Results on Various Models of Mistake-Bounded Online Learning

Raymond Feng* Andrew Lee† Espen Slettnes‡

December 29, 2021

Abstract

We determine bounds for several variations of the mistake-bound model. The first half of our paper presents various bounds on the weak reinforcement model and the delayed, ambiguous reinforcement model. In both models, the adversary gives r inputs in one round and only indicates a correct answer if all r guesses are correct. The only difference between the two models is that in the delayed, ambiguous model, the learner must answer each input before receiving the next input of the round, while the learner receives all r inputs at once in the modified weak reinforcement model. We also prove generalizations for multi-class functions.

Then, we prove a lower and upper bound of the maximum factor gap that are tight up to a factor of r between the modified weak reinforcement model and the standard model.

Lastly, we also introduce several related models for learning with permutation patterns: the order model, the relative position model, and the delayed relative position model. In these models, a learner attempts to learn a permutation from a set of permutations F by guessing statistics related to sub-permutations. We similarly define the notions of weak versus strong reinforcement and of delayed, ambiguous, reinforcement, and determine some sharp bounds by mimicking sorting algorithms.

*rfeng2004@gmail.com
†leeandrew1029@gmail.com
‡espen@slett.net

1 Introduction

We look at several variations of the mistake-bound model [12]. The standard model [1, 13] (also called the standard strong reinforcement learning model [1]) is the situation of a learner attempting to classify inputs (in the set X) with labels (in the set Y) based on a number of possible functions $f : X \to Y$ in F. The learning proceeds in rounds, and each round the adversary gives the learner an input, and the learner must then guess the corresponding label. After each round, the adversary informs the learner of the correct answer (and therefore whether the learner was right or wrong). A variation of this model is the standard weak reinforcement model [1, 2], where the adversary only tells the learner "YES" if they were correct and "NO" otherwise. This variant is also commonly called the *bandit model* [5, 6, 7, 11, 13].

The study of the efficiencies of these learning algorithms, as measured by either the maximum average number of mistakes or maximum number of mistakes (depending on whether non-deterministic learning algorithms are allowed) that they make while learning a function, is relevant to the field of machine learning for services like YouTube's video recommendation algorithm, the tailored ad services provided by Google, or the friendship recommendation process in social media sites like Facebook.

For any learning scenario, we generally let $\text{opt}_{\text{scenario}}(F)$ represent the optimal worst case number of mistakes that a learning algorithm could achieve [1]. For example, for weak reinforcement learning/bandit model, standard model/strong reinforcement learning, or delayed, ambiguous reinforcement learning, the optimal worst case performances of learning algorithms would be denoted $\text{opt}_{\text{weak}}(F) = \text{opt}_{\text{bandit}}(F)$, $\text{opt}_{\text{std}}(F) = \text{opt}_{\text{strong}}(F)$, and $\text{opt}_{\text{amb},r}(F)$, respectively. There are some obvious inequalities that follow by definition, such as $\text{opt}_{\text{strong}}(F) \leq \text{opt}_{\text{weak}}(F)$, just from the fact that the learner has strictly more information in one scenario compared to the other.

In [1], Auer and Long define the delayed, ambiguous reinforcement model and compare it to a modified version of the standard weak reinforcement model (henceforth called the modified weak reinforcement model). The delayed, ambiguous reinforcement model is a situation where the learner receives a fixed number (r) of inputs each round, and each input is given to the learner after they have answered the previous one. On the other hand, the learner receives all r inputs at once for each round in the modified weak reinforcement model. In both situations, at the end of every round of r inputs, the adversary says "YES" if the learner answered all r inputs correctly and "NO" otherwise. To compare the two situations, they define $\text{CART}_r(F)$ (where F is a set of functions $f : X \to Y$) to be a set of functions $f' : X^r \to Y^r$ where each $f \in F$ has a corresponding $f' \in \text{CART}_r(F)$ such that for any $x_1, x_2, \ldots, x_r \in X$, we have $f'((x_1, x_2, \ldots, x_r)) = (f(x_1), f(x_2), \ldots, f(x_r))$. They use $\text{opt}_{\text{weak}}(\text{CART}_r(F))$ to analyze the efficiencies of learning algorithms in the modified weak reinforcement setting. Auer and Long proved that the two situations are not equivalent for the learner in Theorem 16 of [1], which is the assertion that there is some input set X and set F of functions from X to $\{0, 1\}$ such that $\text{opt}_{\text{amb},2}(F) < \text{opt}_{\text{weak}}(\text{CART}_2(F))$.

In Sections 2 and 3, we generalize Theorem 16 from [1] to general r in place of 2; moreover, we give sharp bounds (up to a constant factor) on the exact values of $\text{opt}_{\text{weak}}(\text{CART}_r(F))$ (in Section 2) and $\text{opt}_{\text{amb},r}(F)$ (in Section 3) for all sets F which are a subset of the non-decreasing functions from X to $\{0, 1\}$. The results show that it is possible for there to be an exponential difference in r between the two learning scenarios as the number of functions $|F|$ increases.

In Sections 4 and 5, we extend our bounds from Sections 2 and 3 on $\text{opt}_{\text{amb},r}(F)$ and $\text{opt}_{\text{weak}}(\text{CART}_r(F))$ to multi-class functions, where the labels are not limited to just $Y = \{0, 1\}$.

In a paper from Long [13], he determines the maximum factor gap between the standard model and the standard weak reinforcement model (also called the bandit model) for multi-class functions. He proved the upper bound $\text{opt}_{\text{bandit}}(F) \leq (1 + o(1))(|Y| \ln |Y|) \text{opt}_{\text{std}}(F)$ and constructed infinitely many F for which $\text{opt}_{\text{bandit}}(F) \geq (1 - o(1))(|Y| \ln |Y|) \text{opt}_{\text{std}}(F)$ as a lower bound. A mistake in the proof of the lower bound was corrected by Geneson [9].

In Section 6, we generalize this result to determine a lower bound and upper bound on the maximum factor gap between $\text{opt}_{\text{weak}}(\text{CART}_r(F))$ and $\text{opt}_{\text{std}}(F)$ for multi-class functions using probabilistic methods and linear algebra techniques. The proof uses techniques previously used for experimental design [16, 15] and hashing, derandomization, and cryptography [4, 14]. The bounds are tight up to a factor of r.

In Section 7, we explore a general result about any learning scenario and a method to directly relate the number of mistakes that a learner makes in different learning scenarios. We prove that if the learner is guaranteed to make at most M mistakes in the learning process in some learning model, then the adversary can always force the learner to make M mistakes in the first M inputs. We also prove a preliminary bound relating the standard model with deterministic learning algorithms to the bandit model that allows non-deterministic learning algorithms and conjecture a stronger bound that is related to the upper bound on $\text{opt}_{\text{bandit}}(F)$ from [13].

In Section 8, we define new models where the learner is trying to guess a function from a set of permutations of length n. In the order model, denoted $\text{opt}(\text{PERM}_r(F))$, each turn, the adversary chooses r inputs, and the learner attempts to guess the corresponding sub-permutation. In the relative position model, denoted $\text{opt}(\text{RPOS}_r(F))$, each turn, the adversary chooses r inputs and a distinguished element x among them, and the learner attempts to guess the relative position of x in the corresponding sub-permutation. Finally, in the delayed relative position model, the adversary instead gives the r elements to be compared to x one at a time. We first establish general upper bounds similarly to previous bounds. We then discuss adversary strategies for a few families of permutations that resemble sorting algorithms.

Finally, in Section 9, we collate some areas of future work based on the results in our paper.

2 Bounds on $\text{opt}_{\text{weak}}(\text{CART}_r(F))$

In this section, we establish upper and lower bounds on $\text{opt}_{\text{weak}}(\text{CART}_r(F))$ that are within a constant factor of each other for non-decreasing F, which we define in Section 2.1. We show that for non-decreasing F, $\text{opt}_{\text{weak}}(\text{CART}_r(F)) = (1 \pm o(1))r \ln(|F|)$ (Theorem 5).

2.1 Non-decreasing F

In Sections 2 and 3 we mainly consider non-decreasing sets of functions F.

