Morgan Ciliv
Tom Wiliams
Artificial Intelligence
April 17, 2017

Homework #9

1. MDPs
    1. Value iteration:

- There are 3 states, A, B, and C, which are listed as the columns.
- The values are calculated by using the previous iterations and simply applying the expectation using the previous values of the state and the corresponding probabilities for each state transition.
- The process is done for both the shift and jump options.
- Max is the maximum value of the Shift or Jump for the given k
- k represents the number of iterations. The first row is the initial values for each state which were given. The iterations are as follows:

	k	A	B	C
Initial	**1**	-1	-2	0
Shift	**2**	-2.4	-3	0
Jump	**2**	-1.72	-3.44	0
Max	**2**	-1.72	-3	0
Shift	**3**	-3.144	-3.632	0
Jump	**3**	-2.2384	-4.16	0
Max	**3**	-2.2384	-3.632	0
Shift	**4**	-3.62688	-4.06944	0
Jump	**4**	-2.611648	-4.61504	0
Max	**4**	-2.611648	-4.06944	0
Shift	**5**	-3.9639936	-4.3808768	0
Jump	**5**	-2.88038656	-4.9299968	0
Max	**5**	-2.88038656	-4.3808768	0
Shift	**6**	-4.204603392	-4.604407296	0

Jump	**6**	-3.0738783232	-5.154231296	0
Max	**6**	-3.0738783232	-4.604407296	0
Shift	**7**	-4.37742004224	-4.76520845312	0
Jump	**7**	-3.213192392704	-5.31517325312	0
Max	**7**	-3.213192392704	-4.76520845312	0
Shift	**8**	-4.5017635504128	-4.8809571262464	0
Jump	**8**	-3.31349852274688	-5.4309500862464	0
Max	**8**	-3.31349852274688	-4.8809571262464	0
Shift	**9**	-4.59127398029722	-4.96429053889741	0
Jump	**9**	-3.38571893637775	-5.51428913089741	0
Max	**9**	-3.38571893637775	-4.96429053889741	0
Shift	**10**	-4.655718110614	-5.02428946960613	0
Jump	**10**	-3.43771763419198	-5.57428918800613	0
Max	**10**	-3.43771763419198	-5.02428946960613	0
Shift	**11**	-4.70211720860208	-5.06748847443642	0
Jump	**11**	-3.47515669661823	-5.61748841811642	0
Max	**11**	-3.47515669661823	-5.06748847443642	0
Shift	**12**	-4.7355244239855	-5.09859171285822	0
Jump	**12**	-3.50211282156512	-5.64859170159422	0
Max	**12**	-3.50211282156512	-5.09859171285822	0
Shift	**13**	-4.75957759202796	-5.12098603551072	0
Jump	**13**	-3.52152123152689	-5.67098603325792	0
Max	**13**	-3.52152123152689	-5.12098603551072	0

Using 13 iterations, utilities: A = -3.52152123152689, B = -5.12098603551072, C = 0

Now calculating the optimal action…

U(A) for S: 0.25 * -3.52152123152689 + 0.75 * -5.12098603551072 = -4.7211198345

U(B) for J: 0.9 * -3.52152123152689 + 0.1 * 0 = -3.1693691084

Therefore, choose jump for A.

U(B) for S: 0.25 * -5.12098603551072 + 0.75 * -3.52152123152689 = -3.9213874325

U(B) for J: 0.9 * -5.12098603551072 + 0.1 * 0 = -4.608887432

Therefore, choose Shift for B.

2. Policy Iteration:

- We do the same procedure but start with the initial policy of Jumping first.

	k	A	B	C
Initial	**1**	-1	-2	0
Shift	**2**	-2.4	-3	0
Jump	**2**	-1.72	-3.44	0
Max	**2**	-1.72	-3.44	0
Shift	**3**	-3.408	-3.72	0
Jump	**3**	-2.2384	-4.4768	0
Max	**3**	-2.2384	-3.72	0
Shift	**4**	-3.67968	-4.08704	0
Jump	**4**	-2.611648	-4.6784	0
Max	**4**	-2.611648	-4.08704	0
Shift	**5**	-3.9745536	-4.3843968	0
Jump	**5**	-2.88038656	-4.9426688	0
Max	**5**	-2.88038656	-4.3843968	0
Shift	**6**	-4.206715392	-4.605111296	0
Jump	**6**	-3.0738783232	-5.156765696	0
Max	**6**	-3.0738783232	-4.605111296	0
Shift	**7**	-4.37784244224	-4.76534925312	0
Jump	**7**	-3.213192392704	-5.31568013312	0
Max	**7**	-3.213192392704	-4.76534925312	0
Shift	**8**	-4.5018480304128	-4.8809852862464	0
Jump	**8**	-3.31349852274688	-5.4310514622464	0
Max	**8**	-3.31349852274688	-4.8809852862464	0

