Statistical Techniques in Robotics HW1

Zheyao Zhu

September 2020

Question 2.1

Yes, regret can be negative. If you don't assume reversibility, then there is a possibility that the learner would perform better than the best expert. In this case, the regret could be negative.

Question 2.2

We can construct a hypothesis space that includes all possible labels given any combination of the input feature vector. That way we can guarantee that no matter what the input feature vector is, we will have a perfect expert. In order to achieve that we need to have a hypothesis the size of K^N

Question 2.3

2.3.1

function Bayesian Having Algorithm(H)

$V^{(1)} = H$ initialize version space with all hypothesis

$W = \{w_i = p(e_i)\}_{i=1}^N$ initialize weight with prior

for t= 1.... T **do**:

receive(x^t)

$\hat{y_t}$ = **WeightedMajorityConsensus**$(V^t, x(t), W)$

receive $y^{(t)}$

$V^{t+1} = \{h \in V^t : h(x^t) = y^t\}$

2.3.2

We know that the sum of the prior of all the experts at time step 1 is 1

$$\Phi_{(1)=\sum_i p(e_i)=1}$$

At time step t, the sum of the weight of the reamining experts can be expressed as:

$$\Phi_t = \frac{1}{2}^M * \Phi_1 = \frac{1}{2}^M$$

We know that the lower bound of Φ_t is $p(e*)$ since $e*$ is the perfect expert that never makes mistake and thus would never get eliminated. We then have

$$p(e*) <= \frac{1}{2}^M$$
$$log(p(e*) <= Mlog(\frac{1}{2})$$
$$log(p(e*)) <= -M$$
$$M <= -log_2(p(e*))$$
$$M <= log_2(\frac{1}{p(e*)}$$

2.3.3

When the best expert has high weight, Bayesian halving works better.When there aren't a lot of experts, the regular halving algorithm works better.

Question 2.4

2.4.1
Since in this case, you can get rid of any and all experts that made mistake, at least half of the experts would get eliminated at the end of each iteration. Thus we can express the size of hypothesis at time t as:

$$|V^t| <= \frac{1}{2}|V^{t-1}|$$

We also know that since we assumed realizability, the lower-bound of the hypothesis space is 1. if total number of mistake made is M and the total number of expert in the beginning is E then

$$1 <= |V^t| <= \frac{1}{2}^M N$$
$$M <= log_2 N$$

2.4.2

pseudo code:

function paritally_observable_multi-class_classifier

$V^{(1)} = H$ initialize version space with all hypothesis

for t= 1.... T **do**:

receive(x^t)

$\hat{y_t} =$ **MajorityConsensus**$(V^t, x(t), W)$

If prediction incorrect:

$V^{t+1} = \{h \in V^t : h(x^t) \neq \hat{y}^t\}$ (eliminate the experts that predicted $\hat{y}^t$)

Mistake Bound Derivation:

In the partially observable case, the least amount of expert that can be eliminated each round is $\frac{|V^t|}{k}$, so we can write out:

$V^{t+1} = (1 - \frac{1}{k})^M |V^t|$

$1 <= V^t <= (1 - \frac{1}{k})^M N$

$0 <= Mlog(1 - \frac{1}{K}) + log_2 N$

$Mlog(1 - \frac{1}{K}) >= -log_2 N$

$-Mlog(1 - \frac{K}{K-1}) >= -log_2 N$

$M <= log(E - \frac{K}{K-1})$

Question 2.5

2.5.1

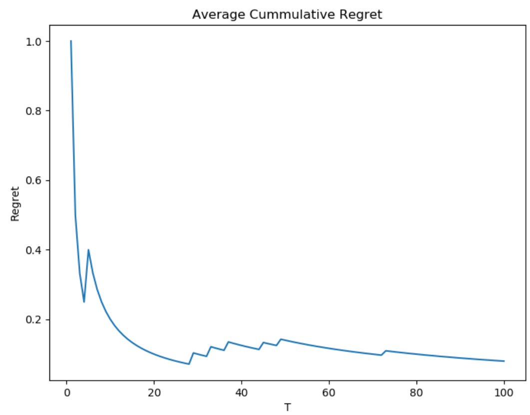
The adversary would pick the ground truth label that the minority of the experts choose. Since the learner would always pick what the majority of the experts choose, this strategy would always maximize the loss.