Definition 1. Without loss of generality impose an ordering on the set X, and call its

elements $\{1, 2, \ldots, |X|\}$. Let $F = \{f_1, f_2 \ldots, f_{|F|}\}$ be a subset of the functions from X to $\{0, 1\}$. We say that F is *non-decreasing* if every function in F is non-decreasing.

In other words, there are integers $1 \leq a_1 < a_2 < \cdots < a_{|F|} \leq |X| + 1$ which are the minimum numbers such that $f_i(a_i) = 1$ (with the convention that $a_{|F|} = |X| + 1$ if $f_{|F|}$ is identically 0) and satisfy the following property for each $1 \leq i \leq |F|$:

- If $x > a_i$, then $f_i(x) = 1$.
- If $x < a_i$, then $f_i(x) = 0$.

Remark 2. Let F be non-decreasing. For a function $f \in F$ and any choice of r inputs from the set X, there are at most $r + 1$ possible values of the corresponding outputs $(f(x_1), \ldots, f(x_r))$, namely $(0, 0, \ldots, 0, 0)$, $(0, 0, \ldots, 0, 1)$, …, $(0, 1, \ldots, 1, 1)$, and $(1, 1, \ldots, 1, 1)$.

2.2 Bounds

The following theorem establishes an upper bound on $\text{opt}_{\text{weak}}(\text{CART}_r(F))$ by illustrating a possible learner strategy.

Theorem 3. *For non-decreasing F (as defined in Section 2.1),*

$$\text{opt}_{\text{weak}}(\text{CART}_r(F)) < (r + 1) \cdot \ln(|F|) = (1 + o(1))r \ln(|F|).$$

Proof. The learner's strategy: pick the answer that corresponds to the most functions. By Remark 2, we know that there are at most $r + 1$ such answers.

Each time the adversary says "NO," if the learner previously knew that there were T possible functions left, the learner is then able to reduce the number of possible functions left by at least

$$\frac{T}{\#\text{ of remaining possible answers to query}} \geq \frac{T}{r+1}.$$

Thus, each answer of "NO" means that the number of remaining possibilities is multiplied by $\frac{r}{r+1}$.

Then, the learner will make at most

$$\log_{\frac{r+1}{r}}(|F|) = \frac{\ln(|F|)}{\ln\left(1 + \frac{1}{r}\right)} < \frac{\ln(|F|)}{\frac{\frac{1}{r}}{1+\frac{1}{r}}} = (r + 1) \cdot \ln(|F|)$$

mistakes, as desired. □

We establish a lower bound on $\text{opt}_{\text{weak}}(\text{CART}_r(F))$ as well by using a possible adversary strategy.

Theorem 4. *For non-decreasing F (as defined in Section 2.1),*

$$\text{opt}_{\text{weak}}(\text{CART}_r(F)) \geq (1 - o(1))r \ln(|F|).$$

Proof. We outline the following strategy for the adversary: Each round, the adversary will say "NO"; when they do this, some functions fail to remain consistent with the answers given by the adversary. Let $F' \subseteq F$ be the current set of functions that are still consistent with the answers that the adversary has given so far. Furthermore, with $F' = \{g_1, g_2, \ldots, g_{|F'|}\}$, define $1 \leq b_1 < b_2 < \cdots < b_{|F'|} \leq |X|+1$ as the minimum numbers such that $g_i(b_i) = 1$ (with the convention that $b_{|F'|} = |X| + 1$ if $f_{|F'|}$ is identically 0).

In each round, the adversary will choose the inputs

$$x_i = b_{i \cdot \left\lceil \frac{|F'|}{r+1} \right\rceil + 1}$$

for $1 \leq i \leq r$. Then, no matter what the learner says, the adversary says "NO." This guarantees that the number of remaining consistent functions decreases by at most $\left\lceil \frac{|F'|}{r+1} \right\rceil$ functions per round. Thus, the adversary can continue for at least

$$(1 - o(1)) \log_{\frac{r+1}{r}}(|F|) \geq (1 - o(1)) \cdot \frac{\ln(|F|)}{\ln\left(1 + \frac{1}{r}\right)} \geq (1 - o(1)) r \ln(|F|)$$

turns, using the bound that $\ln(1 + k) \leq k$ for all $k \geq -1$. Therefore, the adversary guarantees that they can say "NO" at least $(1 - o(1))r \ln(|F|)$ times, as desired. □

Combining the above bounds, this implies that for non-decreasing F, we have the following theorem:

Theorem 5. *For non-decreasing F (as defined in Section 2.1),*

$$\mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F)) = (1 \pm o(1)) r \ln(|F|).$$

3 Bounds on $\mathrm{opt}_{\mathrm{amb},r}(F)$

In this section, we establish upper and lower bounds on $\mathrm{opt}_{\mathrm{amb},r}(F)$ that are within a constant factor of each other for non-decreasing F (defined in Section 2.1). We show that $\mathrm{opt}_{\mathrm{amb},r}(F) = (1 - o(1))2^r \ln(|F|)$ (Theorem 8). The upper bound in this section is applicable to all families of functions F, but the lower bound is only applicable for the sets of functions F as described in Section 2.1, similar to the bounds established in Section 2.

We prove the following theorem, a general upper bound for $\mathrm{opt}_{\mathrm{amb},r}(F)$, using a learner strategy that does not make any assumptions about the set F (in particular, F does not have to be non-decreasing).

Theorem 6. *For* all *X and F (not limited to the conditions for non-decreasing F as described in Section 2.1),*

$$\mathrm{opt}_{\mathrm{amb},r}(F) < \min(2^r \ln(|F|), |F|).$$

Proof. The learner's strategy to get $\mathrm{opt}_{\mathrm{amb},r}(F) < 2^r \ln(|F|)$: for each input that the adversary gives, pick the answer that corresponds to the most functions which were both known to be possible before the round started and consistent with all earlier guesses in the current round.

Each time the adversary says "NO," we claim that the learner can eliminate at least $\frac{1}{2^r}$ of the functions that he knew were possible before the round, i.e. if the learner knew

that there were T possible functions left before the round started, then he can guarantee that his answer for all r inputs is consistent with at least $\frac{T}{2^r}$ of those functions (the learner would then eliminate these functions when the adversary tells him "NO"). To show this, note that by induction, at least $\frac{T}{2^k}$ of the functions will be consistent with the k answers he has given so far for each $1 \leq k \leq r$, by the definition of his strategy.

Therefore, each time the adversary says "NO" the number of remaining possible functions is multiplied by at most $\frac{2^r-1}{2^r}$. So, the learner makes at most

$$\log_{\frac{2^r}{2^r-1}}(|F|) = \frac{\ln(|F|)}{\ln\left(1+\frac{1}{2^r-1}\right)} < \frac{\ln(|F|)}{\frac{\frac{1}{2^r-1}}{1+\frac{1}{2^r-1}}} = 2^r \cdot \ln(|F|)$$

mistakes with this strategy.

The learner's strategy to get $\mathrm{opt}_{\mathrm{amb},r}(F) \leq |F| - 1$: Each time the adversary says "NO," the learner can eliminate at least 1 function. Once the learner has eliminated $|F|-1$ functions, no more errors will be made. □

To establish a lower bound on $\mathrm{opt}_{\mathrm{amb},r}(F)$, we again need the assumptions about non-decreasing F to demonstrate an adversary strategy.

Theorem 7. *For non-decreasing F (as defined in Section 2.1),*

$$\mathrm{opt}_{\mathrm{amb},r}(F) \geq (1 - o(1))2^r \ln(|F|).$$

Proof. We outline the following strategy for the adversary. In this strategy, the adversary will say "NO" at the end of each round. For each round, the adversary will choose a series of input values x_i based on the answers given by the learner. In each subround, the next input x_i is determined as follows: suppose that S is the set of all functions that are consistent with all previous adversary answers from past rounds as well as all the answers of the learner from the current round.

Since $S \subseteq F$, we can then set $S = \{g_1, g_2, \ldots, g_{|S|}\}$ and define $1 \leq b_1 < b_2 < \cdots < b_{|S|} \leq |X| + 1$ as the minimum numbers such that $g_i(b_i) = 1$ (with the convention that $b_{|S|} = |X| + 1$ if $f_{|S|}$ is identically 0).

