Report: Ski Rental Problem with Machine Learning Predictors

Jerry Huang

April 2020

1 Problem Statement and Introduction

In the ski rental problem, a skier faces the decision between buying skis that cost b and renting them for a cost of 1 per day. Clearly, if the skiers know in advance that he will ski for more than b days, it is in his best interest to buy skis at the start, and vice versa. Our research investigates novel algorithms that can leverage one or more experts in the form of machine learning predictors to optimize the costs of the skier when not all of the information is available at the start of the problem. The baseline of our algorithm's performance is the well-known deterministic algorithm's performance, which achieves a competitive ratio of 2.

1.1 Online Algorithms

Online algorithms are algorithms that are fed data piece-by-piece as time goes on, contrasting with the traditional offline algorithms who start with the entire input. The performance of online algorithms are measured through competitive analysis, where the costs of the algorithm are compared to those of the optimal offline algorithm. Namely,

$$CR = \frac{\text{Worst case cost incurred by } \textit{online} \text{ algo}}{\text{Worst case cost incurred by } \textit{offline} \text{ algo}}$$

1.2 Predictors

Initially, we modeled the predictors as binary classifiers that would report a prediction b that would either be greater than or less than 1 with a probability p. However, this formulation seemed to assume too many unknowns as we ran into questions regarding the independence of these predictions. Questions of whether $P(A|B_1)$ and $P(A|B_2)$ were truly independent, and how we would combine the probabilities of multiple predictors motivated us to reformulate our model of these predictors.

In the second iteration of our ML predictors, we modeled our experts as PAC-learnable models, where PAC stands for "probably approximately correct." Formally, they are defined as a class $\mathcal{C}$ of functions $f : \mathbb{R}^d \to \{0, 1\}$, where $\mathcal{C}$ is *PAC-learnable* if there exists an algorithm A such that for every $f \in \mathcal{C}$, probability distribution $\mathcal{D}$, $0 \leq \epsilon < \frac{1}{2}$, and $0 \leq \delta < 1$, A outputs a predictor h over a set of random examples drawn from $\mathcal{D}$ with probability at least $1-\delta$ such that the error, $error(h, f) < \epsilon$. Here $error(h, f) = Pr_{x\in\mathcal{D}}(h(x) \neq f(x))$. Thus $1 - \epsilon$ is the probability of success.

2 Previous Approaches We Tried

Here are previous approaches that we discussed over the semester – some have been abandoned while others were left as possible options we may come back to.

2.1 One Expert Case with Simple Predictors

We considered the case of having 1 expert that outputs a prediction b with the probability of being correct equal to p. We get the following 2 cases:

if $b < 1$, rent till b

$$CR_1 = p(1) + (1-p) \cdot 2 = 2 - p = 1 + (\sqrt{1-p})^2$$

if $b \geq 1$, rent till some time t

$$CR_2 = p(1+t) + (1-p)\frac{1+t}{min(t,1)} = 1 + p \cdot t + \frac{1-p}{t}$$

such that $t \leq 1$ since there is no case where the skier wants to be renting for duration longer than 1.

To minimize CR_2, we let $t = \sqrt{\frac{1-p}{p}}$. Substituting, we get that

$$CR_2 = 1 + 2\sqrt{p(1-p)} = 1 + t$$

In summary, if $p < \frac{1}{2}$, then $t = 1$ or if $p \geq \frac{1}{2}$, then $t = \sqrt{\frac{1-p}{p}}$. The rationale behind this is that when the probability is less than $\frac{1}{2}$, the loss gained from using the prediction outweighs the potential gain.

For multiple simple predictors, we had several discussions on how to combine the probabilities given by multiple discussions. Ideas like "if we have two conflicting predictions with the same probability, we should throw both out," and "we should only accept predictions with a probability greater than 1/2," were discussed. Furthermore, we realized that these predictors could not be categorized as being truly independent, so a p^2 and $(1-p)^2$ weighting system for two predictors that both predicted the same season length would be inaccurate.

Status: This approach was later reformulated to reflect our adoption of the more rigorous PAC-*learnable* model for the predictors.