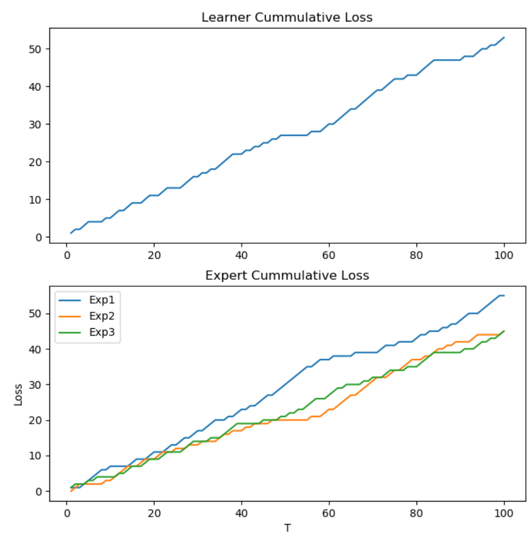
2.5.2

Given the adversary designed above, the deterministic weighted majority algorithm is guaranteed to make a mistake every time whereas the randomized weighed majority algorithm might make acorrect prediction sometimes. So the expected loss of the Randomized Weighted Majority Algorithm against the worst adversary is strictly less than the loss of WMA. In addition, we have also proven in class that the mistake bound of RWMA has a factor of 2 improvement over WMA.

Question 3.3 Weighted Majority Algorithm

Stochastic World



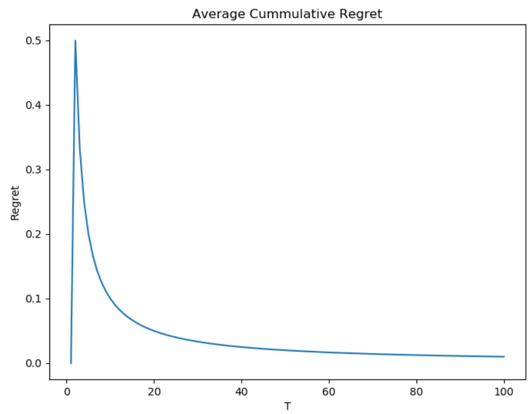


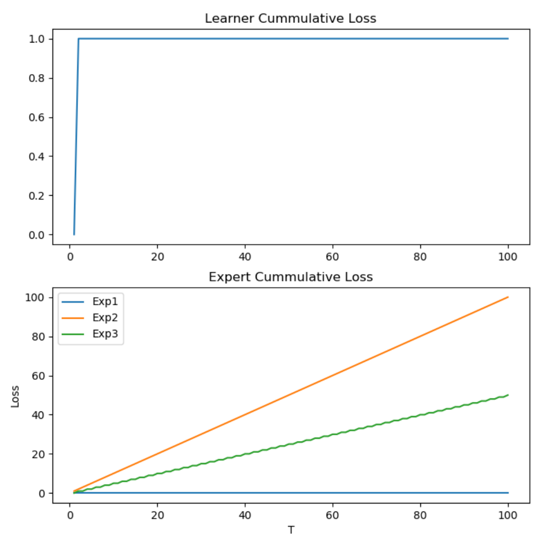
Comments: In this case, the ground truth labels are generated randomly so we can see that learner's loss and expert's loss continuously increase since none of the experts in this case can consistently make the right prediction. In the regret plot, we can see that the regret drops sharply in the beginning. That is because the expert is able to make the correct prediction sometimes so without the averaging of the cumulative loss in the very first step, there is a high regret, but as time step increases and the expert losses start to accumulate, the regret decreases. However, after the initial drop, since the world is stochastic in this case, neither experts or the learner could perform well and so as a result, the regret would remain at relatively the same height.

