COS 511: Theoretical Machine Learning

Lecturer: Rob Schapire
Scribe: Mihir Kulkarni

Lecture # 23
April 30, 2018

1 Recap

In the last lecture, we saw how to invest in stocks using online learning.

1.1 Setting

- There are N stocks.
- We let $w_t(i)$ denote the fraction of wealth in stock i at the start of day t. Hence, we have
$$\sum_{i=1}^{N} w_t(i) = 1$$
- We let $p_t(i)$ denote the Price relative, or price change in stock i.
$$p_t(i) = \frac{\text{price of stock } i \text{ at end of day } t}{\text{price of stock } i \text{ at beginning of day } t}$$
- We let S_t be the wealth at the start of day t.
- We have $S_1 = 1$.

Last time, we saw that

$$S_{t+1} = S_t(\mathbf{w}_t \cdot \mathbf{p}_t)$$

Unrolling the recurrence gives us

$$S_T = \prod_{t=1}^{T} (\mathbf{w}_t \cdot \mathbf{p}_t)$$

Note that maximising $S_T = \prod_{t=1}^{T}(\mathbf{w}_t \cdot \mathbf{p}_t)$ is equivalent to minimizing $\sum_{t=1}^{T} -\ln(\mathbf{w}_t \cdot \mathbf{p}_t)$.

Hence, we can formulate our problem as an online learning problem with the log-loss at step t given by $-\ln(\mathbf{w}_t \cdot \mathbf{p}_t)$. We wish to minimise this loss, and are faced with the following online learning problem:

for $t = 1 \dots T$

- Choose $\mathbf{w}_t$
- Observe $\mathbf{p}_t$
- Suffer loss $= -\ln(\mathbf{w}_t \cdot \mathbf{p}_t)$

As discussed in the previous lecture, we would like to perform this task without any statistical assumptions. We will proceed to do so using an algorithm derived from the Bayes algorithm, and analyse the regret bound with respect to the best stock in hindsight.

2 Using Bayes algorithm to do almost as well as the best stock

We will apply Bayes algorithm to obtain an investment strategy that does almost as well as the best stock. We first review Bayes algorithm. Note that in the description that follows, we are free to choose $\mathcal{X}$, the x_t's and the $\tilde{p}_{t,i}$'s, since they are not part of the problem setting.

2.1 Algorithm

for $t = 1 \dots T$

- Expert i produces distribution $\tilde{p}_{t,i}$ over $\mathcal{X}$
- Learner maintains weights $\tilde{w}_{t,i}$, and predicts $\tilde{q}_t$, where $\tilde{q}_t(x) = \sum_i \tilde{w}_{t,i}\tilde{p}_{t,i}(x)$
- Observe $x_t \in \mathcal{X}$
- Loss suffered is $-\ln \tilde{q}_t(x_t)$

By Bayes algorithm regret bounds, we have

$$-\sum_{t=1}^{T} \ln \tilde{q}_t(x_t) \leq \min_i \sum_{t=1}^{T} -\ln \tilde{p}_{t,i}(x_t) + \ln N$$

We leverage these bounds by defining $\mathcal{X}, x_t$'s and $\tilde{p}_t(i)$'s appropriately. Suppose we know ahead of time that

$$c \geq \max_{t,i} p_t(i)$$

This means that each stock grows by a factor of at most c in each round. It is quite reasonable to assume that such a c must exist. We now choose $\mathcal{X}$. Let $\mathcal{X} = \{0, 1\}$.

Now, we define the predictions of experts as

$$\tilde{p}_{t,i}(1) = \frac{p_t(i)}{c} \leq 1$$
$$\tilde{p}_{t,i}(0) = 1 - \frac{p_t(i)}{c}$$

Since our regret bounds are independent of statistical assumptions, we can choose the x_t's as we wish. We choose $x_t = 1\ \forall t$. The Bayes algorithm computes $\tilde{w}_{t,i}$ for each round, and we will use that to decide the proportions used in the stocks. In other words, we let $\forall\ i\ w_t(i) = \tilde{w}_{t,i}$ be the fraction of wealth invested in stock i.

