

Final Exam

Part 1

Problem 1

I'm going to use iterative deepening search (IDS). This is because when searching on a large board, the state space could be more than 2^{35} (Because if there are 35 holes, each holes can have a peg or don't have a peg). It will take a huge amount of memory if we want to store the all the state. Instead, if we use IDS, we only need to store all the state on a single path, which is far less than 2^{35} .

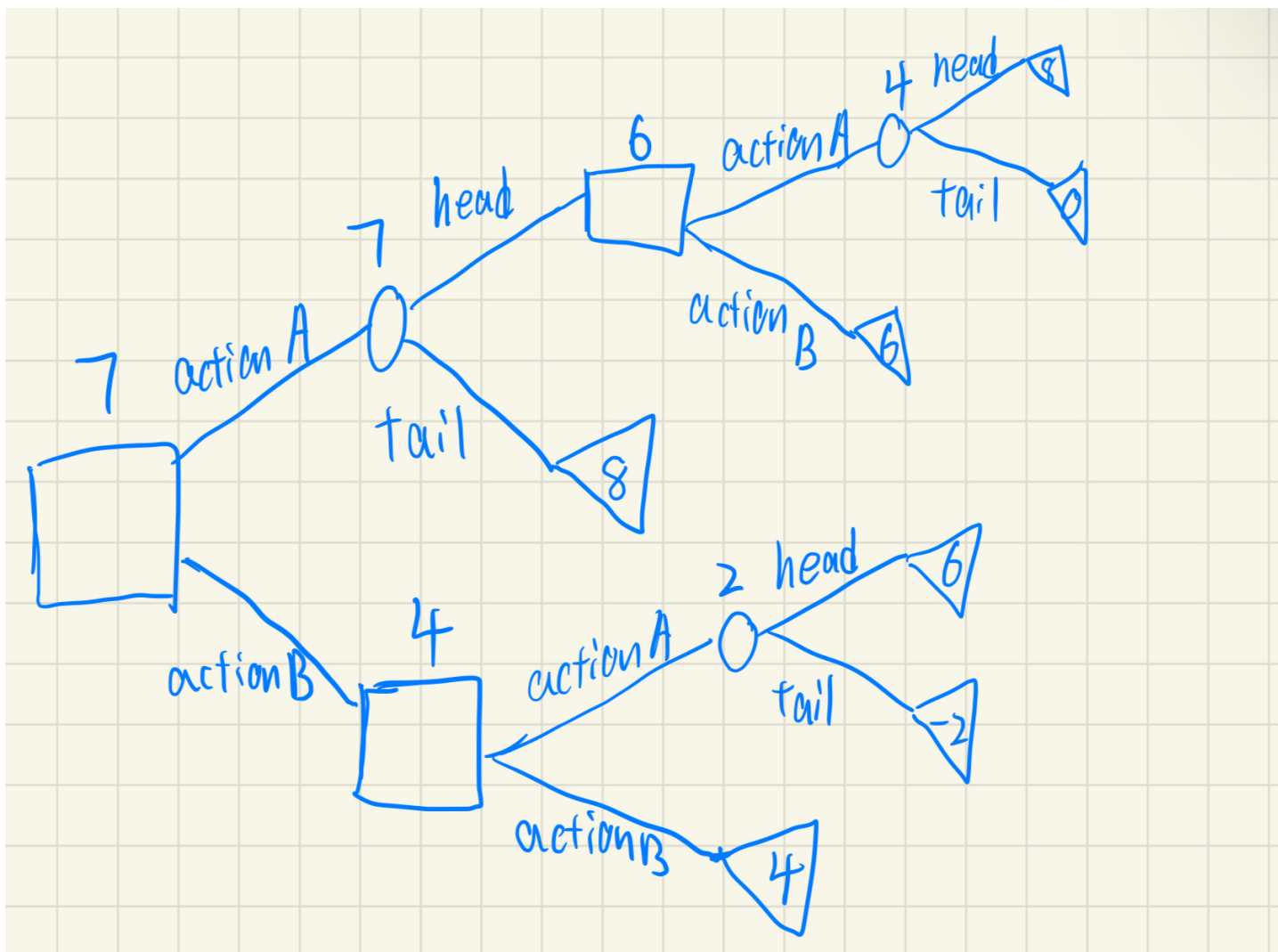
Problem 2

$$\text{A: } Put(3, 2) \implies (Peg(2, 2) \vee Peg(1, 2) \vee Peg(5, 2) \vee Peg(6, 2)) \wedge \neg Peg(3, 2)$$

