MOD202 Excercise 6

Ciprian Bangu

April 22 2024

Static action choice and rewards

a) Given that

$$p(c=1) = \frac{\exp(\beta m_1)}{\exp(\beta m_1) + \exp(\beta m_2)} \tag{1}$$

and

$$p(c=2) = \frac{\exp(\beta m_2)}{\exp(\beta m_1) + \exp(\beta m_2)} \tag{2}$$

$$\begin{aligned}\sum_{c=1}^{2} p(c) &= \frac{\exp(\beta m_1)}{\exp(\beta m_1) + \exp(\beta m_2)} + \frac{\exp(\beta m_2)}{\exp(\beta m_1) + \exp(\beta m_2)} \\ &= \frac{\exp(\beta m_1) + \exp(\beta m_2)}{\exp(\beta m_1) + \exp(\beta m_2)} \\ &= 1\end{aligned}$$

b)

$$\begin{aligned}p(c=1) &= \frac{\exp(\beta m_1)}{\exp(\beta m_1) + \exp(\beta m_2)} \\ &= \frac{1}{1 + \frac{\exp(\beta m_2)}{\exp(\beta m_1)}} \\ &= \frac{1}{1 + \exp(\beta(m_2 - m_1))}\end{aligned}$$

c) d corresponds to the difference in the bee's internal estimate of the reward it will recieve from each type of flower. m_1 is its estimated reward for the blue flowers, while m_2 is its estimated reward for the yellow flowers. When the bee estimates that the yellow flower will provide a lower reward than blue, this difference will be negative. Conversely, when the bee estimates that the yellow flower will provide a higher reward than the blue, this difference will be positive.

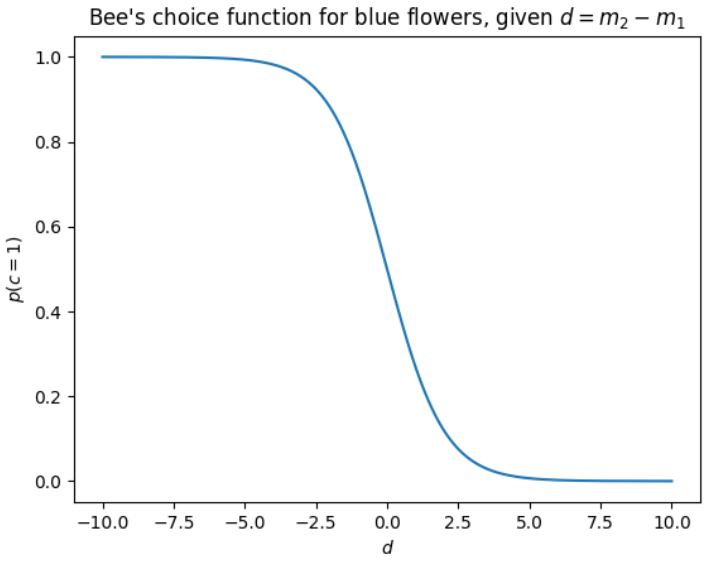

Figure 1: The probability of choosing the blue flower as a function of the difference between the bee's internal reward estimates. $\beta = 1$

Figure 1. shows how the probability of the bee's chosing the blue flowers changes as a function of d. We see that when d is negative, i.e., when $m_2 < m_1$, $p(c = 1) > 0.5$, i.e., the bee is more likely to go to the blue flowers. For $d < -10$ this probaility approaches 1. Conversely, when d is positive, i.e., when $m_2 > m_1$, $p(c = 1) < 0.5$, i.e., the bee is more likely to go to the yellow flowers. For $d > 10$ the probability that the bee chooses blue approaches 0. When $d = 0$, the bee is equally likely to go to either flower.

This intuitively makes sense: if I want to maximise my reward, and I estimate the reward from blue flowers is greater than yellow flowers, I should tend to choose blue flowers (and vice-versa).

d) An exploitative decision strategy is one in which an agent chooses the action that is deemed to have the highest payoff based on current knowledge of the world. An explorative strategy is one in which the agent sometimes chooses an action that is not known to currently have the highest payoff, but that may lead to better outcomes in the future. In our model, the β parameter determines whether the bee's behavior is exploitative or explorative.