Comment on the effect of different T and different η

When I keep T the same as 100, I noticed that with a smaller η, the learner's regret converges much slower and the regret plots becomes less smooth. And when I set the η really small, even in the deterministic case, the learner's loss continuous to increase. This makes sense because the expert that's making the right decisions isn't getting weighted higher fast enough. Keeping the η the same and increasing T however would allow regret to converge in all three cases, and allow the learner's loss to reach a steady state in the deterministic world.

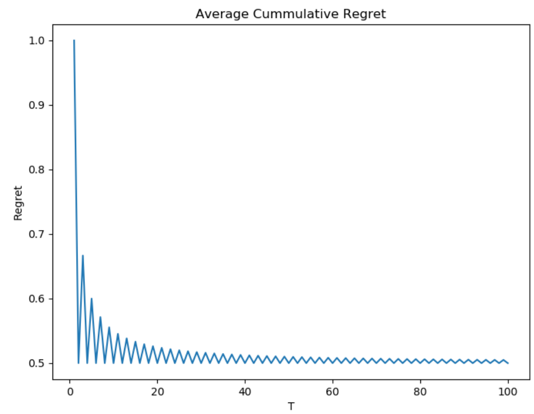
Deterministic World

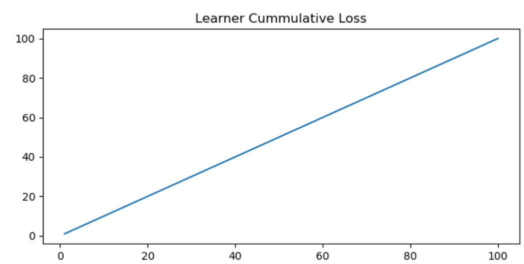


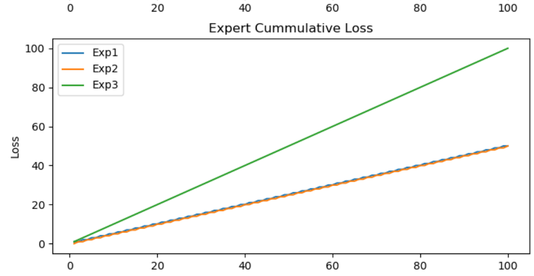


Comments:
In this case the world is deterministic and always gives the ground truth label of 1. We can see in the learner loss graph that the learner quickly learns and consistently makes prediction based on the suggestion of the expert 1 which always suggests 1. We can see that expert 1 has a commutative loss of 0 while expert 2 has the highest loss since it always predicts -1. Expert 3 performs better than expert2 since expert3 is correct every other time. The regret plot is smooth and drops continuously as the learner cumulative loss ceases to increase.

Adversarial World



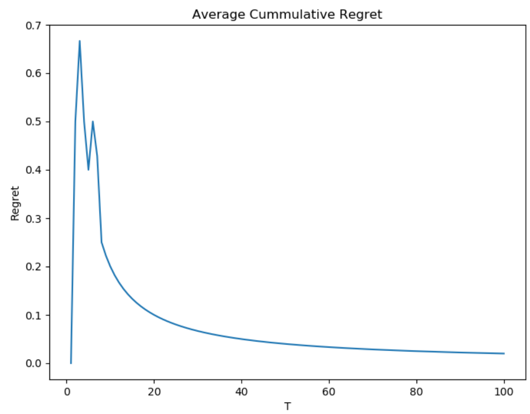


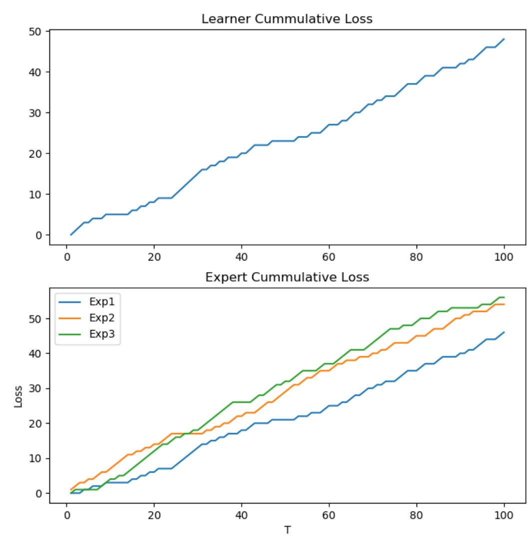


Comments:
In this case, the adversary always chooses what the minority of what the experts choose. It is the complete opposite strategy that the learner uses. As a result we can see that the learner cumulative loss continuously increases.

