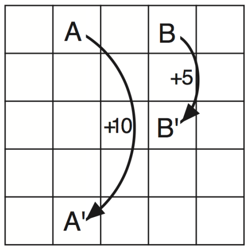
Advanced Machine Learning GR5242
Fall 2019

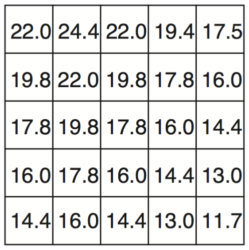
Homework 5

Due: Wednesday 04 December 4pm (for both sections of the class)
Homework submission: please submit your homework by publishing a notebook that cleanly displays your code, results and plots to pdf or html. You may wish to include another pdf file containing typeset or neatly scanned answers to the non-coding questions.

1. **Gridworld value** (60 points)
   In class we discussed Gridworld, its optimal policy $\pi_*(a|s)$, and state-value function $v_*(s)$:

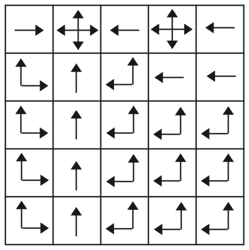


Gridworld



22.0	24.4	22.0	19.4	17.5
19.8	22.0	19.8	17.8	16.0
17.8	19.8	17.8	16.0	14.4
16.0	17.8	16.0	14.4	13.0
14.4	16.0	14.4	13.0	11.7

v_*



π_*

The discount factor for returns in this setup is $\gamma = 0.9$.

Questions:

(a) Recursively calculate $v_*(s = A)$, the value function at state A. Write a simple recursive function in python that follows the optimal policy and collects rewards. Terminate the recursion after $T = 200$ steps, starting from state A. Note that because the optimal policy is deterministic from $s = A$, the code need not be very complex (it doesn't need to know about Gridworld, just the intervals for accruing rewards 10.0, 0.0, 0.0,...). Produce your result to four decimal places. To check your work, note that this number should round to 24.4, as in the above figure.

(b) Mathematically derive the exact expression for the same quantity $v_*(s = A)$. Your answer should not involve any sums, but instead should exploit the following geometric series identity (for some $\beta \in [0, 1]$, which (hint) is not $\gamma = 0.9$). Confirm this answer against your previous answer.

$$\sum_{k=0}^{\infty} \beta^k = \frac{1}{1-\beta}$$

(c) Notice that the values $v_*(s) = \{22.0, 19.8, 17.8, 16.0, 14.4, 13.0\}$ are all repeated multiple times. Explain in words why this is the case, in terms of the optimal policy π_*. Relate these quantities mathematically to $v_*(s = A) = 24.4$.

2. **Text Classification with an RNN** (40 points)

   We will follow and modify Tensorflow's tutorial on text classification using an RNN, provided in the following link:

   `https://www.tensorflow.org/tutorials/text/text_classification_rnn`

   Questions:

   (a) Run the codes in the tutorial. You only need to run up to the first model. For speed, you can train the model with only 5 epochs. Post the accuracy and loss plots. Also report test loss and accuracy.

   (b) Make your own custom review(s) and test it. What prediction value do you get?

   (c) What does the embedding layer do in the model?

   (d) Modify the model in some way. You should detail in writing what you tried and what performance resulted.