EE290S/CS194 Machine Learning for Sequential Decision Making Under Certainty Fall 2018

Lecture 6: September 11th 2018

Lecturer: Prof. Anant Sahai *Scribes: JungYoon Song, Nikunj Jain, Liam Purvis*

Note:

6.1 Recap of Multiplicative Weights

1. **Problem of Sequential Prediction:** $(X_1, \cdots, X_T) \in \{0,1\}^T$, we make no generative assumption and **instanteneous** loss vector $\ell_t = [\ell_{t,0}, \ell_{t,1}]$. We are getting this sequence online and the sequence can take $X_i \in \{0,1\}$. The key difficulty in this problem is that we have no generative assumption on the sequence, i.e. we don't have a probabilistic model. At each round, we have to predict something and the prediction itself is allowed to be randomized. In other words, we can choose it from some sort of probability distribution, W_t, denoting the probability of predicting a one in the next round.

2. Prediction at round t: distribution $\vec{W}_t$ with realization $\widehat{X}_t$ and total loss $\mathbf{L} = [L_{t,0}, L_{t,1}]$.

3. **Goal**: Minimize expected regret i.e. $\mathbf{E}[R_T] := \sum_{t=1}^{T} < \vec{W}_t, \vec{\mathbf{L}}_t > - \min_{x\in\{0,1\}} \ell_{T,x}$, where

$$< \vec{W}_t, \vec{\mathbf{L}}_t >= W_{t,0} \cdot \ell_{t,0} + W_{t,1} \cdot \ell_{t,1}.$$

   The intuition behind this loss function is that in the context of no generative assumption, it's useful to define a realistic benchmark. Our reference class is then how well you could have done if you have decided to predict 0 or 1 all the time, i.e., a constant predictor. Hence, we introduce expected regret, defined as the total loss our algorithm encounters on average minus the the best loss we could have possibly achieved from our reference class.

4. Last class: exponential with respect to update. $W_{t,x} \propto \exp(-\eta L_{t-1,x}) \Rightarrow$

$$\textbf{Theorem: } \mathbf{E}[R_T] \leq \sqrt{T \ln 2} \quad \text{for } \eta \sim \frac{\ln 2}{\sqrt{T}}.$$

   No matter what the sequence looks like, we have that regret is scaling at a sublinear scale. In today's lecture, we are going to derive this algorithm as the optimal solution to a particular kind of regularization, then view it as a response as noise perturbation.

6.2 Multiplicative Weights through Regularization

We first setup a reasonable function to optimize each round by mixing a loss function that is from our reference class, and then regularize it to incorporate our prior belief. Note, our prior is that we believe our predictions should involve randomness, and hence we will try to mathematically express that belief.

"Follow - the - Leader" like: at round t, we simply look at the loss we would have incurred if we have always predicted 0's or 1's. We know that this algorithm is not very good because it is behaving in a predictable manner and hence a sequence requiring a one-step memory (the alternating sequence) forces us to

always incur a loss.

$$\widehat{X}_t = \arg\min_{x\in\{0,1\}} L_{t-1,x}, \quad \widehat{W}_t = \arg\min_{\vec{W}} - < \widehat{W}, \widehat{L}_{t-1} >= \arg\min_{\vec{W}} -W_0 L_{t-1,0} - W_1 L_{t-1,1}.$$

"Pure Guessing": As an attempt at randomization, we could look at simply outputting a purely random guess, incorporating no information about the past sequence. This is a regularizer that is on the extreme side. This "purely guessing" means that you have completely randomized update and completely ignore information about the loss function. You simply have 0 and 1 occurring with probabilities of $\frac{1}{2}$ respectively, and $\widehat{W}_t = [\frac{1}{2}, \frac{1}{2}]$. What we truly want is to combine these two updates, FTL and Pure Guessing, in some meaningful way.

To do this, we want to be able to mathematically express the **"measure of randomness"** in our distribution, and balance increasing the randomness against decreasing the loss incurred.

For now, let's consider a new sequence of length n: $Y_1, \cdots, Y_n \underbrace{\sim}_{\text{IID}} \text{Bern}(p)$ - we assume a sequence generated as independent flips of a biased coin. Now, given such an assumption, clearly, **some sequences are more likely than others**. We are interested in the following two questions:

Question 1: What does a typical sequence looks like?