The adversary then chooses $x_i = b_{\left\lceil \frac{|S|}{2} \right\rceil}$ for the current subround. This guarantees that at each subround, the number functions consistent with all of the adversary's previous answers as well as all of the learner's answers in the current round reduces by at least $\left\lfloor \frac{|S|}{2} \right\rfloor$, i.e. from $|S|$ to at most $\left\lceil \frac{|S|}{2} \right\rceil$. Thus, if T functions were consistent with all of the adversary's previous answers at the beginning of the current round, then at the end of the round, at most $\left\lceil \frac{T}{2^r} \right\rceil$ becoming inconsistent with the adversary's answers (by repeatedly using the fact that $\left\lceil \frac{\lceil x \rceil}{n} \right\rceil = \left\lceil \frac{x}{n} \right\rceil$ for all positive reals x and positive integers n).

This means that the adversary can continue to say "NO" for at least

$$\begin{aligned}(1 - o(1)) \log_{\frac{2^r}{2^r-1}}(|F|) &= (1 - o(1)) \frac{\ln(|F|)}{\ln\left(1+\frac{1}{2^r-1}\right)} \\ &\geq (1 - o(1)) \frac{\ln(|F|)}{\frac{1}{2^r-1}} = (1 - o(1))2^r \ln(|F|)\end{aligned}$$

turns, as desired. □

Combining the above bounds, this implies that for non-decreasing F, we have the following theorem:

Theorem 8. *For non-decreasing F (as defined in Section 2.1),*

$$\mathrm{opt}_{\mathrm{amb},r}(F) = (1 - o(1))2^r \ln(|F|).$$

Theorem 5 and Theorem 8 imply that for large r and large enough $|F|$ where F is non-decreasing, learners who are given all inputs at the beginning of each round do better exponentially in r than their counterparts who receive inputs one at a time in each round.

This is surprising and illustrates how a small change in the flow of information could have massive ramifications on the efficiency of learning algorithms.

4 Bounds on $\mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F))$ for Multi-class Functions

In this section, we determine an upper and lower bound for $\mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F))$ on multi-class functions. The following two theorems prove these bounds using similar strategies as in Section 2.

Theorem 9. *For a subset of functions F of the non-decreasing functions from X to $\{0, 1, \ldots, k-1\}$,*

$$\mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F)) \leq \binom{r+k-1}{k-1} \ln(|F|).$$

Proof. As in Theorem 3, the learner's strategy will be to pick the answer that corresponds to the most functions. There are at most $\binom{r+k-1}{k-1}$ possible answers to a query, so each answer of "NO" multiplies the number of remaining functions by at most $\frac{\binom{r+k-1}{k-1}-1}{\binom{r+k-1}{k-1}}$.

Thus, learner will make at most

$$\log_{\frac{\binom{r+k-1}{k-1}}{\binom{r+k-1}{k-1}-1}}(|F|) = \frac{\ln(|F|)}{\ln\left(1 + \frac{1}{\binom{r+k-1}{k-1}-1}\right)} \leq \frac{\ln(|F|)}{\frac{\frac{1}{\binom{r+k-1}{k-1}-1}}{1+\frac{1}{\binom{r+k-1}{k-1}-1}}} = \binom{r+k-1}{k-1} \ln(|F|)$$

mistakes, proving that

$$\mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F)) \leq \binom{r+k-1}{k-1} \ln(|F|). \qquad \square$$

Theorem 10. *For a subset of functions F of the non-decreasing functions from X to $\{0, 1, \ldots, k-1\}$,*

$$\mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F)) \geq \frac{1}{2^{k-2}}(1 - o(1))r \ln(|F|).$$

Proof. Let $S_i(f)$ for $1 \leq i \leq k-1$ be the value in X such that $f(x) \leq i$ for $x \leq S_i(f)$ and $f(x) > i$ for $x > S_i(f)$.

Then, let F' be the largest subset of F that can be ordered into $f_1, f_2, \ldots, f_{|F'|}$ such that all sequences $S_i(f_1), S_i(f_2), \ldots S_i(f_{|F'|})$ for $1 \leq i \leq k-1$ are monotonic. Consider

ordering the functions in $|F|$ by increasing order of $S_1(f)$. Then, we want to find the largest subsequence of functions in the ordering such that all other sequences of $S_i(f)$ are monotonic as well. By Lemma 3.1 of *Bounding Sequence Extremal Functions with Formations* [10], $|F'| \geq |F|^{\frac{1}{2^{k-2}}}$.

For $1 \leq i \leq r$, there exists a $x_i \in X$ such that $f_{i\cdot\left\lceil\frac{|F'|}{r+1}\right\rceil}(x_i) \neq f_{i\cdot\left\lceil\frac{|F'|}{r+1}\right\rceil+1}(x_i)$. The adversary can choose these x_i with any additional x-values which can be chosen at random if there are repeat x_i. This splits the $|F'|$ functions about evenly into $r+1$ sections.

Because of the monotonicity of S_i, the learner cannot choose a combination of r values that satisfy two functions in different sections. Thus, the learner can guarantee to eliminate at most $\left\lceil\frac{|F'|}{r+1}\right\rceil$ functions. This situation is identical to the one in Theorem 4. Thus,

$$\text{opt}_{\text{weak}}(\text{CART}_r(F)) \geq (1-o(1))r\ln(|F'|) \geq \frac{1}{2^{k-2}}(1-o(1))r\ln(|F|). \qquad \square$$

5 Bounds on $\text{opt}_{\text{amb},r}(F)$ for Multi-class Functions

In this section, we determine an upper and lower bound for $\text{opt}_{\text{amb},r}(F)$ on multi-class functions. The following two theorems prove these bounds using similar strategies as in Section 3.

Theorem 11. *For* all X *and* F *(not limited to* F *being a set of non-decreasing functions)*

$$\text{opt}_{\text{amb},r}(F) < \min(k^r\ln(|F|), |F|).$$

Proof. The learner's strategy is that for each input that the adversary gives, the should pick the answer that corresponds to the most functions which were both known to be possible before the round started and consistent with all earlier guesses in the current round.

With this strategy the learner can guarantee to eliminate at least $\frac{1}{k^r}$ of the possible functions each round. Then, each time the adversary says "NO", the number of remaining possible functions is multiplied by at most $\frac{k^r-1}{k^r}$. Thus, the learner makes at most

$$\log_{\frac{k^r}{k^r-1}}(|F|) = \frac{\ln(|F|)}{\ln\left(1+\frac{1}{k^r-1}\right)} \leq \frac{\ln(|F|)}{\frac{\frac{1}{k^r-1}}{1+\frac{1}{k^r-1}}} = k^r\cdot\ln(|F|)$$

mistakes, establishing the bound

$$\text{opt}_{\text{amb},r}(F) < k^r\ln(|F|). \qquad \square$$

Remark 12. Similar to Theorem 6, this theorem does not make any assumption about the functions in F and so represents a general learner strategy for any F.

