

1 Exercise (*Probabilities (2p)*)

- Let's assume we grab one candy from bag b1 at random (no see through bag). What is the probability of getting a red one/ green one/ yellow one?
 $P(b1 = r) = \frac{5}{10} = 0.5$
 $P(b1 = g) = \frac{3}{10} = 0.3$
 $P(b1 = y) = \frac{2}{10} = 0.2$

- If one bag is chosen at random with probabilities $p(b1) = 0.2$, $p(b2) = 0.3$, $p(b3) = 0.5$, and from that bag we grab one candy at random. Then what is the probability of getting a red one/green one/yellow one?
 $P(Red) = P(Red|B = b1) * (PB = b1) + P(Red|B = b2) * (PB = b2) + P(Red|B = b3) * (PB = b3)$
 $= 0.1 + 0.05 + 0.1818 = \mathbf{0.3318}$
 $P(Green) = P(Green|B = b1) * (PB = b1) + P(Green|B = b2) * (PB = b2) + P(Green|B = b3) * (PB = b3)$
 $= 0.06 + 0.1 + 0.09090909 = \mathbf{0.2509}$
 $P(Yellow) = P(Yellow|B = b1) * (PB = b1) + P(Yellow|B = b2) * (PB = b2) + P(Yellow|B = b3) * (PB = b3)$
 $= 0.06 + 0.15 + 0.22727272 = \mathbf{0.4372727}$

2 Exercise (*Bayes classifier (8p)*)

- Given the data set in Table 2, determine all probabilities required to apply the Bayes classifier for predicting whether a new person is ill or not. Use a naive approach, where all attributes are assumed to be independent. (4p)
We approximate the real probability through the relative frequencies from the data.
 $P(C = ill) = 0.5$, $P(C = healthy) = 0.5$
 $P(N = +|C = ill) = 2/3$, $P(N = -|C = healthy) = 1/3$
 $P(Co = +|C = ill) = 2/3$, $P(Co = -|C = healthy) = 1/3$
 $P(R = +|C = ill) = 2/3$, $P(R = -|C = healthy) = 1/3$
- Calculate the probabilities of persons d1,..., d6 being ill. (2p)
Probability that d1 is ill/healthy 0.88889 0.11111
Probability that d2 is ill/healthy 0.66667 0.33333
Probability that d3 is ill/healthy 0.33333 0.66667
Probability that d4 is ill/healthy 0.33333 0.66667
Probability that d5 is ill/healthy 0.11111 0.88889
Probability that d6 is ill/healthy 0.66667 0.33333

```

% P(C=c | F1= f1 , F2 = f2 , F3 = f3)
% = N * P(F1= f1 , F2 = f2 , F3 = f3 | C = c) * P(C = c)
% = N * P(F1= f1| C = c) * P( F2 = f2| C = c) * P( F3 = f3 | C = c) * P(C = c)

p_cc = [0.5 , 0.5];
p_f1 = [2./3. , 1./3.];
p_f2 = [2./3. , 1./3.];
p_f3 = [2./3. , 1./3.];

p_d1ill = p_cc(1) * p_f1(1) * p_f2(1) * p_f3(1);
p_d1heal = p_cc(2) * p_f1(2) * p_f2(2) * p_f3(2);
prob_d1 = p_d1ill / (p_d1ill + p_d1heal);
display(strcat('Probability that d1 is ill/healthy ',num2str(prob_d1),' ',num2str(1-prob_d1)));

p_d2ill = p_cc(1) * p_f1(1) * p_f2(1) * (1-p_f3(1));
p_d2heal = p_cc(2) * p_f1(2) * p_f2(2) * (1-p_f3(2));
prob_d2 = p_d2ill / (p_d2ill + p_d2heal);
display(strcat('Probability that d2 is ill/healthy ',num2str(prob_d2),' ',num2str(1-prob_d2)));

p_d3ill = p_cc(1) * (1-p_f1(1)) * (1-p_f2(1)) * p_f3(1);
p_d3heal = p_cc(2) * (1-p_f1(2)) * (1-p_f2(2)) * p_f3(2);
prob_d3 = p_d3ill / (p_d3ill + p_d3heal);
display(strcat('Probability that d3 is ill/healthy ',num2str(prob_d3),' ',num2str(1-prob_d3)));

p_d4ill = p_cc(1) * p_f1(1) * (1-p_f2(1)) * (1-p_f3(1));
p_d4heal = p_cc(2) * p_f1(2) * (1-p_f2(2)) * (1-p_f3(2));
prob_d4 = p_d4ill / (p_d4ill + p_d4heal);
display(strcat('Probability that d4 is ill/healthy ',num2str(prob_d4),' ',num2str(1-prob_d4)));

p_d5ill = p_cc(1) * (1-p_f1(1)) * (1-p_f2(1)) * (1-p_f3(1));
p_d5heal = p_cc(2) * (1-p_f1(2)) * (1-p_f2(2)) * (1-p_f3(2));
prob_d5 = p_d5ill / (p_d5ill + p_d5heal);
display(strcat('Probability that d5 is ill/healthy ',num2str(prob_d5),' ',num2str(1-prob_d5)));

p_d6ill = p_cc(1) * (1-p_f1(1)) * p_f2(1) * p_f3(1);
p_d6heal = p_cc(2) * (1-p_f1(2)) * p_f2(2) * p_f3(2);
prob_d6 = p_d6ill / (p_d6ill + p_d6heal);
display(strcat('Probability that d6 is ill/healthy ',num2str(prob_d6),' ',num2str(1-prob_d6)));