Plugging in the definitons, we have that

$$\begin{aligned}
\tilde{q}_t(x_t) &= \tilde{q}_t(1) \\
&= \sum_i \tilde{w}_{t,i}\tilde{p}_{t,i}(1) \\
&= \sum_i w_t(i)\frac{p_t(i)}{c} \\
&= \frac{\mathbf{w}_t \cdot \mathbf{p}_t}{c}
\end{aligned}$$

Having done our calculation, we can check the regret bounds! These are

$$\begin{aligned}
-\sum_{t=1}^{T} \ln\left(\frac{\mathbf{w}_t \cdot \mathbf{p}_t}{c}\right) &= -\sum_{t=1}^{T} \ln \tilde{q}_t(x_t) \\
&\leq \min_i -\sum_{t=1}^{T} \ln \tilde{p}_{t,i}(x_t) + \ln N \\
&= \min_i -\sum_{t=1}^{T} \ln \frac{p_t(i)}{c} + \ln N
\end{aligned}$$

Hence, we have

$$\begin{aligned}
&-\sum_{t=1}^{T} \ln \mathbf{w}_t \cdot \mathbf{p}_t + T \ln c \leq \min_i -\sum_{t=1}^{T} \ln p_t(i) + T \ln c + \ln N \\
&\implies -\sum_{t=1}^{T} \ln \mathbf{w}_t \cdot \mathbf{p}_t \leq \min_i -\sum_{t=1}^{T} \ln p_t(i) + \ln N \\
&\implies -\ln(\text{wealth of learner}) \leq -\ln(\text{wealth of best stock}) + \ln N \\
&\implies \ln(\text{wealth of learner}) \geq \ln(\text{wealth of best stock}) - \ln N \\
&\implies \text{wealth of learner} \geq \frac{\text{wealth of best stock}}{N} \\
&\implies \underbrace{(\text{wealth of learner})^{\frac{1}{T}}}_{\text{Average daily rate of growth of learner}} \geq \underbrace{\left(\frac{1}{N}\right)^{\frac{1}{T}}}_{\text{Goes to 1 as } T\to\infty} \times \underbrace{(\text{wealth of best stock})^{\frac{1}{T}}}_{\text{Average daily rate of growth of best stock}}
\end{aligned}$$

Since the multiplicative factor $\left(\frac{1}{N}\right)^{\frac{1}{T}}$ converges to 1 as $T \to \infty$, we have that in the limit, the learner achieves the rate of growth achieved by the best stock!

It may seem that we have done something complicated to achieve these bounds. However, if we unravel the strategy we just derived, it is essentially the following

"Leave all money in N stocks, divided evenly, on day 1, and do not buy or sell anything after that."

This strategy is also called "Buy and Hold". Note that our analysis for this strategy can also been seen rather trivially: since $\frac{1}{N}$ of our wealth was invested in the best stock, our final wealth will be at least $\frac{1}{N}$ of the wealth attained by that best stock, even if all the other money we invested in other stocks is lost.

What are the other strategies we could use? One approach is to try to do as well as the best shifting sequence of experts. However, we consider now a different approach.

3 Constant Rebalanced Portfolios: CRP

A Constant Rebalanced Portfolio, or CRP, is a portfolio where the following strategy is used:

"Decide proportions for rebalancing ahead of time. Then, rebalance portfolios at the end of each day according to the proportion decided."

For instance, given 3 stocks, we might decide ahead of time to put 50% of our wealth in stock 1, and 25% in each of stocks 2 and 3. Then at the end of each period, after each of the stocks have gone up or down, we rebalance our holdings according to these proportions. So if we begin with \$100, we put \$50 in stock 1 and \$25 each in stocks 2 and 3 on day 1. Say stock 1 halves in value, 2 triples, and stock 3 quadruples, giving us a total wealth of $\$25 + \$75 + \$100 = \$200$ at the end of day 1. Then, at the start of day 2, we rebalance the portfolio according to the initial ratio, and assign a wealth of \$100 to stock 1, and \$50 each to stocks 2 and 3 before the beginning of day 2. In this manner, the rebalancing strategy is implemented at the end of each day.

We will now lead up to an investment strategy that does almost as well as the best CRP, rather than just the best stock.

3.1 Example

Imagine that we have two stocks. One does nothing, staying flat throughout. The other alternates between doubling and halving at the end of alternate days. We now show how implementing a CRP can be highly valuable, even when the underlying stocks are clearly not increasing in value.

Recall that the price relative is the ratio of the stock price at the end of the day to that at the beginning. Note that the first stock remains priced at a constant, and has a price relative of 1. On the other hand, the second stock alternately halves and doubles, and has its price relative oscillate between $\frac{1}{2}$ and 2.