Question 2: What is the size of the set of typical sequences, and how does it compare to the size of the sample space?

To build intuition, notice that in any reasonable setup, a typical sequence is one which does not deviate "too far" from the average. Thus, we can measure the estimate we get from the sequence, $\widehat{p} := \frac{1}{n}\sum_{i=1}^n Y_i$. Then notice, $\mathbf{E}[\widehat{p}] = \frac{1}{n}np = p$.

So, we have that $\widehat{p}$ is an unbiased estimator of the true probability. Now, $\widehat{p}$ is itself a random variable, and we can use the Law of Large Numbers to come up with a stronger statement: $\widehat{p} = p$ with a high probability, that is,

$$\mathbf{P}[|\widehat{p} - p| \leq \epsilon] \geq 1 - 2\exp\left(-n\epsilon^2\right).$$

Thus, as our number of samples increases, the probability that our estimator is away from the true probability dies exponentially with our sample size.

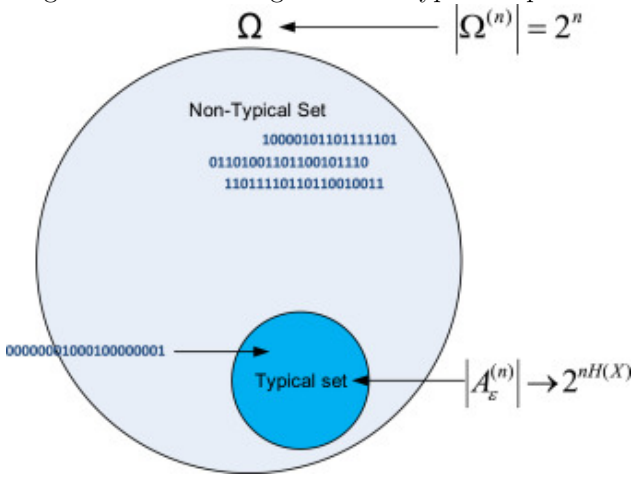
Now, we define a sequence to be typical (up to a factor of ϵ), if the estimator we derive from it is within ϵ of the true value. Then we define the typical set, $A_n^{(\epsilon)}$, as the set of **possible sequences**, which are typical.

$$A_n^{(\epsilon)} := \left\{(Y_1, \cdots, Y_n) \in \{0,1\}^n : p - \epsilon \leq \frac{1}{n}\sum_{i=1}^n Y_i \leq p + \epsilon\right\}$$

What's the size of this set?

Thus, any typical sequence will have between $np - \epsilon$ and $np + \epsilon$ ones. Then, it follows that the size of this set is exactly the number of ways to choose np ones out of n slots. Recall Stiriling's approximation formula: $n! \approx \sqrt{2\pi n}\left(\frac{n}{e}\right)^n$

Figure 6.1: Visualizing the set of typical sequences.