Shift	**9**	-4.59129087629722	-4.96429617089741	0
Jump	**9**	-3.38571893637775	-5.51430940609741	0
Max	**9**	-3.38571893637775	-4.96429617089741	0
Shift	**10**	-4.655721489814	-5.02429059600613	0
Jump	**10**	-3.43771763419198	-5.57429324304614	0
Max	**10**	-3.43771763419198	-5.02429059600613	0
Shift	**11**	-4.70211788444208	-5.06748869971642	0
Jump	**11**	-3.47515669661823	-5.61748922912442	0
Max	**11**	-3.47515669661823	-5.06748869971642	0
Shift	**12**	-4.7355245591535	-5.09859175791422	0
Jump	**12**	-3.50211282156512	-5.64859186379582	0
Max	**12**	-3.50211282156512	-5.09859175791422	0
Shift	**13**	-4.75957761906156	-5.12098604452192	0
Jump	**13**	-3.52152123152689	-5.67098606569824	0
Max	**13**	-3.52152123152689	-5.12098604452192	0

Using 13 iterations: A = -3.52152123152689, B = -5.12098604452192, C = 0

U(A) for S: 0.25 * -3.52152123 + 0.75 * -5.120986 = -4.7211198

U(B) for J: 0.9 * -3.52152123 + 0.1 * 0 = -3.169369

Therefore, choose jump for A.

U(B) for S: 0.25 * -5.120986 + 0.75 * -3.52152123 = -3.921387

U(B) for J: 0.9 * -5.120986 + 0.1 * 0 = -4.608887

Therefore, choose Shift for B.

2. Probabilistic Context-Free Grammars
    1. To show which have nonzero probability of being generated as a complete VP, I'll show the calculation and see if it comes out to zero.
        (a) 0.5 * 0.5 * 0.6 * 0.2 * 0.2 * 0.2 $\neq$ 0. This equation is nonzero because you can keep adding the adverb and it just makes the prob smaller.

(b) 0.2 * 0.8 * 0. This sentence has zero because you cannot have the after “the” copula in this context-free grammar.

(c) 0.5 * 0.5 * 0. This equation has zero probability because “shoots the” can only be followed by a noun and this sentence does not have a noun following it.

2. First, we notice that the sentence is of the form copula adjective, therefore we use the probability 0.2. Then we multiply that by the probability that the copula “is” is chosen, 0.8. Then, multiply that by the probability of the adjective “well,” 0.5 twice. Therefore, we have… $0.2 * 0.8 * 0.5^2$ = 0.04
3. First, there is ambiguity because I could have done #2 in a different way. There is ambiguity in the fact that “is” could have been interpreted as a copula or a verb. Moreover, “well” could have been interpreted as a noun, adjective or an adverb. Because of all these options, the sentence can have many interpretations, and probability values.

3. Looking Back

1. Classical planning works well in static environments because classical planning is logic based. Furthermore, the paradigm works well in fully observable environment and perfect environments where there is low uncertainty. A benefit of classical planning is that it’s a deterministic in that you can predict exactly what it’s going to do. Markov decision processes work well in uncertain environments where things can possibly change. MDPs are also good for cyclic domain structures where decisions may need to be repeated numerous times. MDPs are quite fair as they are based on giving each decision an accurate probability of being chosen.
2. In HMMs, the entire model runs based off of the initial states. Probabilistic Context-free Grammars also for more of a structure to be defined so that the model of the data isn’t just linear as in HMMs. The PCFGs allow for more rules to be incorporated into the model. It also allows for the grammar to be nested and allow words that aren’t adjacent to be connected and affected by each other more directly. These rules in PCFGs are an overall more realistic model of how we

form sentences as the whole sentence must have a framework and our languages have grammars.