Theorem 13. *For a subset of functions* F *of the non-decreasing functions from* X *to* $\{0,1,\ldots,k-1\}$*,*

$$\text{opt}_{\text{amb},r}(F) \geq (1-o(1))2^{r-k+2}\ln(|F|).$$

Proof. As in subsection 10, we find the largest subset F' of F that can be ordered into $f_1, f_2, \ldots, f_{|F'|}$ such that all sequences $S_i(f_1), S_i(f_2), \ldots S_i(f_{|F'|})$ for $1 \leq i \leq k-1$ are monotonic. The strategy for the adversary is that for each input, we look at the set of functions that are possible before the round started, consistent with all earlier guesses in the current round, and in F'. We label these functions $g_1, g_2, \ldots, g_n$ such that all sequences $S_i(g_1), S_i(g_2), \ldots, S_i(g_n)$ are monotonic. Then, the adversary should pick an x such that $g_{\lfloor \frac{n}{2} \rfloor}(x) \neq g_{\lfloor \frac{n}{2} \rfloor + 1}(x)$. This guarantees that each subround reduces the number of functions consistent with all of the previous answers from n to $\lceil \frac{n}{2} \rceil$. This is the exact same situation as in Theorem 7, giving us the bound

$$\mathrm{opt}_{\mathrm{amb},r}(F) \geq (1 - o(1))2^r \ln(|F'|) \geq (1 - o(1))2^{r-k+2} \ln(|F|). \qquad \square$$

6 Bounds Comparing $\mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F))$ to $\mathrm{opt}_{\mathrm{std}}(F)$

In another paper by Long [13], he compares the standard and bandit model for multi-class functions, and determines a bound on the maximum factor gap between them. There was an error in Long's proof of the lower bound, but Geneson fixes this error in [9]. In this section, we take a generalization of this idea and determine a lower bound of the maximum factor gap between $\mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F))$ and $\mathrm{opt}_{\mathrm{std}}(F)$ using similar techniques as in Long's and Geneson's work. The proof uses techniques previously used for experimental design [16, 15] and hashing, derandomization, and cryptography [14, 4]. We also adapt the upper bound in [13] to show that our lower bound is tight when r is fixed.

Lemma 14. *For any subset $S \subset \{1, \ldots, p-1\}^n$, there are elements $u = (u_1, \ldots, u_r) \in (\{1, \ldots, p-1\}^n)^r$ such that for all $z = (z_1, \ldots, z_r) \in \{1, \ldots, p-1\}^r$,*

$$|\{x \in S : \forall 1 \leq i \leq r, x \cdot u_i \equiv z_i \pmod p\}| \leq \frac{|S|}{p^r} + 2\sqrt{|S|}.$$

Proof. Suppose S is any subset of $\{1, \ldots, p-1\}^n$, and let u be chosen uniformly at random from $(\{1, \ldots, p-1\}^n)^r$. For each $z \in \{1, \ldots, p-1\}^r$, let T_z be the set of $x \in S$ for which $x \cdot u_i = z_i$ for all i. By the linearity of expectation, we have $\mathbb{E}(|T_z|) = \frac{|S|}{p^r}$ for all z.

Consider an arbitrary $z \in \{1, \ldots, p-1\}^r$. For each $s \in S$, define the indicator random variable $X_{s,z}$ such that $X_{s,z} = 1$ if $s \cdot u_i = z_i$ for all i, and $X_{s,z} = 0$ otherwise. If $s, t \in S$ are not multiples of each other $\pmod p$, then $\mathrm{Cov}(X_{s,z}, X_{t,z}) = 0$. If s and t are multiples of each other, then $\mathrm{Cov}(X_{s,z}, X_{t,z}) = \mathbb{E}(X_{s,z}X_{t,z}) - \mathbb{E}(X_{s,z})\mathbb{E}(X_{t,z})$.

If z contains any nonzero z_i, then $\mathbb{E}(X_{s,z}X_{t,z}) = 0$, giving $\mathrm{Cov}(X_{s,z}, X_{t,z}) = -\frac{1}{p^{2r}}$. Thus,

$$\begin{aligned}\mathrm{Var}(|T_z|) = \mathrm{Var}\left(\sum_{s \in S} X_{s,z}\right) &= \sum_{s \in S} \mathrm{Var}(X_{s,z}) + \sum_{s \neq t} \mathrm{Cov}(X_{s,z}, X_{t,z}) \\ &\leq \sum_{s \in S} \mathrm{Var}(X_{s,z}) = |S|\left(\frac{1}{p^r} - \frac{1}{p^{2r}}\right) < \frac{|S|}{p^r}.\end{aligned}$$

By Chebyshev's inequality, $\mathbb{P}\left(|T_z| \geq \frac{|S|}{p^r} + 2\sqrt{|S|}\right) \leq \frac{1}{4p^r}$.

Otherwise, $\mathbb{E}(X_{s,z}X_{t,z}) = \frac{1}{p^r}$, giving $\mathrm{Cov}(X_{s,z}, X_{t,z}) = \frac{1}{p^r} - \frac{1}{p^{2r}} < \frac{1}{p^r}$. Note that there are at most $(p-2)|S|$ ordered pairs (s,t) for which s and t are multiples of each other (mod p) with $s \neq t$. Thus,

$$\begin{aligned}\mathrm{Var}(|T_z|) &= \mathrm{Var}\left(\sum_{s\in S} X_{s,z}\right) = \sum_{s\in S}\mathrm{Var}(X_{s,z}) + \sum_{s\neq t}\mathrm{Cov}(X_{s,z}, X_{t,z}) \\ &\leq \sum_{s\in S}\mathrm{Var}(X_{s,z}) + \frac{(p-2)|S|}{p^r} < \frac{|S|}{p^r} + \frac{(p-2)|S|}{p^r} < \frac{|S|}{p^{r-1}}.\end{aligned}$$

By Chebyshev's inequality, $\mathbb{P}\left(|T_z| \geq \frac{|S|}{p^r} + 2\sqrt{|S|}\right) \leq \frac{1}{4p^{r-1}}$.

By the union bound,

$$\mathbb{P}\left(\forall z : |T_z| \leq \frac{|S|}{p^r} + 2\sqrt{|S|}\right) \geq 1 - \frac{p^r-1}{4p^r} - \frac{1}{4p^{r-1}} = \frac{3p^r - p + 1}{4p^r} \geq \frac{1}{2}.$$

Thus, the conditions are satisfied with probability greater than $\frac{1}{2}$ when u is chosen randomly, so there must always exist a u satisfying the conditions. □

Theorem 15. *For all $M > 2r$ and infinitely many k, there exists a set F of functions from a set X to a set Y with $|Y| = k$ such that $\mathrm{opt}_{\mathrm{std}}(F) = M$ and*

$$\mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F)) \geq (1-o(1))\left(|Y|^r \ln |Y|\right)(\mathrm{opt}_{\mathrm{std}}(F) - 2r).$$

Proof. Fix $n \geq 3$ and $p \geq 5$. For all $a \in \{0,\ldots,p-1\}^n$, we define $f_a : \{0,\ldots,p-1\}^n \to \{0,\ldots,p-1\}$ so that $f_a(x) = a\cdot x \pmod p$ and define $F_L(p,n) = \{f_a : a \in \{0,\ldots,p-1\}^n\}$. It is known that $\mathrm{opt}_{\mathrm{std}}(F_L(p,n)) = n$ for all primes p and $n > 0$ [18, 2, 3, 13].

We now determine a bound on $\mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F_L(p,n)))$. Let $S = \{1,\ldots,p-1\}^n$, so $|S| = (p-1)^n$. Let $R_1 = \{f_a : a \in S\} \subset F_L(p,n)$. In each round $t > 1$, the adversary will create a list R_t of members of $\{f_a : a \in S\}$ that are consistent with its previous answers. It will always answer "NO" and choose $(x_1,\ldots,x_r)$ that minimizes

$$\max_{(y_1,\ldots,y_r)} \left|R_t \cap \{f : \forall 1 \leq i \leq r, f(x_i) = y_i\}\right|.$$

By Lemma 14, we have

$$R_{t+1} \geq |R_t| - \frac{|R_T|}{p^r} - 2\sqrt{|R_t|} \geq |R_t| - \frac{|R_t|}{p^r} - \frac{2|R_t|}{p^r\sqrt{\ln p}} = \left(1 - \frac{1 + \frac{2}{\sqrt{\ln p}}}{p^r}\right)|R_t|$$

as long as $|R_t| \geq p^{2r}\ln p$. Thus, we have $|R_t| \geq \left(1 - \frac{1+\frac{2}{\sqrt{\ln p}}}{p^r}\right)^{t-1}(p-1)^n$. Therefore, whenever $\left(1 - \frac{1+\frac{2}{\sqrt{\ln p}}}{p^r}\right)^{t-1}(p-1)^n \geq p^{2r}\ln p$, the adversary can guarantee t wrong guesses. This is true for $t = (1-o(1))(n-2r)p^r\ln p$, which gives us the desired result. □

Remark 16. Since we have the trivial inequality $\mathrm{opt}_{\mathrm{amb},r}(F) \geq \mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F))$ because the learner has strictly more information in the scenario on the RHS, then for the sets F constructed above, we also have

$$\mathrm{opt}_{\mathrm{amb},r}(F) \geq (1-o(1))\left(|Y|^r \ln |Y|\right)(\mathrm{opt}_{\mathrm{std}}(F) - 2r).$$

Now, we establish a similar upper bound relating $\text{opt}_{\text{weak}}(\text{CART}_r(F))$ and $\text{opt}_{\text{std}}(F)$.

