# COS 511: Theoretical Machine Learning

Lecturer: Rob Schapire
Scribe: Sachin Ravi

Lecture 24
May 2, 2013

## 1 Review of Zero-Sum Games

At the end of last lecture, we discussed a model for two player games (call the players Mindy (row) and Max (column)). We said that any such game could be described by its game matrix $M$, which is constructed from Mindy's point of view and thus describes Mindy's loss for a pair of chosen moves. We also said that there were two types of plays: deterministic and randomized. In deterministic play, the players choose according to "pure" strategies where they pick the one move to play. In randomized play, the players choose distributions over their sets of possible moves, according to which their moves will be picked.

In deterministic play, we said the outcome was $M(i,j) \in [0,1]$ if Mindy plays row $i$ and Max plays column $j$. In randomized play, we have an expected outcome. If Mindy has picked distribution $P$ and Max has picked distribution $Q$, the expected outcome is

$$\sum_{ij} P(i)M(i,j)Q(j) = P^T MQ.$$

For randomized games, we often use the notation $M(P,Q) = P^T MQ$ to denote the expected outcome as a function of the distributions chosen by the two players.

## 2 Minimax Theorem

On first look, there seems to be an obvious advantage to playing second, implying that

$$\max_Q \min_P M(P,Q) \leq \min_P \max_Q M(P,Q).$$

However, John von Neumann showed that such an advantage does not exist, that regardless of who goes first, in a game with optimal players, the expected outcome is always the same. Denoting this optimal value as $v$, we have

$$v = \max_Q \min_P M(P,Q) = \min_P \max_Q M(P,Q).$$

We will prove the following theorem using an online learning algorithm.

**Theorem 1** (von Neumann min max theorem)**.** *For randomized zero-sum games of two players:*

$$\min_P \max_Q M(P,Q) = \max_Q \min_P M(P,Q)$$

Assuming $P^* = \arg\min_P \max_Q M(P,Q)$ and $Q^* = \arg\max_Q \min_P M(P,Q)$, the theorem implies the following

$$\forall Q \,:\, M(P^*,Q) \leq v \tag{1}$$
$$\forall P \,:\, M(P,Q^*) \geq v \tag{2}$$

Here, (1) implies that even if Max knows Mindy's strategy the most loss Max can inflict is bounded by $v$ and (2) implies that regardless of what strategy Mindy uses, her loss will be at least $v$. In this sense, $P^*$ is optimal for Mindy.

With knowledge of $M$, we can find $P^*$ through linear programming; However, there are some issues to consider to see why things are not always so simple:

1. We don't know matrix $M$ because arbitrary interactions are being modeled.
2. The matrix $M$ is very large (we cannot apply linear programming).
3. In reality, the opponent may not be optimal and/or adverserial. The strategy $P^*$ only applies for an *optimal* opponent.

Thus, when games are repeatedly played, it is useful to learn the game matrix $M$ and/or the opponent's strategy without the knowledge of either at the beginning of the game. We consider the following online version of $T$ iterations of the game:

---

**for** $t = 1, \ldots, T$ **do**
  Mindy (learner) chooses $P_t$
  Max (environment) chooses $Q_t$ with knowledge of $P_t$
  Learner suffers loss $M(P_t, Q_t)$
  Learner can observer $M(i, Q_t)\ \forall i$
**end for**

---

Two important points to notice here are that Max chooses with knowledge of Mindy's distribution and that Mindy can observe $M(i, Q_t)\ \forall i$, meaning for example if $Q_t$ is concentrated on a pure strategy $j$, then Mindy can observe the entire column $j$ in $M$.

We have that the total loss here is $\sum_{t=1}^{T} M(P_t, Q_t)$. The learner wishes to minimize this total loss when compared to the best possible loss had the learner chosen the best fixed strategy for all of the $T$ iterations. We want to show:

$$\sum_{t=1}^{T} M(P_t, Q_t) \leq \min_P \sum_{t=1}^{T} M(P, Q_t) + \text{small}.$$

### 2.1 Multiplicative Updates

Let us consider a simple multiplicative weight update algorithm, which outputs distributions over rows in the following manner:

$$P_1(i) = \frac{1}{n}\ \forall i$$
$$P_{t+1}(i) = \frac{P_t(i) \cdot \beta^{M(i,Q_t)}}{Z},$$

where $0 < \beta < 1$ and $Z$ is the normalizing constant. This update is similar in style to the weighted majority algorithm. Basically, the bigger the loss for a row $i$, the lesser the probability that we use that row in the future.

Using this update, we get the following result.