$$\text{B: } Put(3, 2) \implies Peg(3, 3)$$

$$\text{C: } \neg Peg(6, 4) \wedge Peg(6, 5) \implies Jump(1, 3, 6, 4) \vee Jump(4, 5, 6, 4) \vee Put(6, 4)$$

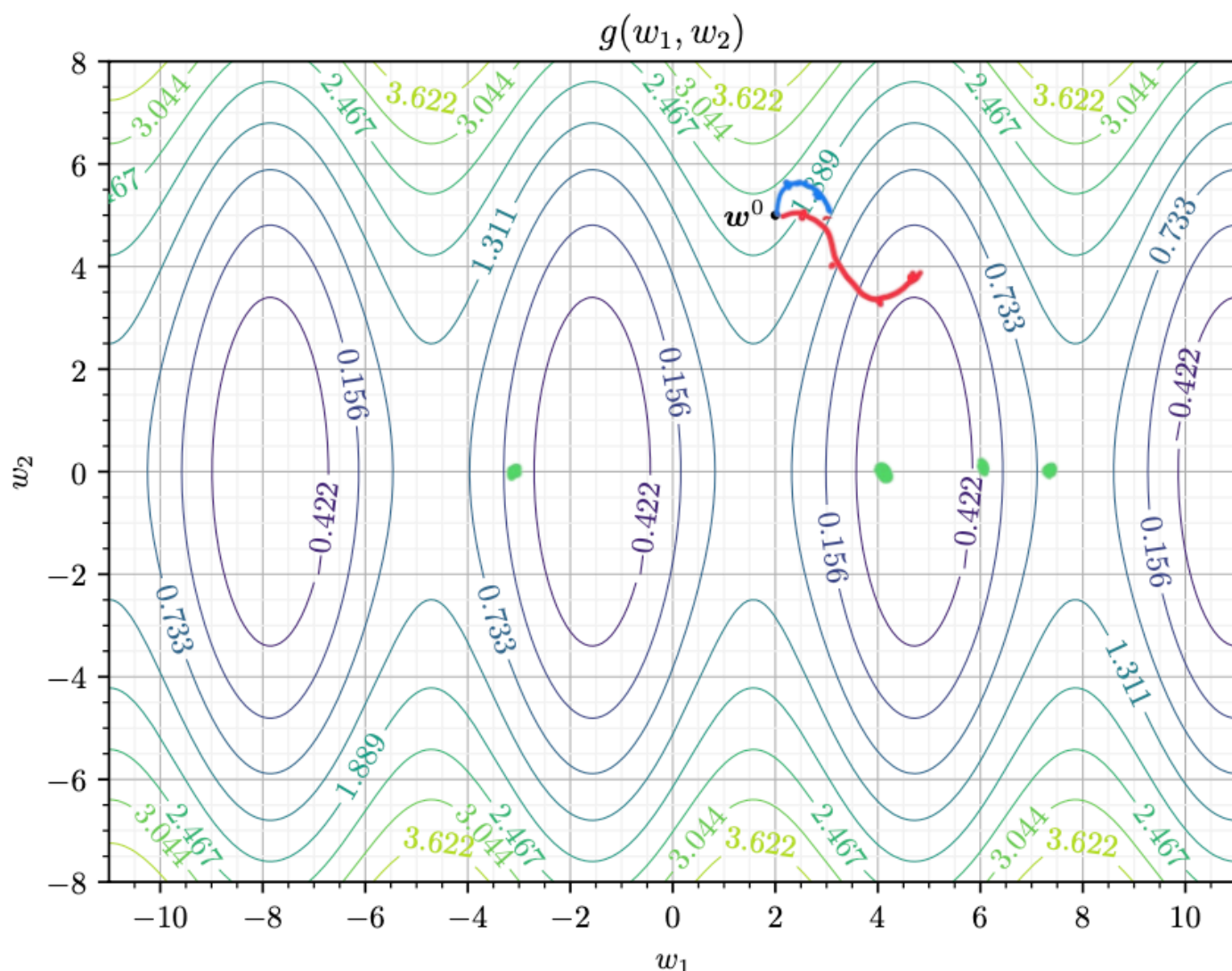
Problem 3



The optimal strategy is: Use action A first, if its tail, you win directly. If it's head, use action B (cost 2) to win. The expected return is $\frac{1}{2} \times 8 + \frac{1}{2} \times 6 = 7$

Problem 4

n	$\eta = 10^{-1}$ w^n	$\eta = 10^{-1}$ $g(w^n)$	$\eta = 1$ w^n	$\eta = 1$ $g(w^n)$	$\eta = 10$ w^n	$\eta = 10$ $g(w^n)$
0	$\langle 2, 5 \rangle$	2.159	$\langle 2, 5 \rangle$	2.159	$\langle 2, 5 \rangle$	2.159
1	$\langle 2.046, 4.95 \rangle$	2.116	$\langle 2.416, 4.5 \rangle$	1.67	$\langle 8.16, 0 \rangle$	-0.121
2	$\langle 2.0870, 4.9005 \rangle$	2.070	$\langle 3.164, 4.08 \rangle$	0.79	$\langle -3.76, 0 \rangle$	0.583
3	$\langle 2.1363, 4.8515 \rangle$	2.021	$\langle 4.164, 3.64 \rangle$	-0.18	$\langle 4.35, 0 \rangle$	-0.838
4	$\langle 2.1699, 4.803 \rangle$	1.968	$\langle 4.685, 3.28 \rangle$	-0.4	$\langle 7.83, 0 \rangle$	0.998
5	$\langle 2.249, 4.758 \rangle$	1.91	$\langle 4.712, 2.95 \rangle$	-0.564	$\langle 7.65, 0 \rangle$	0.981



The first one is going too slow(still at 1.9 after 5 steps) and the third one is going too fast(not stable). And the second case is descending in a good speed and gradually approach to convergence. I expect them to converge at -0.422.

Problem 5

The learning rate of 1 will means model will only "consider" the current information. And use it to correct all previous data using current data. But it could sometimes cause overshoot. When b approach infinity, it will no longer be a "soft" max but will become a "hard" max. In P , The option with highest probability will got a probability that close to 1 and others will be close to 0.

Problem 6

First, you should include all the feature in x . Also, you need to make sure x and z (testing set) have different data. Because you want to do testing in data that is different from training data. Or you will have overfitting problem. x dataset cannot be missing because it's the "feature" data, without x , we have nothing to train on.

Problem 7

We refer cost c be lossely reffered as spring because it is similar with spring because it change parameters toward value that reduce loss. Global loss it defined as the aggregation of loss in every sample in the dataset. It is not accident to have same symbol because L both denote lliely hood. And they both represent how good the model is. (prediction vs actual data).