Lemma 17. *For all sets F of functions $f : X \to Y$, we have* $\text{opt}_{\text{std}}(\text{CART}_r(F)) = \text{opt}_{\text{std}}(F)$.

Proof. First, recall that $\text{CART}_r(F)$ is a set of functions $f : X^r \to Y^r$. Note that when the adversary always uses r copies of the same input and asks the same questions as they would in the standard model with F, then the learner is guaranteed to make at least as many mistakes with $\text{CART}_r(F)$ as they would with F. Therefore, $\text{opt}_{\text{std}}(\text{CART}_r(F)) \geq \text{opt}_{\text{std}}(F)$.

On the other hand, consider every time that the learner makes a mistake while learning $\text{CART}_r(F)$ under the standard model. This means that there is some input $x \in X$ for which the learner answers incorrectly and subsequently receives the correct answer for (in addition to the correct answers for the $r-1$ remaining inputs for the round). If the learner ignores the rest of the information and only takes into account the new information about x, then this is equivalent to making a mistake about the input x in the standard model with F. Therefore, there is a way for the learner to reduce learning with $\text{CART}_r(F)$ to learning with F under the standard model, implying that $\text{opt}_{\text{std}}(\text{CART}_r(F)) \leq \text{opt}_{\text{std}}(F)$.

The lemma is then proven by putting these two bounds together. □

Now, in [13], Long proved the upper bound

$$\text{opt}_{\text{weak}}(F) \leq (1 + o(1))(|Y| \ln |Y|) \, \text{opt}_{\text{std}}(F).$$

We will use Lemma 17 to adapt this to the modified weak reinforcement model.

Theorem 18. *For any set F of functions from some set X to $\{0, 1, \ldots, k-1\}$ and for any $r \geq 1$,*

$$\text{opt}_{\text{weak}}(\text{CART}_r(F)) \leq (1 + o(1)) \left(|Y|^r r \ln |Y|\right) \text{opt}_{\text{std}}(F).$$

Proof. Substituting $\text{CART}_r(F)$ for F (and therefore setting Y^r in place of Y) in the upper bound from [13] and using Lemma 17, we get

$$\begin{aligned} \text{opt}_{\text{weak}}(\text{CART}_r(F)) &\leq (1 + o(1))(|Y|^r \ln (|Y|^r)) \, \text{opt}_{\text{std}}(\text{CART}_r(F)) \\ &= (1 + o(1))(|Y|^r r \ln |Y|) \, \text{opt}_{\text{std}}(F), \end{aligned}$$

as desired. □

Remark 19. With r fixed, Theorems 15 and 18 demonstrate that the upper bound in Theorem 18 is sharp up to a constant factor.

7 Relating Different Learning Scenarios

Let $\text{opt}(F)$ be the worst-case number of mistakes that the learner will make in some learning scenario (e.g. weak reinforcement learning, strong reinforcement learning, or delayed, ambiguous reinforcement learning). The three aforementioned choices would have $\text{opt}(F) = \text{opt}_{\text{weak}}(F)$, $\text{opt}(F) = \text{opt}_{\text{strong}}(F)$, and $\text{opt}(F) = \text{opt}_{\text{amb},r}(F)$, respectively.

In this section, we first prove Theorem 23, which states that there is a way for the adversary to formulate their questions such that the learner is guaranteed to perform with

at least the worst-case number $\text{opt}(F)$ of mistakes, and that all these mistakes happen in the first $\text{opt}(F)$ questions. This theorem is very general and applies to all learning scenarios.

Definition 20 (Streetegy)**.** A *streetegy* is a k-nary tree which represents a possible adversary's strategy as follows: each node represents an input from X that the adversary can ask the learner. The root is the first input that the adversary gives the learner, and each node has $k = |Y|$ children: one for each possible element of Y that the learner could answer. Each edge from parent to child is also marked by an ordered pair (L, A) where $L, A \in Y$. If the learner answers L, then the adversary will answer A (and the learner has made a mistake if $L \neq A$), and continue with the strategy by following the edge. The streetegy is truncated below node N if all nodes N' in the subtree under N have an edge coming out of it with $L = A$ (this means that it is possible for the learner to answer correctly for all nodes under N, so the learner may be able to avoid any future mistakes). Finally, the streetegy's edges must be consistent; i.e. for every path along the streetegy, all pairs (L, A) from the edges along that path can be represented by a single function from the family F.

Definition 21 (Optimal Streetegy)**.** An *optimal streetegy* is a streetegy such that all leaves represent states where the learner has made at least $\text{opt}(F)$ mistakes.

Definition 22 (Depth)**.** The *depth* of a streetegy is the maximum length of a path in the streetegy that an optimal learner could possibly follow.

It is clear by definition that all optimal streetegies have depth at least $\text{opt}(F)$, since the paths must be at least length $\text{opt}(F)$ to have that many mistakes.

Theorem 23. *There exists an optimal streetegy for the adversary with depth exactly* $\text{opt}_{\text{strong}}(F)$.

Proof. Suppose for sake of contradiction that there was not. Consider some optimal streetegy with minimal depth (which exists by the Well Ordering Principle), and consider any path (that an optimal learner could possibly follow) from the root to a leaf with length larger than $\text{opt}(F)$ (since by assumption, the minimal depth is strictly greater than $\text{opt}(F)$). Since the optimal learner makes at most $\text{opt}(F)$ mistakes, an optimal learner is guaranteed to have at least one correct answer along this path. This means that there is at least one node N along this path such that one of the edges protruding from it satisfies $L = A$; let this edge point to node N'. Replace the subtree rooted at N (including N) with the subtree rooted at N' (including N'). This does not violate the validity (consistency of edges) or the optimality of the streetegy. It also decreases the length of the optimal learner's possible path by 1. If we repeat this for all possible paths of length larger than $\text{opt}(F)$, we will have constructed a new streetegy with strictly lesser depth, contradiction. □

Inspired by this commonality between different learning scenarios, we establish a bound relating the number of mistakes made in two different situations using a novel technique and provide a stronger conjecture which we believe is true.

In what follows, let $\text{opt}_1(F)$ be the worst case number of errors using strong reinforcement learning, where the learner receives the correct answer after each round of learning (so it is another name for $\text{opt}_{\text{strong}}(F)$ from [1] and $\text{opt}_{\text{std}}(F)$ from [13]), and let $\text{opt}_2(F)$ be the

worst case expected number of errors using weak reinforcement learning with a possibly non-deterministic learning algorithm, where the learner only receives an indication of whether their guess was correct or not after each round of learning (this corresponds to $\text{B-Err}_H^r(T)$ in [7]).

Proposition 24. *For all families of functions $F : X \to \{1, 2, \ldots, k\}$, the following inequality relating $\text{opt}_1(F)$ and $\text{opt}_2(F)$ holds:*

$$\text{opt}_2(F) \leq (k + |X| - 1)\,\text{opt}_1(F).$$

Proof. Let S be an optimal strategy which guarantees at most $\text{opt}_1(F)$ mistakes in the learning process with strong reinforcement learning. We construct a strategy S' for the scenario of weak reinforcement learning which guarantees at most $(k + |X| - 1)\,\text{opt}_1(F)$ mistakes, average. The strategy works in phases: each phase mimics the process of S making one mistake, and receiving the correct answer.

For the first phase, S' would guess whatever S would when given no information; if an input is repeated, then it will then guess randomly among the remaining $k - 1$ possible answers. Clearly, there will be at most $k + |X| - 1$ mistakes, on average, before the learner guesses an answer correctly which was not what S would have guessed. This is then used a the first "mistake" of S, and ends the phase.