Question 3.4 Randomized Weighted Majority Algorithm

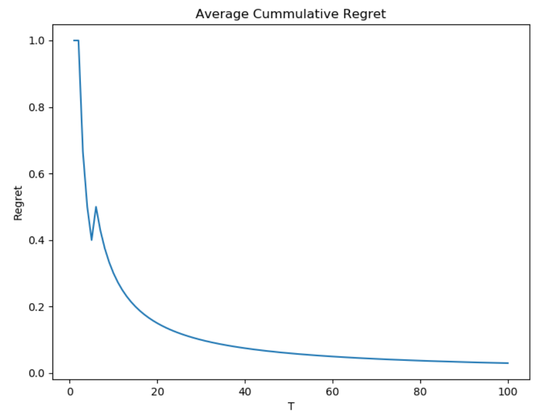
Stochastic World

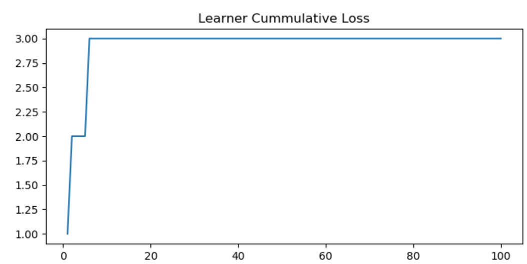


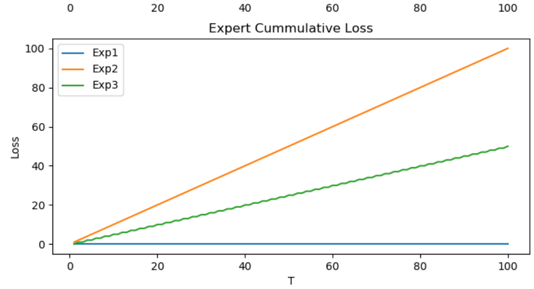


Comments:
Here we are seeing very similar behaviour as the stochastic case when using regular weighted majority algorithm. Since the world is random, the learner and the expert continues to perform poorly.