Table 1: Price

	Stock 1 price	Stock 2 price
Start of day 1	1	1
Start of day 2	1	$\frac{1}{2}$
Start of day 3	1	1
Start of day 4	1	$\frac{1}{2}$
Start of day 5	1	1
$\vdots$	$\vdots$	$\vdots$
Start of day N	1	1

Table 2: Price relative

	Stock 1 price relative	Stock 2 price relative
Day 1	1	$\frac{1}{2}$
Day 2	1	2
Day 3	1	$\frac{1}{2}$
Day 4	1	2
Day 5	1	$\frac{1}{2}$
$\vdots$	$\vdots$	$\vdots$
Day N	1	$\frac{1}{2}$

3.2 Analysing Uniform CRP

Recall the Buy and Hold strategy. With this strategy, we do not make any money at the end for the stocks specified above: we either lose money or stay at the same amount of wealth, depending on the day.

Consider using the uniform CRP strategy, which rebalances wealth equally among the N stocks in the portfolio at the start of each day. Let us analyse the wealth S_t at the end of each day t. We start off with a total wealth of \$1, uniformly balanced, so that $\$\frac{1}{2}$ is invested in each stock.

1. $S_1 = 1$ since both stocks do not change in value during day 1.

2. At the end of day 2, stock 1 does not change in value, while stock 2 halves in value. Hence, we have overall wealth
$$\begin{aligned} S_2 &= S_1 \left(\frac{1}{2} \cdot 1 + \frac{1}{2} \cdot \frac{1}{2} \right) \\ &= S_1 \frac{3}{4} \end{aligned}$$

3. At the start of day 3, we rebalance the total wealth S_2 across both stocks. Hence, we invest $S_2 \frac{1}{2}$ in each stock. At the end of this day, stock 1 remains unchanged while stock 2 doubles in value. Hence, at the end of day 3, we have overall wealth

$$
\begin{aligned}
S_3 &= S_2\left(\frac{1}{2}\cdot 1 + \frac{1}{2}\cdot 2\right) \\
&= S_2\frac{3}{2} \\
&= S_1\frac{9}{8}
\end{aligned}
$$

Since the price-relative for stock 2 oscillates with a period of 2, and that of stock 1 is always 1, we have

$$S_{t+2} = S_t\frac{9}{8}$$

Hence, our wealth increases by a constant factor every two rounds, and this is therefore clearly a better strategy than Buy and Hold for this setting! We now move beyond the uniform distribution and consider CRPs for general distributions.

4 General CRP

We can define a general CRP by the policy

$$\mathbf{b} = \langle b_1, \ldots, b_N \rangle$$

with

$$b_i \geq 0 \ \forall i, \ \sum_{i=1}^{N} b_i = 1$$

such that $\mathbf{b}$ defines how we want to redistribute stocks at the start of each round. That is to say,

$$\forall t \ \mathbf{w}_t = \mathbf{b}$$

Which $\mathbf{b}$ should we use? We would like to learn the optimal CRP, or do almost as well as it does. To solve this problem, we use the Universal Portfolio algorithm.

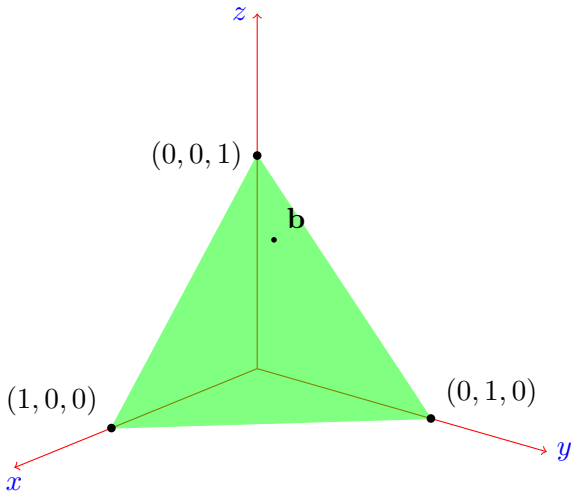
5 Universal Portfolio

The basic idea is to do as well as a whole "universe" of strategies. We would like to do almost as well as the best CRP. To do this, we will implement Buy and Hold over the space of all CRPs, and divide money across all CRPs infinitesimally. The resulting algorithm is called the Universal Portfolio algorithm. This is one example of using a continuous space for what is a discrete problem over N stocks. It is also an example of applying an algorithm defined for a finite set of experts and extending it to a continuous (uncountably infinite) space of experts.

Note that each CRP corresponds to a distribution on the stocks. Consider the probablilty simplex Δ, which is defined as

$$\Delta = \left\{ \mathbf{b} \in \mathbb{R}^N : b_i \geq 0 \ \forall i, \ \sum_{=1}^{N} b_i = 1 \right\}$$

Note that this set is $N-1$ dimensional, since it lies in $\mathbb{R}^N$ and has one equality constraint.