2.2 One Expert Case with PAC Learning Model

Here we revisit the one predictor case. Consider a general binary classification problem with feature space $\mathbb{K}$, with $(x, y) \in \mathbb{K}$ for $x \in \mathbb{R}^d$ and $y \in \{0, 1\}$. Suppose there is a predictor over the training set $f_T : \mathbb{R}^d \to \{0, 1\}$ such that $Pr_{(x,y)\sim\mathbb{K}}(f_T(x) = y) \geq p \;\forall\; x$, and $Pr_{(x,y)\sim\mathbb{K}}(y = 1) = q$. We will define the event $y = 1$ to be a ski season lasting beyond b, which we set equal to 1 here, and the event $y = 0$ to be a ski season lasting shorter than b.

Bounds on the two cases

$$Pr(y = 1, f_T(x) = 1) = Pr(y = 1) \cdot Pr(f_T(x) = 1|y = 1) \leq q \cdot p$$

$$Pr(y = 0, f_T(x) = 1) \geq (1-p) \cdot (1-q)$$

From which we can obtain:

$$Pr(y = 1|f_T(x) = 1) \geq \frac{p \cdot q}{pq + (1-p)(1-q)} = p_0 \tag{1}$$

$$Pr(y = 0|f_T(x) = 0) \geq \frac{p \cdot (1-q)}{p(1-q) + (1-p)q} = p_1 \tag{2}$$

Let p_0 and p_1 equal the RHS of equations (1) and (2) respectively.

Now, to show that we have successfully bounded the competitive ratio, we consider the following cases:

Case 1: When $f_T(x) = 0$, $CR \le p_1 + 2 \cdot (1 - p_1) = 2 - p_1$. Intuitively this is a proper upper bound since increasing p_1 decreases the competitive ratio.

Case 2: When $f_T(x) = 1$, the strategy is to buy at some α. By the same analysis of the one expert case as before, $\alpha = \sqrt{\frac{1-p_0}{p_0}}$ minimizes the competitive ratio for $p_0 > \frac{1}{2}$. When $p_0 \ge \frac{1}{2}, CR \le 1 + 2\sqrt{p_0(1 - p_0)}$.

Case 3: Consider when $q = \frac{1}{2}$, $p_1 = p_0 = p$, and the overall competitive ratio becomes

$max(1 + 2\sqrt{p(1 - p)}, 2 - p) = 1 + 2\sqrt{p(1 - p)}$.

Status: When we started deriving scenarios where we had two or three predictors, the conditional probabilities grew increasingly complex and we got stuck trying to obtain tighter bounds, and so we decided to table this approach.

3 Concrete (Active) Findings

This final section consists of approaches we are currently actively working on.

3.1 One Expert Case

Notation:

$$\begin{aligned} y &= \text{true value of length of ski season - adversarial to algorithm} \\ \hat{y} &= \text{value of the predictor - either } < B \text{ or } \ge B \\ p &= \text{probability the predictor will be correct (long run fraction of times correct)} \\ B &= \text{cost of skis - will be set as 1 for simplicity} \end{aligned}$$

3.2 Lemmas

Lemma 3.2.1

$$\forall\, p \in [0, 1] \qquad \beta(p), \alpha(p) \le 1 \tag{3}$$

Proof. Consider a situation where $\beta, \alpha > 1$ and suppose that the true length of the season, y, is within $[0, 1]$. Regardless of when the adversary chooses to end the season, both strategies are clearly optimal as the buying points are after 1. Suppose that $y \ge 1$. Since the optimal cost when $y \ge 1$ is 1, the adversary only seeks to force the algorithm to incur the highest cost. Thus, $y \in \big[\max(\alpha(p), \beta(p)), \infty\big)$.

Furthermore, any y in this range is valid since after $\max(\alpha(p), \beta(p))$, the algorithm has bought skis and can no longer incur cost. Therefore, since $CR = p \cdot \left(\frac{\beta(p)+1}{1}\right) + (1 - p) \cdot \left(\frac{\alpha(p)+1}{1}\right)$, in order to minimize the competitive ratio $\beta(p)$ and $\alpha(p)$ should never be greater than 1.

Lemma 3.2.2

$$\forall\, \alpha, \beta \le 1 \text{ and } \frac{1}{2} \le p \le 1, \qquad \beta \le \alpha \tag{4}$$

Proof by contradiction. Assume instead that $\alpha < \beta$,

Case 1: $y < \alpha \le 1$

Competitive ratio is always 1 (i.e. we match the performance of the offline algorithm).