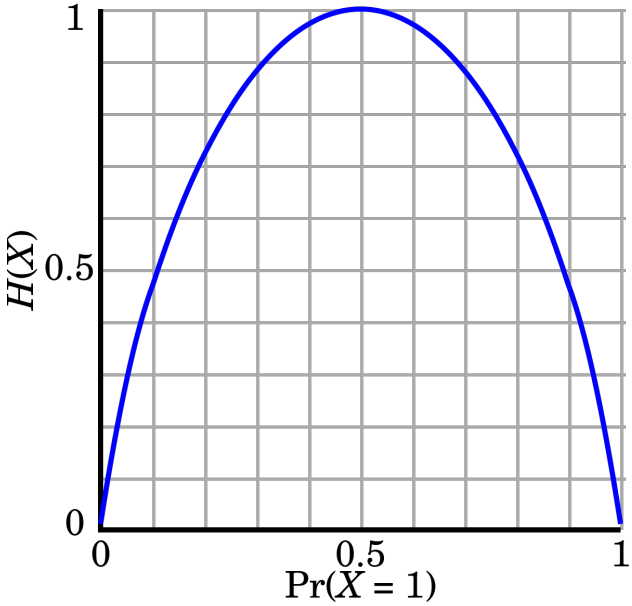
$$
\begin{aligned}
|A_n^{(\epsilon)}| = \sum_{k=n(p-\epsilon)}^{n(p+\epsilon)} \binom{n}{k} \approx \binom{n}{np} = \frac{n!}{(np)!(n(1-p))!} &\approx \frac{n^n e^{-n}}{(np)^{np} e^{-np} (n(1-p))^{n(1-p)} e^{-n(1-p)}} && \text{(Stirling's approximatio} \\
&= \frac{n^n e^{-n}}{n^{np} p^{np} e^{-np} n^{n(1-p)} (1-p)^{n(1-p)} e^{-n(1-p)}} \\
&= \frac{1}{p^{np}(1-p)^{n(1-p)}} \\
&= \left(\frac{1}{p}\right)^{np} \left(\frac{1}{1-p}\right)^{n(1-p)} \\
&= \left(2^{\ln_2(\frac{1}{p})}\right)^{np} \left(2^{\ln_2(\frac{1}{1-p})}\right)^{n(1-p)} \\
&= 2^{\left(np\ln_2\left(\frac{1}{p}\right) + n(1-p)\ln_2\left(\frac{1}{1-p}\right)\right)} \\
&= 2^{\left(nH(p)\right)}
\end{aligned}
$$

where we define $H(p) := p\log_2 \frac{1}{p} + (1-p)\log_2 \frac{1}{1-p}$. This is the standard definition of H, and it is commonly called the "entropy" function. Some final comments on the size of the typical set:

1. if $p = 1$, then $|A_n^{(\epsilon)}| = 2^{nH(1)} = 2^{n \cdot 0} = 1$. Note that this is a constant, as the only typical sequence is the one with all ones.
2. if $p = 0.5$, then $|A_n^{(\epsilon)}| = 2^{nH(1/2)} = 2^n$. Note that the difference between these two is exponential. The more the random the Bernoulli sequence is, the larger the size of the set we should expect to see. Hence, it is far more difficult for the adversary to fool you if the p value is close to 0.5.

We now build some intuition for the behavior of H.

Figure 6.2: Plot of the entropy v/s $p = P(X = 1)$.



Notice, entropy achieves its peak at $p = 0.5$, and is symmetric in p and $1 - p$. Thus, if a sequence is "highly random" (i.e., it has p values close to 0.5), the sequence has a high measure of entropy. When a sequence is almost constant (p close to 0 or 1), the sequence has low entropy. Hence, we can then view entropy as the mathematical measure of randomness we want, based on a derivation with a foundation in information theory through typical sets.

Entropy function: We will use this as a regularizer to express the randomness in our prediction vector. Intuition: a function that expresses randomness in prediction, such that the amount of randomness is maximized when the prediction is just purely guessing.

$$H(\vec{W}) = \sum_{x \in \{0,1\}} W_x \ln\left(\frac{1}{W_x}\right).$$

Observe that in the case of having only two outcomes,

$$\vec{W} = \begin{bmatrix} 1-p & p \end{bmatrix}, \quad H(\vec{W}) = H(p) = (1-p)\ln\left(\frac{1}{1-p}\right) + p\ln\left(\frac{1}{p}\right).$$

Through the plot, observe that this function is concave in p.

Idea: FTL + Entropy Regularization.

$$\vec{W}_t = \arg\max_{\vec{W}} \underbrace{- < \vec{W}, \vec{L}_{t-1} >}_{\text{loss objective function}} + H(\vec{W}) \cdot \frac{1}{\eta} \qquad \eta = \text{ regularizing paramter.}$$

Here, we will use entropy with the natural logarithm instead of logarithm in base 2 - this does not affect the

intuitive behavior, but makes analysis mildly easier.

