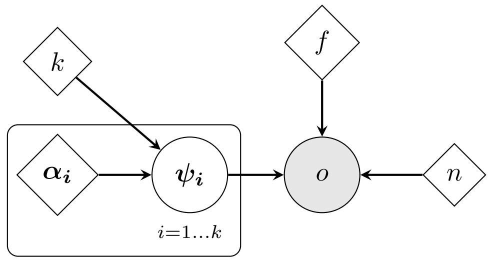

Figure 2: Graphical model of the agent

Generative model:

$$\psi_i \mid \alpha_i, k \sim Dirichlet(\alpha_i, k) \tag{3}$$
$$o \mid \psi_i, f, n \sim Categorical(\psi_f, n) \tag{4}$$

Figure 2 shows the graphical model of the agent. Each of the feature-specific concentration vectors $\boldsymbol{\alpha}$ is initially set to a uniform vector with one pseudo-observation for each object. The probability distribution $\boldsymbol{\psi_i}$ over the objects for each of the features is sampled from a Dirichlet distribution with a feature specific concentration vector $\boldsymbol{\alpha_i}$. Also here k represents the number of possible features and n the number of possible objects. The object is predicted based on a categorical distribution over the probability distribution $\boldsymbol{\psi}$ given the observed feature f. Note that the hyperparameters k and n are constants within a simulation and thus they differ from the hyperparameters $\boldsymbol{\alpha_i}$ and f that are updated after each observation.