Case 2 $\alpha < y < \beta \leq 1$

$$\begin{aligned} CR_2 &= p \cdot \left(\frac{\alpha+1}{\alpha}\right) + (1-p) \cdot 1 \\ &= p \cdot \left(1 + \frac{1}{\alpha}\right) + 1 - p \\ &= \frac{p}{\alpha} + 1 \end{aligned}$$

In this case, as p increases, the competitive ratio increases as well.

Case 3: $\beta < y \leq 1$

$$\begin{aligned} CR_3 &= p\left(\frac{\alpha+1}{\beta}\right) + (1-p)\left(\frac{\beta+1}{\beta}\right) \\ &= \frac{p}{\beta}(\alpha+1) + 1 + \frac{1}{\beta} - p - \frac{p}{\beta} \\ &= \frac{1}{\beta}(p(\alpha+1) + 1 - p) + 1 - p \\ &= \left(\frac{p\alpha+1}{\beta}\right) + 1 - p \end{aligned}$$

Case 4: $y > 1$

$$CR_4 = p(\beta+1) + (1-p)(\alpha+1) \leq 2$$

Fix any β, then find optimal choice for α. Since $\alpha, \beta \geq 0$. CR_3 is monotone decreasing with respect to p and CR_2 is monotone increasing with respect to p.

Since that the worst-case competitive ratio we obtained (when p was equal to 1/2) was slightly less than 1.81, if this $(0, \alpha, \beta, 1)$ strategy is going to have a lower ratio, looking at CR_2, we must have $p/\alpha \leq 0.81$. This means that $\alpha \geq p/0.81$. Plugging this into CR_3, we get $CR_3 \geq 1.23p^2 + 2 - p$, which is always at least around 1.8 when $p \geq 0.5$.

Note: these values were obtained through numerical simulations.

Additional cases that cover when $\alpha \vee \beta > 1$.

Case 5: $1 < \alpha < \beta$

The relevant competitive ratios are $\alpha \leq y < \beta$ and $y \geq \beta$. The former gives competitive ratio $p(1+\beta) + (1-p)(1+\alpha)$ and the latter gives $p(1+\beta) + (1-p)(1+\alpha)$. This is minimized by setting $\alpha = \beta = 1$, resulting in a competitive ratio of 2.

Case 6: $0 < \alpha < 1 < \beta$

The three intervals are $[\alpha, 1), [1, \beta), [\beta, \infty)$. The respective competitive ratios are

$$\begin{aligned} CR_1 &= p\left(1 + \frac{1}{\alpha}\right) + (1-p) \\ CR_2 &= p\beta + (1-p)(1+\alpha) \\ CR_3 &= p(1+\beta) + (1-p)(1+\alpha). \end{aligned}$$

Notice that $CR_3 \geq CR_2$, so we can ignore CR_2. The algorithm sets $\beta = 1$. Solving $CR_1 = CR_3$ for α (and taking the positive solution) yields

$$\alpha = \frac{p - \sqrt{-p(3p-4)}}{2(p-1)}.$$

Plugging this into CR_1 (or CR_3) yields an increasing function in p passing through the points $(0.5, \approx 1.81)$ and $(1, 2)$. In fact, it looks like the same graph of the best competitive ratio we got, flipped about the $p = 0.5$ line. For reference, that ratio was

$$\frac{1}{2}\left(3 - p + \sqrt{(2-3p)p+1}\right).$$

Lemma 3.2.3

$$\forall\, p \in [0,1] \qquad \alpha(p) = 1 \tag{5}$$

This lemma is proved in the Section 3.3 when we consider Regions I, II, and III, which together bound α to a value of 1.

3.3 Deterministic Single Expert

In a deterministic setting the algorithm only has two strategies depending on the value of the prediction. These strategies are restricted to when the algorithm should buy skis, specifically the algorithm will buy at

$$\begin{array}{ll} \alpha(p) & \text{if } \hat{y} < 1 \\ \beta(p) & \text{if } \hat{y} \geq 1 \end{array}$$

where α corresponds to the rent strategy and β corresponds to the buy strategy.

Thus, the adversary has three regions that it can choose from.