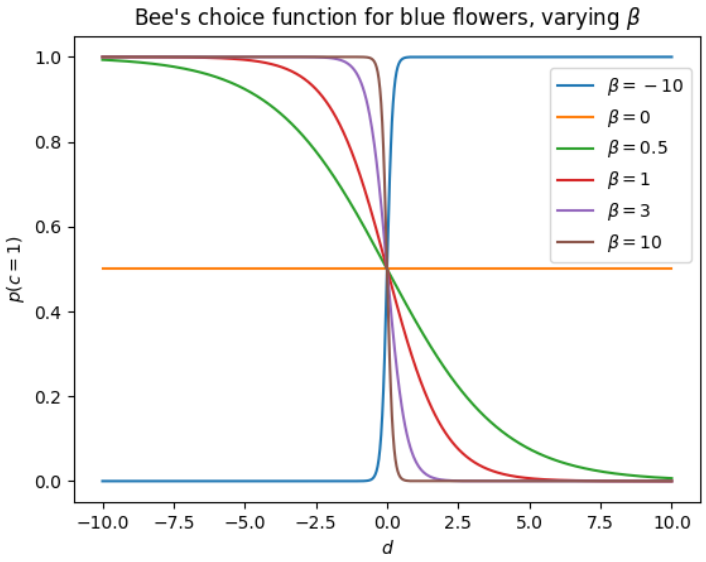

Figure 2: Probaility of choosing a blue flower given different values of β

Figure 2 demonstrates this. As β changes, the slope of the funciton changes. When β is small but greater than 0, the bee is more exploratory than exploitative. The probabilty it will choose the blue flower decresease gradually as d increases. Even though the bee estimates that the blue flower will provide a higher reward, it is willing to sometimes choose the yellow flower.

When $\beta = 0$, the bee is purely exploratory in that it chooses randomly: $p(c = 1) = 50$ for any d, so the bee is equally likely to choose either flower regardles of its internal reward estimates.

When β is large, the bee is more exploitative: i.e., the probability it will choose the blue flower remains high until d gets very close to 0. This results in a much steeper slope of the curve around $d = 0$ than when β is small. The bee is only willing to choose the yellow flower when it esimtates that it is nearly as good, or better, than the blue flower.

It is also technically possible for β to be negative. Shown in Fig 2., this results in the curve being mirrored: the bee is less likely to choose blue if $m_1 > m_2$. However, this behavior would imply that the bee is looking to minimize its reward, which, if actually occuring in nature, would likely be a special case. Although, this model can also capture this behavior.

e) To extend the choice action strategy to N flowers, let us first consider the extension to 3 flowers.

From a), we know that the $\sum_{c=1}^{N} p(c) = 1$. Therefore, the probability of choosing any flower when the total number of flowers = 3, is:

$$\begin{aligned} p(c=1) &= \frac{\exp(\beta m_1)}{\exp(\beta m_1) + \exp(\beta m_2) + \exp(\beta m_3)} \\ p(c=2) &= \frac{\exp(\beta m_2)}{\exp(\beta m_1) + \exp(\beta m_2) + \exp(\beta m_3)} \\ p(c=3) &= \frac{\exp(\beta m_3)}{\exp(\beta m_1) + \exp(\beta m_2) + \exp(\beta m_3)} \end{aligned} \tag{3}$$

since this formulation allows for $\sum_{c=1}^{3} p(c) = 1$.

Therfore, the action-choice rule from b) can be extended as follows:

$$\begin{aligned} p(c=1) &= \frac{\exp(\beta m_1)}{\exp(\beta m_1) + \exp(\beta m_2) + \exp(\beta m_3)} \\ &= \frac{1}{1 + \frac{\exp(\beta m_2) + \exp(\beta m_3)}{\exp(\beta m_1)}} \\ &= \frac{1}{1 + \frac{\exp(\beta m_2)}{exp(\beta m_1)} + \frac{\exp(\beta m_3)}{\exp(\beta m_1)}} \\ &= \frac{1}{1 + \exp(\beta(m_2 - m_1)) + \exp(\beta(m_3 - m_1))} \end{aligned}$$

We can then extend the above formula to N flowers:

$$p(c=1) = \frac{1}{1 + \sum_{i=2}^{N} \exp(\beta(m_i - m_1))} \tag{4}$$

Or, more generally:

$$p(c=i) = \frac{1}{1 + \sum_{j=1, j\neq i}^{N} \exp(\beta(m_j - m_i))} \tag{5}$$

In the case of N flowers, the bee can either trade off exploration and exploitation by changing β, which causes the action choice strategy to behave in the way described in d), just with more flowers to choose from.

f) The bee could adapt its internal estimates using an update rule, like the delta learning rule. Every time the bee visits a flower, it can update its internal estimate of the reward of that flower type by adding the difference between the recieved reward and the estimated reward, multiplied by a learning rate.