For the next phase, S' would guess what S would when given the information from the first "mistake." Continuing in this manner, since we know that there are at most $\text{opt}_1(F)$ phases by definition of S, and that each phase has at most $k + |X| - 1$ mistakes on average, this implies that

$$\text{opt}_2(F) \leq (k + |X| - 1)\,\text{opt}_1(F),$$

as desired. □

Remark 25. This way of relating the number of mistakes in different learning situations revolves around adapting the strategies of one situation to another. This involves similar thinking to the way that more general results like Theorem 23 are proven (i.e. thinking about tree-like structures and how the learner can mimic movement on the tree for one scenario in a different learning environment). We believe this method of proof can be made to work for our conjecture below.

In fact, we believe that the following stronger statement is true, which eliminates $|X|$ from the RHS:

Conjecture 26. For all families of functions $F : X \to \{1, 2, \ldots, k\}$, the following inequality relating $\text{opt}_1(F)$ and $\text{opt}_2(F)$ holds:

$$\text{opt}_2(F) \leq (k - 1)\,\text{opt}_1(F).$$

For the $k = 2$ case, we have equality, which is expected. We believe that a variant of the above proof of the weaker proposition could work for this conjecture. This result would be related to the established result from [13] that

$$\text{opt}_{\text{bandit}}(F) \leq (1 + o(1))(k \ln k)\,\text{opt}_{\text{std}}(F),$$

the difference being that the LHS of this bound is for deterministic algorithms while we allow non-deterministic algorithms in the LHS of our conjecture (and measure worst case *expected number* of mistakes in $\text{opt}_2(F)$).

8 New Models for Permutation Functions

We define and explore new models where the family of possible functions F is a set of permutations of length n and where the leaner tries to guess information about the relative orders of inputs.

8.1 The Order Model

We define a new variant model called the order model.

Definition 27. In the *order model*, for a set F of permutations of n numbers, the learner tries to guess a permutation function $f \in F$. On each turn, the adversary chooses r inputs to f (in other words a set $S \subseteq [n], |S| = r$). The learner guesses the sub-permutation of f corresponding to the outputs.

Under weak reinforcement, the adversary informs the learner if they made a mistake, and under strong reinforcement, the adversary gives the correct sub-permutation to the learner. We denote the worst-case amount of mistakes for the learner with weak reinforcement as $\text{opt}_{\text{weak}}(\text{PERM}_r(F))$ and with strong reinforcement as $\text{opt}_{\text{strong}}(\text{PERM}_r(F))$. Note that $\text{opt}_{\text{strong}}(\text{PERM}_r(F)) \leq \text{opt}_{\text{weak}}(\text{PERM}_r(F))$, and that if $r = 2$, strong and weak reinforcement are identical, so equality holds.

We first find an upper bound by presenting a strategy for the learner, analogous to Theorem 6 for the order model.

Theorem 28. *For $n > 1$, $\text{opt}(\text{PERM}_r(F)) < r! \ln |F|$.*

Proof. For each input that the adversary gives, the learner can pick the answer that corresponds to the most possible permutations. After each incorrect guess, at least $\frac{1}{r!}$ of the previously possible permutations get eliminated. Therefore, the number of incorrect guesses, and consequently the number of mistakes, is at most

$$\ln_{\frac{r!}{r!-1}} |F| < r! \ln |F|.$$

□

We now find a bound on $\text{opt}(\text{PERM}_2(S_n))$.

Definition 29. For $n \geq 1$, let $v_n := \lfloor \log_2 n \rfloor$ be the largest integer such that $2^{v_n} \leq n$. We define $p(n) := \sum_{1 \leq m \leq n} v_m$ to be the cumulative sum of v_n.

Lemma 30. *For $n \geq 1$, $p(n) = (n+1)v_n - 2(2^{v_n} - 1)$.*

Proof. We show this by induction. The base case $n = 1$ holds.

If n is not a power of two, $v_{n-1} = v_n$, so

$$p(n) = p(n-1) + v_n = (nv_n - 2(2^{v_n} - 1)) + v_n = (n+1)v_n - 2(2^{v_n} - 1),$$

as desired.

Otherwise, $n = 2^n$ is a power of two, so $v_{n-1} = v_n - 1$ and

$$p(2^{v_n}) = p(2^{v_n} - 1) + v_n = (2^{v_n}(v_n - 1) - 2(2^{v_n - 1} - 1)) + v_n = (2^{v_n} + 1)v_n - 2(2^{v_n} - 1),$$

as desired. □

Using these properties, we now present two strategies for the adversary that both achieve a lower bound of $p(n)$.

Theorem 31. *The adversary can achieve* $\text{opt}(\text{PERM}_2(S_n)) \geq p(n)$ *under the order model.*

Proof. The first strategy resembles quicksort. The adversary chooses all pairs of inputs (i, j) for $i > j$ in lexicographically decreasing order, and whatever the learner says, the adversary always responds "NO" if it is possible. We use strong induction to show the desired bound. The base case of $n = 0$ holds.

After guessing the permutations for all pairs of the form (n, j), the learner knows what $f(n)$ is and which of the other $n - 1$ numbers have smaller values. After this point, the learner cannot make a mistake on a comparison between a smaller number and a larger number. Thus, the learner will at most be able to guarantee

$$(n-1)+\text{opt}(\text{PERM}_2(S_{f(n)-1}))+\text{opt}(\text{PERM}_2(S_{n-f(n)})) \geq (n-1)+p(f(n)-1)+p(n-f(n))$$

mistakes in total. Because $p(n)$ is convex, this is minimized at $f(n) = \left\lceil \frac{n}{2} \right\rceil$. Plugging this in gives

$$\begin{aligned}(n-1)+p(f(n)-1)+p(n-f(n)) &= (n-1)+p\Big(\left\lceil\frac{n}{2}\right\rceil-1\Big)+p\Big(n-\left\lceil\frac{n}{2}\right\rceil\Big)\\ &= (n-1)+p\left(\left\lfloor\frac{n-1}{2}\right\rfloor\right)+p\Big(\left\lfloor\frac{n}{2}\right\rfloor\Big).\end{aligned}$$

If n is not a power of two, $v_{\lfloor \frac{n-1}{2} \rfloor} = v_{\lfloor \frac{n}{2} \rfloor} = v_n - 1$, so this becomes

$$\begin{aligned}(n-1)+p(f(n)-1)+p(n-f(n)) &= (n-1)\\ &\quad+\left(\left(\left\lfloor\frac{n-1}{2}\right\rfloor+1\right)(v_n-1)-2(2^{v_n-1}-1)\right)\\ &\quad+\left(\left(\left\lfloor\frac{n}{2}\right\rfloor+1\right)(v_n-1)-2(2^{v_n-1}-1)\right)\\ &= (n-1)+(n+1)(v_n-1)-2(2^{v_n}-2)\\ &= (n-1)-(n+1)+(n+1)v_n-2(2^{v_n}-1)+2\\ &= (n+1)v_n-2(2^{v_n}-1)\\ &= p(n),\end{aligned}$$

as desired. Otherwise, $n = 2^{v_n}$ is a power of two, so $v_{\lfloor \frac{n-1}{2} \rfloor} + 1 = v_{\lfloor \frac{n}{2} \rfloor} = v_n - 1$ and this becomes

$$\begin{aligned}(n-1)+p(f(n)-1)+p(n-f(n)) &= (2^{v_n}-1)+p\big(2^{v_n-1}-1\big)+p\big(2^{v_n-1}\big)\\ &= (2^{v_n}-1)\\ &\quad+2^{v_n-1}(v_n-2)-2(2^{v_n-2}-1)\\ &\quad+(2^{v_n-1}+1)(v_n-1)-2(2^{v_n-1}-1)\\ &= 2^{v_n}v_n+v_n-4\cdot 2^{v_n-1}+2\\ &= (2^{v_n}+1)v_n-2(2^{v_n}-1)\\ &= p(n),\end{aligned}$$

as desired.

A second strategy resembles the insertion sort. The adversary withholds any inquiries about $f(i)$ until the order of the smaller inputs $j < i$ is known. We use induction to show the desired bound. The base case of $n = 0$ holds.

