The Duality Theorem, the Maxmin Theorem, and Multiplicative Updates
CS170, Fall 2016

October 26, 2016

1 Introduction

The Duality Theorem of linear programming, proved by Dantzig in 1947 states that

> *If the primal and the dual are both feasible and bounded, then their optimum values coincide.*

On the other hand, von Neumann's the Maxmin Theorem from 1929 states that

> *In any zero-sum game,*

$$\max_x \min_j \{\mathrm{col_j(A) \cdot x}\} = \min_{\mathrm{y}} \max_{\mathrm{i}} \{\mathrm{row_i(A) \cdot y}\}$$

We know from the lecture and the textbook that

$$\text{Duality} \Rightarrow \text{Maxmin}.$$

It is also known that the other implication holds (Ilan Adler, 2013 *International Journal of Game Theory*):

$$\text{Duality} \Leftrightarrow \text{Maxmin}.$$

Here we shall prove the Maxmin Theorem; thus the Duality Theorem will also follow. The proof will be through the introduction of an algorithm, called *Multiplicative Updates* (MU), which is an important thing to know in this course on its own right.

2 The Experts Problem and Multiplicative Updates.

Imagine that every day you take the advice of one of n experts, and in the end of the day you find out how much money the advice of each expert will **cost** you: The ith expert on day t costs you c_i^t, a number in $[0, 1]$. If you chose expert j the morning of that day, you incur a cost of c_j^t. This goes on for T days (think of T as a huge number).

Our **goal** is to have *zero regret.* Once we have made choices of an expert $i(t)$ for each day t, we define the regret to be

$$R = \frac{1}{T}(\sum_{t=1}^{T} c_{i(t)}^t - \min_{1 \leqslant i \leqslant n} \sum_{t=1}^{T} c_i^t).$$

That is, our regret is the amount by which the *best single expert* was better than our actual performance. We want to find an algorithm for for choosing the $i(t)$s such that $R \approx 0$ *no matter what c_i^t's bad luck may throw our way.*

There is an algorithm that achieves this lofty goal, namely the MU algorithm explained below. It is a randomized algorithm, in that it makes its choice of $i(t)$ by "flipping coins." Notice that randomness is necessary: If our algorithm deterministically chose an $i(t)$ every morning, then by setting $c_{i(t)} = 1$ and $c_j^t = 0$ for all $j \neq i(t)$, the regret becomes huge (almost 1).

Another observation is this: The more ambitious goal of competing against the best expert *in each day* (instead of against the single expert who does the best overall) is also impossible: No matter what probabilities our algorithm assigns to the experts each day, the costs could be chosen every day to be 0 for the expert with the lowest probability and 1 for all others — again creating a regret of at least $1 - \frac{1}{n}$.

Here is the **Multiplicative Updates (MU)** algorithm:

Each day t, each expert i has a weight, denoted w_i^t.

Initially, $w_i^1 = 1$ for all i
Repeat for $t = 1, 2, \ldots, T$ days
Let $W^t = \sum_{i=1}^n w_i^t$
Choose expert i with probability $\frac{w_i^t}{W^t}$
Update weights: $w_i^{t+1} = w_i^t (1-\epsilon)^{c_i^t}$

Since we are multiplying the weight of each expert by a negative power of the expert's cost, each expert's weight "remembers" the whole performance of this expert:

$$w_i^t = (1-\epsilon)^{\sum_{s=1}^t c_i^t}$$

Experts with better performance have relatively larger weights. This is the basic intuition why MU works, but the precise reason is a bit more subtle, captured in the proof given below:

Theorem 1. *If* $\mathrm{MU} = \sum_{t=1}^T c_{i(t)}^t$ *is the performance of MU, and we denote* $\min_{1 \leqslant i \leqslant n} \sum_{t=1}^T c_i^t$ *by OPT, then the regret* $R = \mathrm{MU} - \mathrm{OPT} \leqslant 2\sqrt{\frac{\ln n}{T}}$.

Proof: Let us look at W^T, the sum of weights at the end. It is easy to see that

$$(1-\epsilon)^{\mathrm{OPT}} < W^T, \tag{1}$$

because the left-hand side is the weight of the best expert, and this smaller than the sum of all weights.

Now let us calculate W^T by looking at the way W^t changes, starting from $W^1 = n$ (this is true since initially all n weights are one).

$$W^{t+1} = \sum_i w^{t+1} =^1 \sum_i w^t (1-\epsilon)^{c_i^t} \leqslant^2 \sum_i w_i^t (1 - \epsilon c_i^t) = W^t (1 - \epsilon \sum \frac{w_i^t}{W^t} c_i^t)$$

Now notice that the last sum is the expected value of the cost incurred by MU at day t, and we shall denote it MU_t. Obviously, $\mathrm{MU} = \sum_t \epsilon \mathrm{MU}_t$.

[1]By definition.

[2]This inequality holds because it so turns out that if $\epsilon < \frac{1}{2}$ and $x \in [0,1]$, $(1-\epsilon)^x \leqslant (1-\epsilon x)$. This follows from the expansion of $\ln(1-\epsilon)$, see below.