Deterministic World

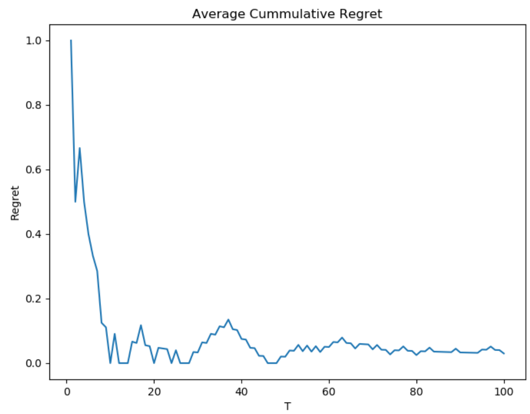


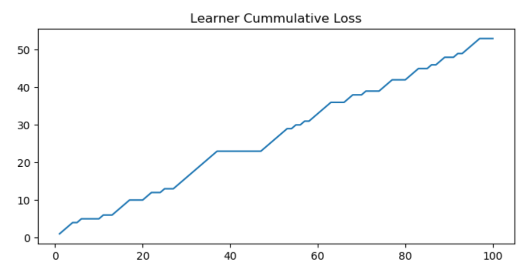


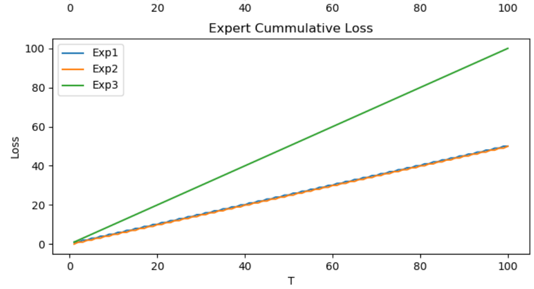


Comments:
Here we are seeing that the learner's loss is similar to the deterministic case when using weighted majority algorithm. However, the learner commutative loss is higher in this case. That's due to the randomness introduced in the algorithm. Even though expert1 always makes the correct prediction and gets weighted incrementally higher, initially, the mutlinomial random sampling might not pick the adivce of expert 1 every time. As the weight of expert gets higher and higher and the weight of other experts approach zero, the multinomial sampling eventually would always pick expert 1. As a result, we can see the learner cumulative loss flattens out.

Adversarial World





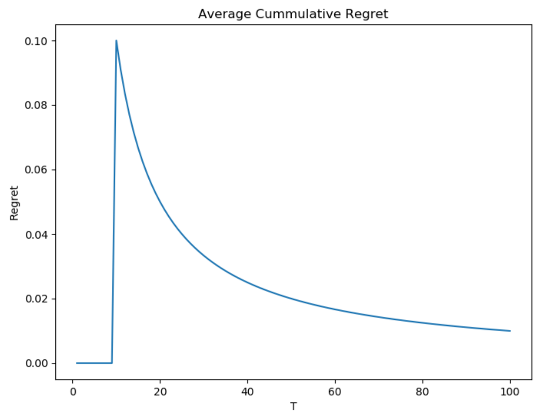


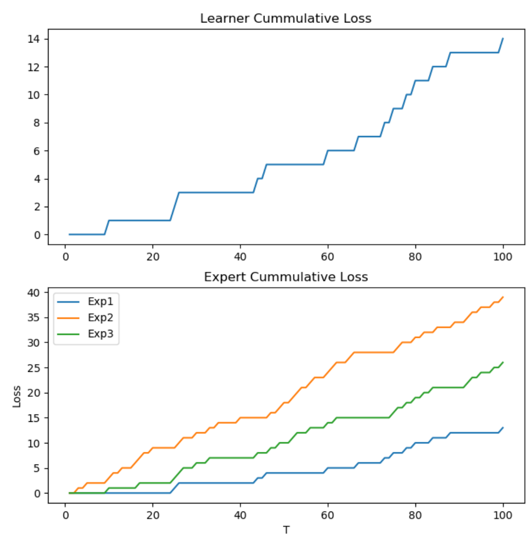
Comments:
We can see that the regret is a lot more random as a result of the randomness of the learner prediction as oppose to the oscillatory behaviour we see with the WMA. It's also worth noting that even though the learner cumulative loss continuously increases, the final cumulative loss is almost half of the learner's loss when using WMA in this world. This makes sense, because the learner is not always choosing what the majority of the experts choose which is opposite to what the adversary would choose.

Question 3.5 More Experts and Observation/Features

Implementation:
The three kinds of observation I added are weather, game location and player morale. The observations are generated randomly. I also added three new experts. The first expert predicts win if the weather is good. The second expert predicts win if it's a home game. The third expert predicts win if the weather is good and is a home game. The three additional worlds I added are designed as follows. In the first world, there is a 90% chance of winning if the weather is good and 90% chance of losing if the weather is bad. In the second world, there is a 90% chance of winning if it's home game and 90% chance of losing if it's not. In the third world, there is a 90% chance of winning if the weather is good and the game is a home game, else there is a 90% chance of losing. I used WMA to generate all plots in this section
.

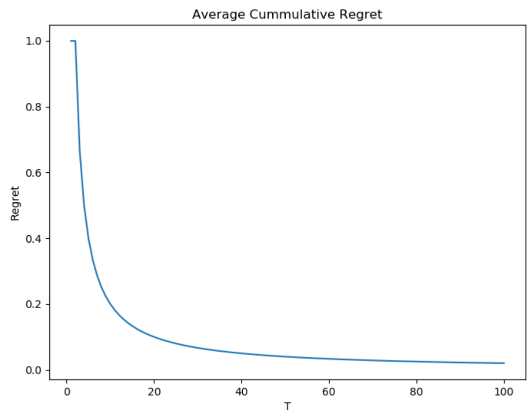
World 1: A world that favors weather.