let $r_{i,t}$ be the reward given by flower i at time t, and let $m_{i,t}$ be the internal estimate of the reward of flower i at time t, and ϵ is the learning rate, i.e., the weight of the difference between actual and expected reward. ϵ can range between 0 and 1. Then the update rule would be:

$$m_{i,t} = m_{Ni,t-1} + \epsilon(r_{i,t} - m_{i,t-1}) \tag{6}$$

Note, the bee only carries out the update if it visits the flower. Otherwise $r_{i,t} = 0$ and the bee would update m_i based on $m_{i,t} - \epsilon m_{i,t}$, meaning it would update its internal estimate despite not getting any feedback from the environment at that timestep. So, the full condition for the update, $m_i(t)$, would be:

$$m_i(t) = \begin{cases} m_{i,t-1} + \epsilon(r_{i,t} - m_{i,t-1}) & \text{if bee visits flower } i \text{ at time } t \\ m_{i,t-1} & \text{otherwise} \end{cases} \tag{7}$$

g) If the rewards of the flowers stay constant, the bee will eventually learn the true reward of each flower type because the update rule in f) will converge to the true reward value. That is, if the bee applies the update rule for a sufficient amount of iterations, I_{sf}, eventually $m_i(t) = r_i$, for all i which the bee has visited at least I_{sf} times.

This process depends on β in that, if β is high, the bee will be more exploitative. Therefore, it will initally heavily bias its learning towards the flowers it estimated to have the highest reward. At least initially, it will be very unlikely that the bee visits a flower which it estimates to have a lower reward than another flower. As it begins to learn the rewards and, m_i converges to r_i, the bee will become more likely to visit other flowers, since the difference between the estimated rewards will be smaller. However, if there are any flowers whose estimated reward is **much** lower than both $m_{i(initial)}$ and r_i of every other flower, then it is very unlikely that the bee will ever visit those flowers if it only has finite time. This is because, by the process sketched above, its estimate of the rewards of the other flowers is now the true reward, and thus d between the true reward and the estimate of the low-reward flower(s) is very high, and can only get smaller if it visits that/those flower/s. But since β is high, it is unlikely to visit that/those flower/s, given a finite time window. Therefore, it is unlikely to learn the true reward of that/those flower/s. Likewise, if through the initial exploration, it finds a flower/s with a true reward much higher than its internal estimate for all the other flowers, the likelyhood it visits the other flowers, and thus learns their true reward, is low given a finite amount of time.

If β is lower, the bee is more exploratory, and will visit different flowers more often despite the difference in estimated rewards. Thus, the above situation is less likely to occur: the bee's visiting of all the flowers is not blocked by the high d between the low-reward flower and the other flowers. In finite time, however, given a small learning rate, bad internal estimates, and a large amount of flowers, the bee may not learn the true reward of **all** the flowers.

Finally, if β is 0, the bee will be purely exploratory, and will have no bias towards learning the true reward of any flower. However, if the learning rate is too small, and the esitmates are

bad, it may not learn the true reward of **any** flower, since it may not visit any single flower I_{sf} times given a finite amount of time.

The characteristic time constant of convergence, τ, is commonly defined as the time it takes for the error to reduce $\frac{1}{e}$ of its initial value.

We can find this value by looking at the error reduction during each timestep. Let e_t be the error at time t, and since the rewards are constant, let $r_{i,t} = r_i$. Then, the error at time t is:

$$e_t = r_i - m_{i,t} \tag{8}$$

Similarly, the error term at time $t-1$ is:

$$e_{t-1} = r_i - m_{i,t-1} \tag{9}$$

We can then substitute the update rule from f) into the error equation:

$$e_t = r_i - (m_{i,t-1} + \epsilon(r_i - m_{N,t-1}))$$

Expanding, we get:

$$e_t = r_i - m_{i,t-1} - \epsilon(r_i - m_{N,t-1})$$

Substituting e_{t-1}:

$$e_t = e_{t-1} - \epsilon(e_{t-1})$$

re-arranging again:

$$e_t = (1-\epsilon)e_{t-1} \tag{10}$$

From equation 9, we see that the error term at each timestep is reduced by $(1-\epsilon)$. Therefore, we can find the time constant of convergence by solving the below equation for τ:

$$(1-\epsilon)^\tau = \frac{1}{e} \tag{11}$$

$$\tau \ln(1-\epsilon) = \ln \frac{1}{e}$$
$$\tau \ln(1-\epsilon) = -1$$

Therefore:

$$\tau = \frac{-1}{\ln(1-\epsilon)} \tag{12}$$

Equation 11 give the characteristic time constant of convergence for the bees internal reward estimates. This time constant depends on the learning rate, ϵ. If ϵ is high, the bee will learn the true reward values quickly, and τ will be small. Conversely, if ϵ is low, the bee will take longer to learn the true reward values, and τ will be large.