1. Region 1: Between 0 and $\beta(p)$
2. Region 2: Between $\beta(p)$ and $\alpha(p)$
3. Region 3: Between $\alpha(p)$ and 1

Visually, the regions look like

$$0 \text{ —— } \beta(p) \text{ —— } \alpha(p) \text{ —— } 1$$

And so, the goal of the adversary is to maximize the competitive ratio across these regions and our algorithm's goal is to minimize this maximum.

Now, we calculate the competitive ratios in each of region. Here we assume that $\alpha = 1$ and that $\beta = \beta(p)$.

Region 1 ($0 < y < \beta$): It is clear that the adversary does not gain by setting y to any value in region 1. Logically, if we plan to rent until some value $\beta(p)$ and we end up stop skiing before then, we have achieved a competitive ratio of 1.

Region 2 ($\beta \leq y < \alpha = 1$):

$$CR_2 = \frac{py + (1-p)(1+\beta)}{y}$$

Let $y = \beta$ to maximize this expression

$$= \frac{p\beta + (1-p)(1+\beta)}{\beta} = 1 + \frac{1-p}{\beta}$$

Region 3 ($y \geq 1$):

$$CR_3 = \frac{p(1+\beta) + (1-p)(1+\beta)}{1} = 2 + \beta p - p$$

To find the maximum of CR_2 and CR_3, we can set them equal to one another given one function is monotonically increasing and the other is monotonically decreasing. We can verify this to be true with a simple check. For CR_2, as β increases from 0 to 1 we see that CR_2 decreases, and for CR_3, as β increases, CR_3 increases.

Solve $CR_2 = CR_3$:

$$\begin{aligned} CR_2 &= CR_3 \\ 1 + \frac{1-p}{\beta} &= 2 + \beta p - p \\ \frac{1-p}{\beta} &= 1 - p + p\beta \\ \beta &= \frac{\sqrt{-3p^2 + 2p + 1} + p - 1}{2p} \end{aligned}$$

Note: when $p = 1, \beta = 0$. When $p = 1/2$, $\beta \approx 0.618$.

Plugging into CR_2:

$$CR_2 = \frac{1}{2}\left(3 - p + \sqrt{(2-3p)p + 1}\right).$$

When $p = 1/2$, $CR_2 \approx 1.81$.

The *additional* following cases cover the ranges where α is not equal to 1.

Region I: $0 < \beta < \alpha < 1$

The three relevant competitive ratios are the following:

$$\begin{aligned} CR_1 &= p + (1-p)\left(1 + \frac{1}{\beta}\right) \\ CR_2 &= p\left(1 + \frac{1}{\alpha}\right) + (1-p)\frac{1+\beta}{\alpha} \\ CR_3 &= p(1+\beta) + (1-p)(1+\alpha). \end{aligned}$$

Notice $CR_2 \geq CR_3$, so we can ignore CR_3. (For any values of α, β, we can view them as linear functions in p on $[0.5, 1]$. Throughout this interval, $CR_2 \geq CR_3$.) To minimize the maximum, the algorithm sets $\alpha = 1$.

Region II: $0 < \beta < 1 < \alpha$

The three relevant competitive ratios are the following:

$$\begin{aligned} CR_1 &= p + (1-p)\left(1 + \frac{1}{\beta}\right) \\ CR_2 &= p(1+\beta) + (1-p)\alpha \\ CR_3 &= p(1+\beta) + (1-p)(1+\alpha). \end{aligned}$$

To minimize the maximum, the algorithm sets $\alpha = 1$.

Region III: $1 < \beta < \alpha$

The three relevant competitive ratios are the following:

$$\begin{aligned} CR_1 &= \beta \\ CR_2 &= p(1+\beta) + (1-p)\alpha \\ CR_3 &= p(1+\beta) + (1-p)(1+\alpha). \end{aligned}$$

To minimize the maximum, the algorithm sets $\alpha = \beta = 1$.

3.4 Randomized Single Expert

For the randomized case, the algorithm follows the same α and β conventions defined in the deterministic algorithm: if the prediction says rent, then we pick a threshold determined by $\alpha_p(x)$; otherwise, we follow $\beta_p(x)$. For simplicity, we assume that they are both supported on $[0, 1]$.