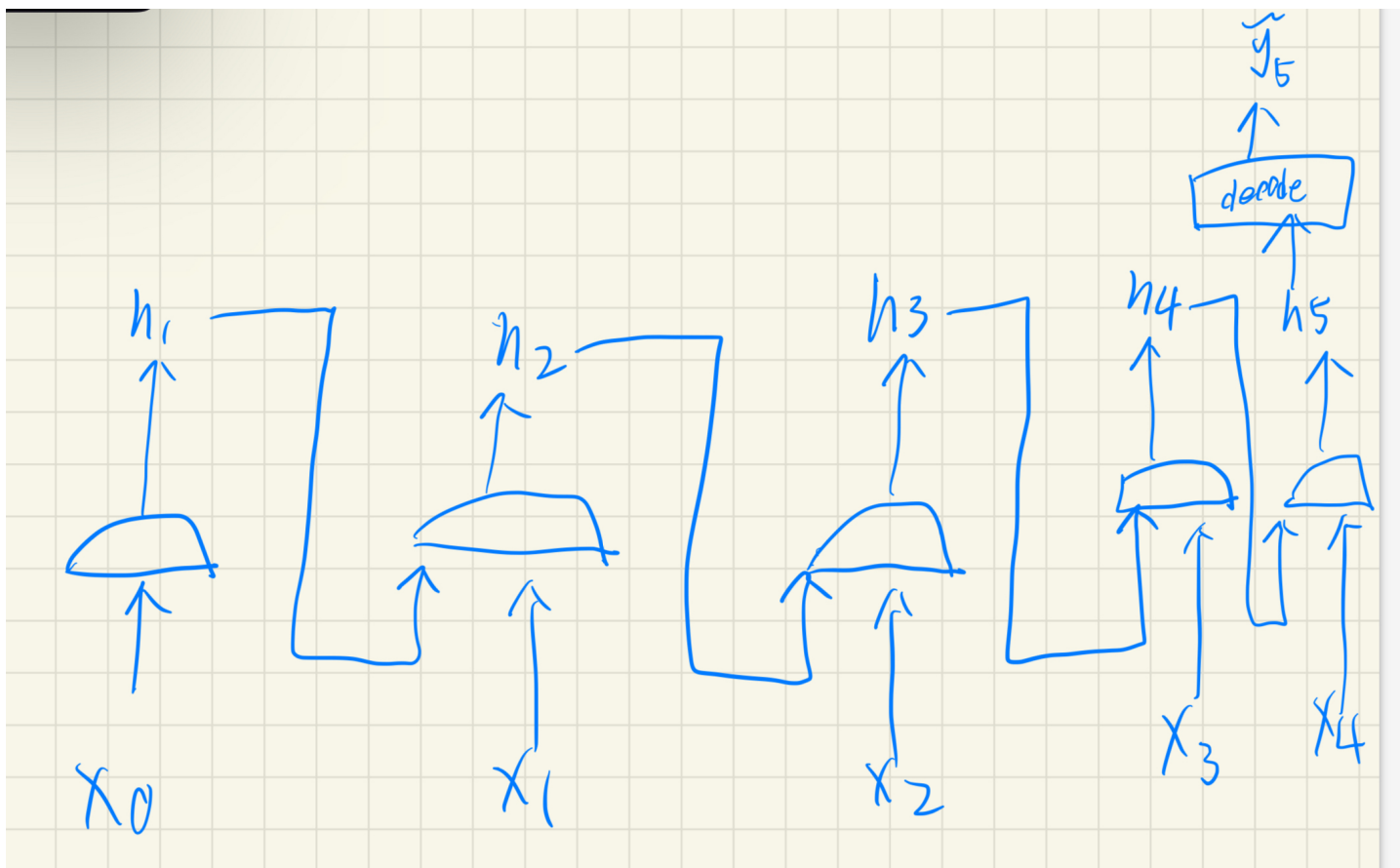
Problem 8

3 signals are spacial feature, local connectivity, share weight. local connectivity are connection between pixels and surrunding. Share weight meanse the weight used for processing input are same, the benefit is detecting same feature anywhere. And spacial feature means CNN can detect features from different part: higher layers can detect complex feature and lower layer can detech low-level feature, this is also the benefit.

Padding add layers of zeros and stride means how far filter move each time. For no padding and normal stride, stride is 1 and padding is 0. For pooling layer, the stride is equal to the filter's size. For instance, for 3×3 filter, stride is 3.

Problem 9





The unit delay will take the input, and wait for a "cycle" and then output the data it takes as input. it shild the state forward in time. And it helps run to remember past data, This is why rnn need to have such architecture because it can remember the earilier information and memory is crucial for tasks like NLP because you need to know the previous text of a paragraph.