For $n > 0$, by the inductive hypothesis, the adversary can force the learner to make at least $p(n-1)$ mistakes without learning anything about $f(n)$. Assume without loss of generality that $f(1), \ldots, f(n-1)$ are in increasing order.

The adversary then prolongs the learner from finding the position of $f(n)$ by making the learner do a binary search. Specifically, if at some point the learner's bounds on $f(n)$ are $a < f(n) \leq b$, the adversary asks about (n, m) where m is $\frac{a+b}{2}$ rounded to the nearest integer. In this way, the adversary ensures that the number of possible values for $f(n)$ is at least $\left\lfloor \frac{b-a}{2} \right\rfloor$ after the learner's guess. Since the number of possible values starts at n, the learner will at most be able to guarantee v_n mistakes to find the value of $f(n)$. Thus,

$$\operatorname{opt}(\operatorname{PERM}_2(S_n)) \geq \operatorname{opt}(\operatorname{PERM}_2(S_{n-1})) + v_n \geq p(n-1) + v_n = p(n)$$

as desired. □

We can also get a bound for $r > 2$ with a strategy that resembles merge sort.

Theorem 32. *For general r, the adversary can achieve*

$$\operatorname{opt}(\operatorname{PERM}_r(S_n)) \geq (1 - o(1))(r-1)! n \log_r n.$$

Proof. We use strong induction on n.

First, the adversary divides $[n]$ into $\left\lfloor \frac{n}{r} \right\rfloor$ sets S_i each of size r (ignoring any remaining elements). For each i, the adversary repeatedly asks for ordering of S_i, saying "NO" each time until the order is known by the learner. This takes a total of $\left\lfloor \frac{n}{r} \right\rfloor (r! - 1)$ mistakes.

Then, the adversary forms sets C_j from the relative jth elements of each set, and uses the induction hypothesis on n; note that knowing any of C_j's orders does not eliminate possibilities for the orders of other C_j. This gives a recursion of the form

$$\operatorname{opt}(\operatorname{RPOS}_r(S_n)) \geq (1-o(1))n(r-1)! + r \operatorname{opt}(\operatorname{PERM}_r(S_{\lfloor \frac{n}{r} \rfloor})) = (1-o(1))(r-1)! n \log_r(n),$$

as desired. □

8.2 The Relative Position Model

We define a third variant called the relative position model.

Definition 33. In the *relative position model*, for a set F of permutations of n numbers, the learner tries to guess a permutation function $f \in F$. On each turn, the adversary chooses a set S of r inputs to f and an element $x \notin S$, and asks about the pair (x, S). The learner guesses the position of x in the subpermutation of f corresponding to $\{x\} \cup S$.

Under weak reinforcement, the adversary informs the learner if they made a mistake, and under strong reinforcement, the adversary gives the correct position to the learner. We denote the worst-case amount of mistakes for the learner with weak reinforcement as $\operatorname{opt}_{\text{weak}}(\operatorname{RPOS}_r(F))$ and with strong reinforcement as $\operatorname{opt}_{\text{strong}}(\operatorname{RPOS}_r(F))$. Note that $\operatorname{opt}_{\text{strong}}(\operatorname{RPOS}_r(F)) \leq \operatorname{opt}_{\text{weak}}(\operatorname{RPOS}_r(F))$, and that if $r = 1$, both sides of the equation are equal to $\operatorname{opt}(\operatorname{PERM}_2(S_n)) = \operatorname{opt}(S_n)$.

We again imitate Theorem 6 for the relative position model to obtain an upper bound.

Theorem 34. *If $F \subseteq S_n$, $\text{opt}(\text{RPOS}_r(F)) < (r+1)\ln|F|$.*

Proof. For each input that the adversary gives, the learner can pick the answer that corresponds to the most possible permutations. After each incorrect guess, at least $\frac{1}{r+1}$ of the previously possible permutations get eliminated. Therefore, the number of incorrect guesses, and consequently the number of mistakes, is at most $\log_{\frac{r+1}{r}}(|F|) < (r+1)\ln|F|$. □

We generalize the insertion sort proof of Theorem 31 to obtain the following result:

Theorem 35. *The adversary can achieve*

$$\text{opt}(\text{RPOS}_r(S_n)) \geq (1-o(1))rn\ln n.$$

under the relative position model.

Proof. We present a strategy that resembles the insertion sort by strong induction on n. The adversary withholds any inquiries about $f(i)$ until the order of the smaller inputs $j < i$ is known. We use induction to show the desired bound. The base case of $n = 0$ holds.

For $n > 0$, by the inductive hypothesis, the adversary can force the learner to make at least $p(n)$ mistakes without learning anything about $f(n+1)$. Assume without loss of generality that $f(1), \ldots, f(n)$ are in increasing order. Let F_i be the function on $[n]$ that maps m to 1 if $m \geq i$ and 0 otherwise. Note that these functions are non-decreasing.

The problem is equivalent to the learner guessing $F_{f(n+1)}$, where the adversary queries r values at a time. Thus by Theorem 4, the adversary can force the learner to make at least $\text{opt}_{\text{weak}}(\text{CART}_r(F)) = (1-o(1))r\ln n$ mistakes, as desired. □

In a similar fashion, we can prove a similar result for pattern-avoiding permutations:

Theorem 36. *If S_π is the set of π-avoiding permutations of length n,*

$$\text{opt}(\text{RPOS}_r(S_\pi)) \geq (1-o(1))rn\ln(|\pi|-1),$$

where $|\pi|$ denotes the length of the permutation pattern.

Proof. The adversary withholds any inquiries about $f(i)$ until the order of the smaller inputs $j < i$ is known. We use induction to show the desired bound. The base case of $n \leq k$ holds.

For $n > 0$, by the inductive hypothesis, the adversary can force the learner to make at least $p(n)$ mistakes without learning anything extra about $f(n+1)$. Let N be the set of possible values of $f(n+1)$. Any number less than $\pi(k)$ or more than $n-k+1+\pi(k)$ must be in N as it would not be able to form the permutation pattern π, so $|N| \geq \pi(k)+k-1-\pi(k) = k-1$. Let the elements of N in increasing order be $s_1, \ldots, s_{k-1}$.

Let F_i be the function on $[k-1]$ that maps m to 1 if $m \geq s_i$ and 0 otherwise. Note that these functions are non-decreasing.

The problem is equivalent to the learner guessing $F_{f(n+1)}$, where the adversary queries r values at a time. Thus by Theorem 4, the adversary can force the learner to make at least

$$\mathrm{opt}_{\mathrm{weak}}(\mathrm{CART}_r(F)) = (1 - o(1))r \ln |N| \geq (1 - o(1))r \ln(k-1)$$

mistakes, as desired. □

When $\pi = I_k$, the identity permutation, the size of F is $(k-1)^{O(n)}$ [17, 8]. Combining Theorem 36 with 34 gives

Corollary 37. *For F the family of I_k-avoiding permutations,* $\mathrm{opt}(\mathrm{RPOS}_r(F)) = \Theta(rn \ln k)$.

8.3 The Delayed Relative Position Model

We define delayed reinforcement for the relative position model:

Definition 38. In the *delayed relative position model*, for a set F of permutations of n numbers, the learner tries to guess a permutation function $f \in F$. On each turn, the adversary picks an input x and proceeds to give the r elements of a set S one by one (with the requirement that $x \notin S$). After each of the adversary's inquiries, the learner guesses either "HIGHER" or "LOWER". After each round, the learner's final guess for the relative position of x in S is one plus the number of times they said "HIGHER."

Under weak reinforcement, the adversary informs the learner if their final guess is incorrect, and under strong reinforcement, the adversary gives the correct position to the learner. We denote the worst-case amount of mistakes for the learner with weak reinforcement as $\mathrm{opt}_{\mathrm{wrpos},r}(F)$ and with strong reinforcement as $\mathrm{opt}_{\mathrm{srpos},r}(F)$. Note that $\mathrm{opt}_{\mathrm{srpos},r}(F) \leq \mathrm{opt}_{\mathrm{wrpos},r}(F)$, and that if $r = 1$, both sides of the equation are equal to $\mathrm{opt}(\mathrm{PERM}_2(S_n)) = \mathrm{opt}(S_n)$.