$$\vec{W}_t = \begin{bmatrix} 1-p_t & p_t \end{bmatrix}, \quad W = \begin{bmatrix} 1-p & p \end{bmatrix}$$

$$p_t = \arg\max_{p\in[0,1]} \underbrace{-L_{t-1,0}(1-p) - L_{t-1,1}\cdot p}_{\text{linear function}} + \underbrace{\frac{1}{\eta}H(p)}_{\text{concave function}} \qquad *$$

$$\begin{aligned}
f(p) &= p\Big(L_{t-1,0} - L_{t-1,1}\Big) - L_{t-1,0} + p\ln\Big(\frac{1}{p}\Big)\frac{1}{\eta} + (1-p)\ln\Big(\frac{1}{1-p}\Big)\frac{1}{\eta} \\
&= p\Big(L_{t-1,0} - L_{t-1,1}\Big) - L_{t-1,0} - p\ln(p)\frac{1}{\eta} - (1-p)\ln(1-p)\frac{1}{\eta} \\
f'(p) &= \Big(L_{t-1,0} - L_{t-1,1}\Big) - (1+\ln p)\frac{1}{\eta} + \Big(1+\ln(1-p)\Big)\frac{1}{\eta} \\
&= \Big(L_{t-1,0} - L_{t-1,1}\Big) + \frac{1}{\eta}\ln\Big(\frac{1-p}{p}\Big) \\
f'(p_t) = 0 &\Rightarrow \ln\Big(\frac{1-p_t}{p_t}\Big) = \eta\Big(L_{t-1,0} - L_{t-1,1}\Big) \\
&\iff \frac{1-p_t}{p_t} = \exp\Big(\eta(L_{t-1,0} - L_{t-1,1}\Big) \\
&\iff p_t = \frac{\exp\Big(\eta(L_{t-1,0} - L_{t-1,1})\Big)}{\exp\Big(\eta(L_{t-1,0} - L_{t-1,1})\Big) + 1} = \frac{\exp\Big(-\eta L_{t-1,1}\Big)}{\exp\Big(-\eta L_{t-1,1}\Big) + \exp\Big(-\eta L_{t-1,0}\Big)}
\end{aligned}$$

With this setup, multiplicative weight updates emerge as the optimal choice.

For (*), entropy is a concave function in p and we have a linear loss function, so effectively we are maximizing a combination of a linear function and a concave function. This means that all we need to do is differentiate with respect to p and find a critical point, and concavity implies that this must be the globally optimal point.

Remark: The reason we want to go through this is to connect machine learning and optimization, and as we go further in the class, we will choose a continuous action space and work with a much bigger set than $\{0,1\}$. When generalizing though, the ideas of regularization will follow a similar pattern.

6.3 Multiplicative Weights through Perturbation

Firstly, perturbed, noise injection and smoothing all mean the same thing. Also, as we will see the plots towards the end of section, adding random noise will make the system stable.

$$\widehat{X}_t = \arg\min_{x\in\{0,1\}} \Big(L_{t-1,x} + \underbrace{N_{t,x}}_{\text{in some distribution}}\Big), \quad N_{t,x} \text{ is a random.}$$

Suppose $N_{t,x} \sim \text{Gumbel}(0, \frac{1}{\eta})$ IID with p.d.f of the distribution $f_{\text{Gumbel}}(x;\eta) = \eta\exp\Big(-\eta x^{\exp(-\eta x)}\Big)$. Suppose X_1, X_2 are IID Gumbel$(0, \frac{1}{\eta})$. Then $X_1 - X_2 \sim \text{Logistic}\Big(\frac{1}{\eta}\Big)$. Its CDF is given by

$$F_{\log}(x) = \mathbf{P}[X_1 - X_2 \le x] = \frac{1}{1+e^{-\eta x}}.$$

$$
\begin{aligned}
p_t = \mathbf{P}[\widehat{X}_t = 1] &= \mathbf{P}[L_{t-1,1} + N_{t,1} \leq L_{t-1,0} + N_{t,0}] \\
&= \mathbf{P}[N_{t,1} - N_{t,0} \leq L_{t-1,0} - L_{t-1,1}] \\
&= F_{\log}(L_{t-1,0} - L_{t-1,1}) = \frac{1}{1 + \exp\Big(- \eta(L_{t-1,0} - L_{t-1,1})\Big)} \\
&= \frac{\exp\Big(- \eta L_{t-1,1}\Big)}{\exp\Big(- \eta L_{t-1,1}\Big) + \exp\Big(- \eta L_{t-1.0}\Big)}
\end{aligned}
$$