**Theorem 2** (Multiplicative Weights Update)**.** *Using the multiplicative weights update, we get*

$$\sum_{t=1}^{T} M(P_t, Q_t) \leq a_\beta \min_P \sum_{t=1}^{T} M(P, Q_t) + c_\beta \ln n,$$

*where $a_\beta$ and $c_\beta$ are functions of $\beta$.*

We will not prove the above result, as it involves analysis similar to the one done for the weighted majority algorithm, mainly using a potential function argument.

**Corollary 3.** *We can choose $\beta$ so that*

$$\frac{1}{T}\sum_{t=1}^{T} M(P_t, Q_t) \leq \min_P \frac{1}{T}\sum_{t=1}^{T} M(P, Q_t) + \Delta_T,$$

*where $\Delta_T = O\left(\sqrt{\frac{\ln n}{T}}\right)$.*

We can see that $\Delta_T \to 0$ as $T \to \infty$ and so the corollary states that the average per-round loss for the learner approaches the best possible average per-round loss.

Using the algorithm and its corresponding result, we will prove the min-max theorem from above. We say Mindy uses the multiplicative weights algorithm to set $P_t$ and that Max sets $Q_t$ so as to maximize Mindy's loss:

$$Q_t = \arg\max_Q M(P_t, Q)$$

We also define

$$\bar{P} = \frac{1}{T}\sum_{t=1}^{T} P_t$$
$$\bar{Q} = \frac{1}{T}\sum_{t=1}^{T} Q_t$$

*Proof.* We can see that both $\bar{P}$ and $\bar{Q}$ are distributions since they are the average of distributions. Since we know that $\max_Q \min_P M(P, Q) \leq \min_P \max_Q M(P, Q)$, to show equality, we need to show that $\max_Q \min_P M(P,Q) \geq \min_P \max_Q M(P,Q)$. We have

$$\begin{align}
\min_P \max_Q P^T MQ &\leq \max_Q \bar{P}^T MQ \tag{3}\\
&= \max_Q \frac{1}{T}\sum_{t=1}^{T} P_t^T MQ \tag{4}\\
&\leq \frac{1}{T}\max_Q \sum_{t=1}^{T} P_t^T MQ \tag{5}\\
&= \frac{1}{T}\sum_{t=1}^{T} P_t^T MQ_t \tag{6}\\
&\leq \min_P \frac{1}{T}\sum_{t=1}^{T} P^T MQ_t + \Delta_T \tag{7}\\
&= \min_P P^T M\bar{Q} + \Delta_T \tag{8}\\
&\leq \max_Q \min_P P^T MQ + \Delta_T, \tag{9}
\end{align}$$

where again $\Delta_T \to 0$ as $T \to \infty$, meaning that the desired result follows. Here, (3) follows by definition of the minimum, (4) by definition of $\bar{P}$, (5) by convexity, (6) by defintion of $Q_t$, (7) by Corollary 2.1, (8) by definition of $\bar{Q}$, and (9) by definition of the maximum.

□

If we skip the first inequality, we get that

$$\max_Q \bar{P}^T MQ \leq v + \Delta_T,$$

where $v = \max_Q \min_P P^T MQ$. Thus, taking the average of $P_t$'s computed over the rounds of the algorithm, we get a distribution $\bar{P}$ that is within $\Delta_T$ of the optimal. By running the algorithm for more rounds, we can get closer and closer to the optimal and thus we call $\bar{P}$ an approximate min max strategy. By a similar argument, we can show that $\bar{Q}$ is an approximate max min strategy.

## 3 Relation to Online Learning

We will now try to relate our previous analysis to the setting of online learning. The problem is as follows:

---

**for** $t = 1, \ldots, T$ **do**
  Observe $x_t \in \mathcal{X}$
  Predict $\hat{y}_t \in \{0, 1\}$
  Observe true label $c(x_t)$ (mistake if $c(x_t) \neq \hat{y}_t$)
**end for**

---

Let $\mathcal{H}$ be the set of all possible hypotheses. We associate each $h \in \mathcal{H}$ as being an expert and we wish to perform as well as the best expert. As in the past, we would like to show

$$\#\text{mistakes} \leq \#\text{mistakes of best } h + \text{small}$$

Assuming the sets $\mathcal{H}$ and $\mathcal{X}$ are finite, for this problem, we set up the game matrix $M$ as follows:

$$M = \begin{array}{c} \\ h_1 \\ h_2 \\ \vdots \\ h_m \end{array} \begin{array}{c} \begin{array}{cccc} x_1 & x_2 & \dots & x_n \end{array} \\ \begin{pmatrix} M(1,1) & M(1,2) & \dots & M(1,n) \\ M(2,1) & M(2,2) & \dots & M(2,n) \\ \vdots & \vdots & \ddots & \vdots \\ M(m,1) & M(m,2) & \dots & M(m,n) \end{pmatrix} \end{array},$$

where

$$M(h,x) = \begin{cases} 1, & h(x) \neq c(x) \\ 0, & \text{otherwise} \end{cases}$$

Given a particular $x_t \in \mathcal{X}$, the algorithm uses the distribution $P_t$ to make a prediction on $x_t$. Choose $h$ according to $P_t$. Then, let $\hat{y}_t = h(x_t)$.