For example, the simplex in $\mathbb{R}^3$ is the green triangle shown above.

Consider any distribution $\mathbf{b}$ over all the stocks, which lies in Δ. For this one CRP, we want to calculate its wealth. Let us assume the initial (infinitesimal) wealth assigned to $\mathbf{b}$ is $d\mu(\mathbf{b})$. Then, we have the wealth invested in $\mathbf{b}$ at the beginning of the t^{th} day as

$$\prod_{s=1}^{t-1}(\mathbf{b}\cdot\mathbf{p}_s)d\mu(\mathbf{b})$$

Hence, we have that the total wealth at the start of day t is

$$\int_{\mathbf{b}\in\Delta}\prod_{s=1}^{t-1}(\mathbf{b}\cdot\mathbf{p}_s)d\mu(\mathbf{b})$$

For each CRP $\mathbf{b}$, the fraction of wealth in stock i is b_i. Hence, the total wealth invested in stock i at the start of day t is

$$\int_{\mathbf{b}\in\Delta} b_i\prod_{s=1}^{t-1}(\mathbf{b}\cdot\mathbf{p}_s)d\mu(\mathbf{b})$$

We would like to know how much to invest in stock i according to the universal portfolio's strategy. Hence, for this algorithm, we set

$$w_t(i)=\frac{\int_{\mathbf{b}\in\Delta} b_i\prod_{s=1}^{t-1}(\mathbf{b}\cdot\mathbf{p}_s)d\mu(\mathbf{b})}{\int_{\mathbf{b}\in\Delta}\prod_{s=1}^{t-1}(\mathbf{b}\cdot\mathbf{p}_s)d\mu(\mathbf{b})}$$

Hence, given $w_t(i)$, we can implement Buy and Hold on the N stocks to implement the universal portfolio selection algorithm. In practice, the integrals can be hard to compute, and discretised versions are used.

5.1 Regret bounds

We now look at a bound that tells us that the rate of growth of the universal portfolio algorithm will be almost as good as the rate of the best CRP.

Theorem 1. *At the end of T rounds,*

$$\textit{(Wealth of Universal Portfolio algorithm)} \geq \textit{(Wealth of best CRP)} \times \left(\frac{1}{T+1}\right)^{N-1}$$

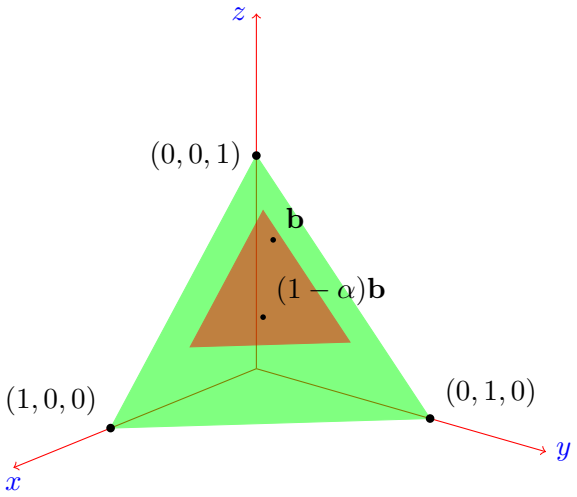
Further, it can be shown that this regret bound is almost the best possible in a certain sense. We prove a slightly weaker version of this theorem, namely that

$$\text{(Wealth of Universal Portfolio algorithm)} \geq \text{(Wealth of best CRP)} \times \frac{1}{e(T+1)^{N-1}}$$

Proof. Let the best CRP be $\mathbf{b}^*$. We will analyse a subset of Δ containing $\mathbf{b}^*$, called a neighbourhood of $\mathbf{b}^*$, that is reasonably large. We will then argue that all CRPs close to $\mathbf{b}^*$ achieve wealth approximately equal to that achieved by $\mathbf{b}^*$. Since $\mathbf{b}^*$ is the best CRP, and the neighbourhood is not too small, we will be able to show that the total wealth generated by the neighbourhood is large.

We define this neighbourhood to be all CRPs which are like a slightly scaled-down version of $\mathbf{b}^*$ + another nearby CRP. Formally, for a fixed $\alpha \in [0, 1]$ (that we will define later), we define

$$\eta(\mathbf{b}^*) = \{(1-\alpha)\mathbf{b}^* + \alpha\mathbf{z} \mid \mathbf{z} \in \Delta\}$$



In the figure, $\eta(\mathbf{b}^*)$ is the red triangle centered at $(1-\alpha)\mathbf{b}^*$.