We state the analogs of the relative position model results for the delayed version:

Theorem 39. *Given a set of permutations $F \subseteq S_n$ and a positive integer r, these analogs of the results in Section 8.2 hold:*

- *Analog of Theorem 34:* $\mathrm{opt}_{\mathrm{wrpos},r}(F) < 2^r \ln |F|$.
- *Analog of Theorem 35: The adversary can achieve*

  $$\mathrm{opt}_{\mathrm{wrpos},r}(F) \geq (1 - o(1))2^r n \ln n.$$

  under the relative position model.
- *Analog of Theorem 36: If S_π is the set of π-avoiding permutations of length n,*

  $$\mathrm{opt}(\mathrm{RPOS}_r(S_\pi)) \geq (1 - o(1))2^r n \ln(|\pi| - 1),$$

  where $|\pi|$ denotes the length of the permutation pattern.
- *Analog of Corollary 37:* $\mathrm{opt}(\mathrm{RPOS}_r(F)) = \Theta(2^r n \ln k)$, *where F is the family of I_k-avoiding permutations.*

The proofs of these results are nearly identical to those of the previous section, with $(r+1)$ usually replaced by 2^r and cited results from Section 2 replaced with analogous results from Section 3, so in the interest of concision we omit them here.

9 Future Work

In this final section, we outline some possible areas for future work based on the results in our paper.

In Sections 2 and 3, we mainly focused on non-decreasing F (as defined in Section 2.1) for the bounds. What bounds can be obtained for general sets of functions F?

In Sections 4 and 5, we determined some bounds for the models on non-decreasing multi-class functions. Are there tighter bounds? What bounds can be obtained for general sets of functions?

In Section 6, we determined a lower and upper bound on the maximum factor gap between the modified weak reinforcement model and the standard model. Can we establish a stricter upper bound by using the strategy found in Long's paper [13]? Can similar lower and upper bounds on the maximum factor gap between the delayed, ambiguous model and the standard model be established?

In Section 7, we proved Lemma 24, which stated that

$$\operatorname{opt}_2(F) \leq (k + |X| - 1) \operatorname{opt}_1(F).$$

As in Conjecture 26, we believe that

$$\operatorname{opt}_2(F) \leq (k - 1) \operatorname{opt}_1(F)$$

is true. Furthermore, what bounds can we establish between other pairs of learning scenarios (such as the delayed, ambiguous reinforcement learning scenario)?

In Section 8, Theorem 32, we described an adversary strategy for the order model that gives a bound of

$$\operatorname{opt}(\operatorname{PERM}_r(S_n)) \geq (1 - o(1))(r - 1)!n \log_r n,$$

whereas Theorem 28 gives an upper bound of

$$\operatorname{opt}(\operatorname{PERM}_r(S_n)) < r! \ln n! = (1 - o(1))r!n \ln n.$$

We conjecture that the latter bound is tight, i.e. that $\operatorname{opt}(\operatorname{PERM}_r(S_n)) = (1-o(1))r!n \ln n$.

Acknowledgments

We thank our mentor, Professor Jesse Geneson of San José State University, who inspired us to pursue this topic and has guided us throughout the project. We are grateful for his valuable insights and advice on this paper. We would also like to thank the MIT PRIMES-USA program and everyone involved in providing us with this research opportunity.

References

[1] Peter Auer and Philip M. Long. "Structural Results About On-line Learning Models With and Without Queries". In: *Machine Learning* 36 (3 Sept. 1999), pp. 147–181. DOI: 10.1023/A:1007614417594.

[2] Peter Auer, Philip M. Long, Wofgang Maass, and Gerhard J. Woeginger. "On the complexity of function learning". In: *Machine Learning* 18 (2 Feb. 1995), pp. 187–230. ISSN: 1573-0565. DOI: 10.1007/BF00993410.

[3] Avrim Blum. "On-Line Algorithms in Machine Learning". In: *Online Algorithms: The State of the Art*. Ed. by Amos Fiat and Gerhard J. Woeginger. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 306–325. ISBN: 978-3-540-68311-7. DOI: 10.1007/BFb0029575. URL: https://doi.org/10.1007/BFb0029575.

[4] J. Lawrence Carter and Mark N. Wegman. "Universal Classes of Hash Functions (Extended Abstract)". In: *Proceedings of the Ninth Annual ACM Symposium on Theory of Computing*. STOC '77. Boulder, Colorado, USA: Association for Computing Machinery, 1977, pp. 106–112. ISBN: 9781450374095. DOI: 10.1145/800105.803400. URL: https://doi.org/10.1145/800105.803400.

[5] K. Crammer and C. Gentile. "Multiclass classification with bandit feedback using adaptive regularization". In: *Machine Learning* 90 (2012), pp. 347–383.

[6] Varsha Dani, Thomas Hayes, and Sham Kakade. "Stochastic Linear Optimization under Bandit Feedback." In: Jan. 2008, pp. 355–366.

[7] Amit Daniely and Tom Helbertal. "The price of bandit information in multiclass online classification". In: *CoRR* abs/1302.1043 (2013). arXiv: 1302.1043. URL: http://arxiv.org/abs/1302.1043.

[8] Jacob Fox. "Stanley-Wilf limits are typically exponential". In: (2013). arXiv: 1310.8378 [math.CO].

[9] Jesse Geneson. "A note on the price of bandit feedback for mistake-bounded online learning". In: *CoRR* abs/2101.06891 (2021). arXiv: 2101.06891. URL: https://arxiv.org/abs/2101.06891.

[10] Jesse Geneson, Rohil Prasad, and Jonathan Tidor. "Bounding Sequence Extremal Functions with Formations". In: *The Electronic Journal of Combinatorics* 21 (3 2014). DOI: 10.37236/3663. URL: https://www.combinatorics.org/ojs/index.php/eljc/article/view/v21i3p24.

[11] Elad Hazan and Satyen Kale. "NEWTRON: An efficient bandit algorithm for online multiclass prediction". English (US). In: *Advances in Neural Information Processing Systems* (Dec. 2011). 25th Annual Conference on Neural Information Processing Systems 2011, NIPS 2011 ; Conference date: 12-12-2011 Through 14-12-2011, pp. 891–899.

[12] Nick Littlestone. "Learning Quickly When Irrelevant Attributes Abound: A New Linear-Threshold Algorithm". In: *Machine Learning* 2 (4 Apr. 1988), pp. 285–318. ISSN: 1573-0565. DOI: 10.1023/A:1022869011914.

[13] Philip M. Long. "New bounds on the price of bandit feedback for mistake-bounded online multiclass learning". In: *Theoretical Computer Science* 808 (2020). Special Issue on Algorithmic Learning Theory, pp. 159–163. ISSN: 0304-3975. DOI: https://doi.org/10.1016/j.tcs.2019.11.017. URL: https://www.sciencedirect.com/science/article/pii/S0304397519307297.

[14] Michael Luby and Avi Wigderson. "Pairwise Independence and Derandomization". In: *Found. Trends Theor. Comput. Sci.* 1.4 (Aug. 2006), pp. 237–301. ISSN: 1551-305X. DOI: 10.1561/0400000009. URL: https://doi.org/10.1561/0400000009.

[15] C. R. Rao. "Factorial Experiments Derivable from Combinatorial Arrangements of Arrays". In: *Supplement to the Journal of the Royal Statistical Society* 9.1 (1947), pp. 128–139. DOI: 10.2307/2983576.

[16] C. R. Rao. "Hypercubes of strength "d" leading to confounded designs in factorial experiments". In: *Bulletin of the Calcutta Mathematical Society* (38 1946), pp. 67–68. ISSN: 0008-0659.

[17] Amitai Regev. "Asymptotic values for degrees associated with strips of young diagrams". In: *Advances in Mathematics* 41.2 (1981), pp. 115–136. ISSN: 0001-8708. DOI: https://doi.org/10.1016/0001-8708(81)90012-8. URL: https://www.sciencedirect.com/science/article/pii/0001870881900128.

[18] H. Shvaytser. "Linear manifolds are learnable from positive examples". Unpublished manuscript. 1988.