We then let $Q_t$ be the distribution concentrated on $x_t$. This setup implies the following result

$$\underbrace{\sum_{t=1}^{T} M(P_t, x_t)}_{\text{E[\# mistakes]}} \leq \underbrace{\min_{h \in \mathcal{H}} M(h, x_t)}_{\text{\# mistakes of best } h} + \text{small}, \tag{10}$$

since $M(P_t, x_t) = \sum_{h \in \mathcal{H}} P_t(h) \cdot 1\{h(x) \neq c(x)\} = \Pr_{h \sim P_t}[h(x) \neq c(x)]$. If we properly plug into the above result, we will find that we acheive the same bound as in the analysis done for the weighted majority algorithm in the online learning model.

# 4 Relation to Boosting

We now turn to boosting and see how to set it up as a game between two players: the boosting algorithm and the weak learner. Here, $\mathcal{H}$ is the set of weak hypotheses and $\mathcal{X}$ is the training set.

**for** $t = 1, \dots, T$ **do**
  Boosting Algorithm chooses distribution $D_t$ on $\mathcal{X}$
  Weak Learner chooses $h_t \in \mathcal{H}$
**end for**

where it is assumed that

$$\Pr_{x \sim D_t}[h_t(x) \neq c(x)] \leq \frac{1}{2} - \gamma.$$

We cannot use the game matrix $M$ used in the last section because it would give us a distribution on rows ($h$'s), whereas here we want a distribution on columns ($x$'s). Thus, we flip the game matrix $M$, and renormalize so that the matrix values are in $[0,1]$. We get the game matrix

$$M' = 1 - M^T$$

where

$$M' = \begin{array}{c c} & \begin{array}{cccc} h_1 & h_2 & \dots & h_n \end{array} \\ \begin{array}{c} x_1 \\ x_2 \\ \vdots \\ x_m \end{array} & \begin{pmatrix} M'(1,1) & M'(1,2) & \dots & M'(1,n) \\ M'(2,1) & M'(2,2) & \dots & M'(2,n) \\ \vdots & \vdots & \ddots & \vdots \\ M'(m,1) & M'(m,2) & \dots & M'(m,n) \end{pmatrix} \end{array},$$

meaning

$$M'(x,h) = \begin{cases} 1, & h(x) = c(x) \\ 0, & \text{otherwise} \end{cases}$$

We then have that $D_t = P_t$ and that $Q_t$ is the distribution fully concentrated on the $h_t$ given to us. To be clear, the booster is simulating the multiplicative weights algorithm on the game matrix. Applying the multiplicative weights algorithm, we have

$$\frac{1}{T}\sum_{t=1}^{T} M'(P_t, h_t) \leq \min_x \frac{1}{T}\sum_{t=1}^{T} M'(x, h_t) + \Delta_T.$$

Notice that

$$\begin{aligned} M'(P_t, h_t) &= \sum_{x \in \mathcal{X}} P_t(x) \cdot 1\{h_t(x) = c(x)\} \\ &= \Pr_{x \sim P_t}[h_t(x) = c(x)] \\ &\geq \frac{1}{2} + \gamma, \end{aligned}$$

implying that

$$\frac{1}{2} + \gamma \leq \frac{1}{T}\sum_{t=1}^{T} M'(P_t, h_t) \leq \min_x \frac{1}{T}\sum_{t=1}^{T} M'(x, h_t) + \Delta_T.$$

Since the second inequality applies to the minimum $x$, it also applies for all $x$:

$$\forall x : \frac{1}{T}\sum_{t=1}^{T} M'(x, h_t) \geq \frac{1}{2} + \gamma - \Delta_T > \frac{1}{2},$$

where the second inequality is true since $\Delta_T \to 0$ as $T \to \infty$. Note that $\frac{1}{T}\sum_{t=1}^{T} M'(x, h_t)$ is the fraction of weak hypotheses that correctly classify $x$.

Since we have shown that for any $x$, the fraction of weak hypothesis that correctly classify $x$ is greater than $\frac{1}{2}$, we have that the majority vote

$$MAJ(h_1(x), \dots, h_T(x)) = c(x)$$

for all $x$.

Thus, this game formulation solves the problem of Boosting in the simple case.