Note that for any $\mathbf{b} \in \eta(\mathbf{b}^*)$ we have

$$\begin{aligned}
&\mathbf{b} = (1-\alpha)\mathbf{b}^* + \alpha\mathbf{z} \\
\implies &\mathbf{b}\cdot\mathbf{p}_t = (1-\alpha)\mathbf{b}^*.\mathbf{p}_t + \alpha\underbrace{\mathbf{z}.\mathbf{p}_t}_{\geq 0} \\
&\qquad \text{(Taking dot product of both sides with } \mathbf{p}_t\text{)} \\
\implies &\mathbf{b}\cdot\mathbf{p}_t \geq (1-\alpha)\mathbf{b}^*\cdot\mathbf{p}_t \\
&\qquad \text{(since } \mathbf{z}, \mathbf{p}_t \geq 0\text{)} \\
\implies &\prod_{t=1}^{T}\mathbf{b}\cdot\mathbf{p}_t \geq (1-\alpha)^T\prod_{t=1}^{T}\mathbf{b}^*\cdot\mathbf{p}_t \\
&\qquad \text{(Taking the product across all } T \text{ rounds)} \\
\implies &\text{Wealth for CRP } \mathbf{b} \geq (1-\alpha)^T(\text{Wealth for CRP } \mathbf{b}^*)
\end{aligned}$$

We now check the size of $\eta(\mathbf{b}^*)$

$$\begin{aligned}
\text{Vol}(\eta(\mathbf{b}^*)) &= \text{Vol}(\{(1-\alpha)\mathbf{b}^* + \alpha\mathbf{z} | \mathbf{z} \in \Delta\}) \\
&= \text{Vol}(\{\alpha\mathbf{z} | \mathbf{z} \in \Delta\}) \\
&\text{(Since translating the origin does not change the volume)} \\
&= \alpha^{N-1}\text{Vol}(\Delta) \\
&\text{(Since } \Delta \text{ is } N-1 \text{ dimensional, and scaling each} \\
&\text{length by } \alpha \text{ therefore scales the total volume by } \alpha^{N-1}\text{)}
\end{aligned}$$

Hence, we have that α^{N-1} fraction of the portfolio performs at least $(1-\alpha)^T$ as well as the best portfolio. Since the total wealth of the universal portfolio algorithm is at least as much as this portion, we have

$$\text{(Wealth of Universal Portfolio algorithm)} \geq \alpha^{N-1}(1-\alpha)^T(\text{Wealth of best CRP})$$

Plugging in $\alpha = \frac{1}{T+1} \in [0,1]$ gives us the bound

$$\text{(Wealth of Universal Portfolio algorithm)} \geq \left(\frac{1}{T+1}\right)^{N-1}\left(1-\frac{1}{T+1}\right)^T(\text{Wealth of best CRP})$$

If we can show that

$$\left(1-\frac{1}{T+1}\right)^T \geq \frac{1}{e}$$

this will give us

$$\text{(Wealth of Universal Portfolio algorithm)} \geq \text{(Wealth of best CRP)} \times \frac{1}{e(T+1)^{N-1}}$$

as required. Rearranging, this is equivalent to showing

$$\left(\frac{1}{1-\frac{1}{T+1}}\right)^T \leq e$$

Simplifying the left hand side, this is the same as

$$\left(1+\frac{1}{T}\right)^T \leq e$$

which is true since $1 + x \leq e^x$ for all x.

□

We now move on to an introduction to the next topic, game theory and its connections to machine learning.

6 Introduction to Game Theory

Games are models of interactions between agents. Since a lot of machine learning problems involve agents "learning" from repeated interactions with the environment, it is very natural for connections to exist between machine learning and the theory of games. But what is a game?

We first look at a simple model called a matrix game. Here, we have a row player, Mindy, and a column player Max. Simultaneously, Mindy, the row player, chooses a row i, and Max, the column player, chooses a column j. We have a matrix M such that the loss for Mindy is $M(i, j)$, which is the same as the gain or payoff to Max. For example, we consider the rock-paper-scissors game, with loss 0 for a win, 1 for a loss, and $\frac{1}{2}$ for a tie. The loss matrix M for Mindy is then

		Max		
		R	P	S
Mindy	R	$\frac{1}{2}$	1	0
	P	0	$\frac{1}{2}$	1
	S	1	0	$\frac{1}{2}$

Games of this type are called two-person, zero-sum games since we assume that Max's payoff is always equal to Mindy's loss. It can be argued that even larger, ordinary, zero-sum games can be put in this form.

In the next lecture, we will analyse two-player zero-sum games and consider an online learning algorithm for a player to achieve loss close to that of the best strategy over multiple rounds.