Intuitively, if $p = 1/2$, then these distributions should both be $\frac{1}{e-1}e^x$ (i.e., the no-prediction algorithm), and if $p = 1$, then $\beta_p(x)$ should place a lot of mass at 0, and $\alpha_p(x)$ should place a lot of mass at 1.

The competitive ratios are:

$$\begin{aligned}
CR_1 &= \max_{y\in[0,1]} \left[p\left(\frac{1}{y}\int_0^y (x+1)\alpha_p(x)dx + \int_y^1 \alpha_p(x)dx\right) + (1-p)\left(\frac{1}{y}\int_0^y (x+1)\beta_p(x)dx + \int_y^1 \beta_p(x)dx\right)\right] \\
CR_2 &= p\left(\int_0^1 (x+1)\beta_p(x)dx\right) + (1-p)\left(\int_0^1 (x+1)\alpha_p(x)dx\right) = p(1+\mu(\beta_p)) + (1-p)(1+\mu(\alpha_p)) \\
&= 1 + p\cdot\mu(\beta_p) + (1-p)\mu(\alpha_p),
\end{aligned}$$

where $\mu(\alpha_p)$ and $\mu(\beta_p)$ denote the expectation of $\alpha_p(x)$ and $\beta_p(x)$, respectively.

Let's consider the case where $p = 1 - \epsilon$ for some small value of ϵ. If we set $\alpha_p(x)$ to be a point mass at 1, then

$$\begin{aligned}
CR_1 &= 1 - \epsilon + \epsilon \max_y \left(\frac{1}{y}\int_0^y (x+1)\beta_p(x)dx + \int_y^1 \beta_p(x)dx\right) \\
&\le 1 + \epsilon \max_y \int_0^y \frac{x+1}{y}\beta_p(x)dx \\
&\le 1 + 2\epsilon \max_y \int_0^y \frac{\beta_p(x)}{y}dx \\
CR_2 &= (1-\epsilon)\left(\int_0^1 (x+1)\beta_p(x)dx\right) + 2\epsilon \\
&= 1 + \epsilon + (1-\epsilon)\mu(\beta_p(x)) \\
&\le 1 + \epsilon + \mu(\beta_p(x)).
\end{aligned}$$

Intuitively, to minimize CR_2, the mean of β should be low. However, if it is too low, then the adversary can choose a small y such that $[0, y]$ contains a lot of mass, so CR_1 is large. To balance these two quantities, consider setting $\beta_p(x)$ to be a point mass at $\sqrt{\epsilon}$. In this case, we have

$$\begin{aligned}
CR_1 &\le 1 + 2\epsilon \cdot \frac{1}{\sqrt{\epsilon}} = 1 + 2\sqrt{\epsilon} \\
CR_2 &\le 1 + \epsilon + \sqrt{\epsilon} \le 1 + 2\sqrt{\epsilon},
\end{aligned}$$

so our competitive ratio is at most $1 + 2\sqrt{\epsilon}$.

To see that this is optimal, suppose we aim for a competitive ratio less than $1+\epsilon+\sqrt{\epsilon}/8$. Then $\mu = \mu(\beta_p(x))$ must be less than $\frac{\sqrt{\epsilon}}{8(1-\epsilon)} < \sqrt{\epsilon}/4$. By Markov's inequality, the interval $[0, 2\mu]$ contains at least half of the total mass. The adversary can pick $y = 2\mu < \sqrt{\epsilon}/2$ to achieve

$$\begin{aligned}
CR_1 &\ge 1 - \epsilon + \epsilon \max_y \int_0^y \frac{\beta_p(x)}{y}dx \\
&\ge 1 - \epsilon + \frac{\sqrt{\epsilon}}{4},
\end{aligned}$$

which for small enough ϵ, is greater than $1 + \epsilon + \sqrt{\epsilon}/8$.

4 Acknowledgments and References

Acknowledgements

I thank Dr. Panigrahi for his guidance through my first semester of research in the field of computer algorithms, PhD candidate Kevin Sun for leading discussions and spearheading the research project, and fellow students Dorian Barber and Rahul Ramesh for working together with me on this exciting project. While this marks the end of my second year at Duke, I'm excited to continue exploring this research area.

References

Original Ski Learning Problem Statement: "Online Algorithms for Rent-or-Buy with Expert Advice"

PAC Learning Models: `https://www.cs.utexas.edu/~klivans/f06lec2.pdf`