```

3. Calculate the probabilities of the following subjects being ill: (2p)

- a person who is coughing and has fever

$$\begin{aligned}
 &P(C = ill|Co = +, F = +) \\
 &= N * P(Co = +, F = +|C = ill) * P(C = ill) \\
 &= N * (2/3 * 1/3 * 1/2) = N * 1/9 \\
 &P(C = healthy|Co = +, F = +) \\
 &= N * P(Co = +, F = +|C = healthy) * P(C = healthy) \\
 &= N * (1/3 * 0 * 1/2) = N * 0 \\
 &\rightarrow P(C = ill|Co = +, F = +) = 1
 \end{aligned}$$
- a person whose nose is running and who suffers from fever

$$\begin{aligned}
 &P(C = ill|N = +, F = +) \\
 &= N * P(N = +, F = +|C = ill) * P(C = ill) \\
 &= N * (2/3 * 1/3 * 1/2) = N * 1/9 \\
 &P(C = healthy|N = +, F = +) \\
 &= N * P(N = +, F = +|C = healthy) * P(C = healthy) \\
 &= N * (1/3 * 0 * 1/2) = 0 \\
 &\rightarrow P(C = ill|N = +, F = +) = 1
 \end{aligned}$$
- a person with a running nose and reddened skin $P(C = ill|N = +, R = +)$

$$\begin{aligned}
 &= N * P(N = +, R = +|C = ill) * P(C = ill) \\
 &= N * (2/3 * 2/3 * 1/2) = N * 2/9 \\
 &P(C = healthy|N = +, R = +) \\
 &= N * P(N = +, R = +|C = healthy) * P(C = healthy) \\
 &= N * (1/3 * 1/3 * 1/2) = N * 1/18 \\
 &\rightarrow P(C = ill|N = +, R = +) = 4/5
 \end{aligned}$$

3 Exercise (*Reinforcement Learning (10p)*)

1. Approximating the Value Function. Assume the action sequence of the mouse ($\rightarrow \uparrow \leftarrow \uparrow \rightarrow \rightarrow$) is applied to the Figure 1. Calculate $V(s_t)$ with discount factor $\gamma = 0.9$. (2p)

$$V(s_t) = 0 + 0 + 0 + 0 + 0 + 100 * 0.9^6 \approx 53.1$$

2. Let's start the first episode from the lower left corner where the mouse is. Perform three episodes of Q-Learning for this example. (8p)

We don't implement the exercise, but instead just imagine an appropriate case where all interesting combinations are played through. Conveniently in this made-up example our mouse is always randomly placed in the bottom left corner after finding the cheese. We assume the following numbering: bottom left corner is (1,1) and then as usual (i.e. the field to the right of that is (1,2)).

1st episode: We assume that our mouse is kind of clever and goes straight to the cheese (without visiting the bottom right corner). Since all rewards to get there are equal to zero no updates occur before that. When the

mouse does the transition to the top right corner from the right middle (tile (2,3)) it gets the reward 100 and therefore the value of $q((2,3), \uparrow)$ is updated to $100 + 0.9 * \max_{a'} q((3,3), a') = 100 + 0 = 100$

2nd episode: We assume that our mouse is kind of adventurous and goes straight to the cheese (without visiting the bottom right corner or the (2,3) tile). Since all rewards to get there are equal to zero no updates occur before that. When the mouse does the transition to the top right corner from the upper middle (tile (3,2)) it gets the reward 100 and therefore the value of $q((3,2), \rightarrow)$ is updated to $100 + 0.9 * \max_{a'} q((3,3), a') = 100$

3rd episode: Now we can't make any convenient choices any more and our mouse has to take a path like $(1,1) \rightarrow (1,2) \rightarrow (1,3) \rightarrow (2,3) \rightarrow (3,3)$. When doing the transition from $(1,2) \rightarrow (1,3)$ the mouse gets a reward of 20 and therefore $q((1,2), \rightarrow)$ is updated to $20 + 0.9 * \max_{a'} q((1,3), a') = 20$. When the mouse then does the transition $(1,3) \rightarrow (2,3)$ it does not receive a reward but the value of $q((1,3), \uparrow)$ is updated as $0 + 0.9 * \max_{a'} q((2,3), a') = 0.9 * 100 = 90$.