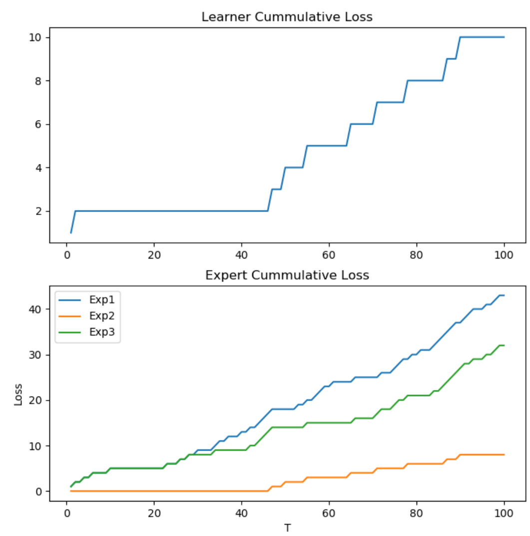




comment In this world, there is a 90% chance that the tartans will win if the weather is good. It makes sense that expert 1 would perform the best since expert 2 always predicts win if it's a home game. However, since the observation are generated randomly, we can see that even expert 2's commutative loss keeps on increasing.I also noticed that the loss of the learner is closely correlated to the loss of the expert 1. That makes sense because expert 1 would have a higher weight and the learner would pretty closely mimic the decisions by exper 1.

World 2: A world that favors game location.

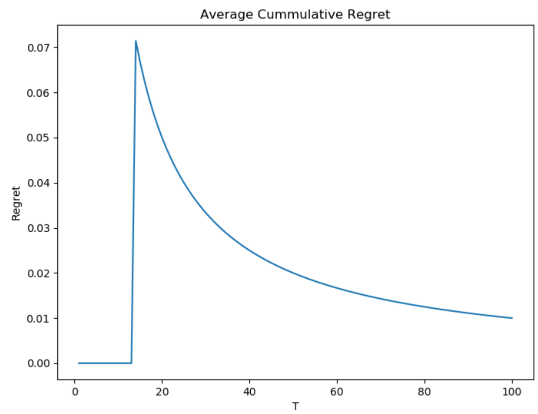


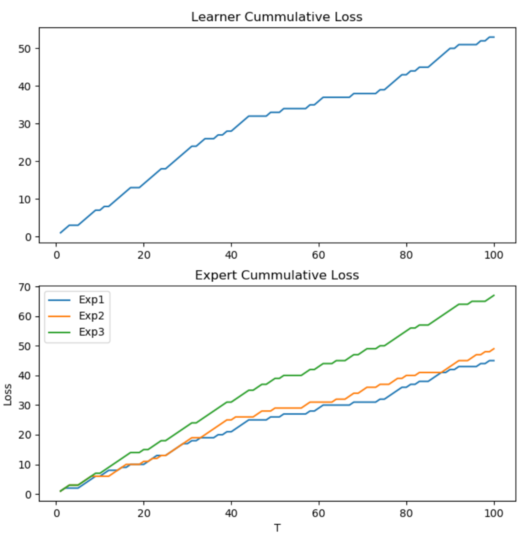


comment

In this world, there is a 90% chance that the tartans will win if it's a home game. It makes sense that expert 2 would perform the best since expert 1 always predicts win if the weather is good. However, since the observation are generated randomly, we can see that even expert 1's commutative loss keeps on increasing.

World 3: A world that favors weather and game location.





comment

In this world, there is a 90% chance that the tartans will win if it's a home game and the weather is good. We can see that both expert 1 and expert 2 have relatively similar performance and they both perform better than expert three. It's because the observations expert 1 and expert 2 can both increase the possibility of winning whereas the observation expert 3 uses is irrelevant in this world.