We conclude that $W^{t+1} \leqslant W^t(1+\epsilon \mathrm{MU}_t)$, and hence

$$W^T \leqslant n(1+\epsilon \mathrm{MU}_1)\cdot(1+\epsilon \mathrm{MU}_2)\cdots(1+\epsilon \mathrm{MU}_T). \tag{2}$$

Comparing with (1), we conclude that

$$(1-\epsilon)^{\mathrm{OPT}} \leqslant n(1-\epsilon \mathrm{MU}_1)\cdot(1-\epsilon \mathrm{MU}_2)\cdots(1-\epsilon \mathrm{MU}_T). \tag{3}$$

Let us take logarithms of (3):

$$\ln(1-\epsilon)\cdot \mathrm{OPT} \leqslant \ln n + \sum_t \ln(1-\epsilon \mathrm{MU}_t). \tag{4}$$

No recall that, for $x < 1$, $ln(1-x) = -x - \frac{x^2}{2} - \frac{x^3}{3} - \ldots$, and hence $-x - x^2 \leqslant \ln(1-x) \leqslant -x$. Hence, we can replace $\ln(1-\epsilon)$ in the left-hand side of (4) by $-\epsilon-\epsilon^2$, and each $(1-\epsilon\mathrm{MU}_t)$ in the rhs by $-\epsilon\mathrm{MU}_t$, and the inequality will still be valid:

$$(-\epsilon-\epsilon^2)\mathrm{OPT} \leqslant \ln n - \epsilon\sum_t \mathrm{MU}_t = \ln n - \epsilon\mathrm{MU},$$

or, dividing by ϵT, rearranging, and observing that $\mathrm{OPT} \leqslant T$:

$$\frac{1}{T}(\mathrm{MU}-\mathrm{OPT}) \leqslant \frac{\ln n}{\epsilon T} + \epsilon.$$

The right-hand side is the su of two terms, one increasing with ϵ and the other decreasing with it, so we can roughly optimize it by making these terms equal, setting $\epsilon = \sqrt{\frac{\ln n}{T}}$, thus obtaining the theorem.

3 Proof of the Maxmin Theorem

Recall that the maxmin theorem states that, in any two-player zero-sum game described by the $m \times n$ payoff matrix A,

$$\max_x \min_j\{\mathrm{col_j(A)\cdot x}\} = \min_{\mathrm{y}} \max_{\mathrm{i}}\{\mathrm{row_i(A)\cdot y}\}.$$

Now it is easy to see that the left-hand side is at most equal to the right-hand side, since announcing the Row player's probabilities x can only hurt her. Hence, for some $\delta \geqslant 0$,

$$\max_x \min_j\{\mathrm{col_j(A)\cdot x}\} = \min_{\mathrm{y}} \max_{\mathrm{i}}\{\mathrm{row_i(A)\cdot y}\} - \delta. \tag{5}$$

We shall prove that $\delta = 0$.

Suppose that both players play this game repeatedly, for a huge number of days, *by the MU algorithm!* The strategies of each players are her experts. We can assume that the entries of A are all in $[0,1]$ (this is easy to achieve, by adding appropriate number to all payoffs and rescaling). Then, each player at each repetition will incur costs (caused by the probabilities of the other player) that are in $[0,1]$ (assume that the row player incurs cost $1-A_{ij}$). Let w_i^t and u_j^t be the weight of the Row and Column players, respectively, and W^t, U^t the sums over all "experts.". After T days of repeated play by MU, Column has incurred average cost $\mathrm{MU} = \frac{1}{T}\sum_{i,j,t} A_{ij}\frac{w_i^t}{W^t}\frac{u_j^t}{U^t}$. For Row, this will be gain.

Define now the "long term average probabilities" of play for the two players:

$$\hat{x}_i = \frac{1}{T}\sum_t \frac{w_i^t}{W^t}, \hat{y}_i = \frac{1}{T}\sum_t \frac{u_i^t}{U^t}.$$

We shall show that these are in some sense very close to the maxmin/minmax probabilities of play! In particular, they come very close (vanishingly as T increases) to verifying the Maxmin theorem.

Let us follow the inequalities:

$$\begin{aligned}
\max_x \min_j\{\mathrm{col_j(A)\cdot x}\} &\geqslant \\
\min_j\{\mathrm{col_j(A)\cdot \hat{x}}\} &\geqslant \\
\mathrm{MU} - 2\sqrt{\frac{\ln n}{T}} &\geqslant \\
(\max_i\{\mathrm{row}_i(A)\cdot\hat{y}\} - 2\sqrt{\frac{\ln m}{T}}) - 2\sqrt{\frac{\ln n}{T}} &\geqslant \\
\min_y \max_i\{\mathrm{row}_i(A)\cdot y\} - \frac{2(\sqrt{m}+\sqrt{n})}{\sqrt{T}}&.
\end{aligned}$$

The second and third inequalities are implied by the Multiplicative Updates Theorem, for the Column player and the Row player, respectively. The other two inequalities follow from the definition of $\max_x$ and $\min_y$, respectively.

Comparing with (5), we conclude that $\delta \leqslant \frac{2(\sqrt{m}+\sqrt{n})}{\sqrt{T}}$; but since this expression can be made arbitrarily small (by playing many-many rounds T), we see that $\delta = 0$, concluding the proof.