Multi-task Learning with a Shared Annotator

Haim Cohen

Multi-task Learning with a Shared Annotator

Research Thesis

In Partial Fulfillment of The
Requirements for the Degree of
Master of Science in Electrical Engineering

Haim Cohen

Submitted to the Senate of
the Technion — Israel Institute of Technology
Tevet, 5776 Haifa December 2015

The Research Thesis Was Done Under The Supervision of Prof. Koby Crammer in the Faculty of Electrical Engineering.

Publications

Parts of this work were published in
Weighted Multi-task Learning with a Shared Annotator.
Haim Cohen and Koby Crammer. NIPS 2014.

Contents

List of Figures

List of Tables

Abstract

We introduce a new multi-task framework, in which K online learners are sharing a single annotator with limited bandwidth. On each round, each of the K learners receives an input, and makes a prediction about the label of that input. Then, a shared (stochastic) mechanism decides which of the K inputs will be annotated. The learner that receives the feedback (label) may update its prediction rule, and we proceed to the next round. We develop online algorithms for multi-task binary classification that learns in this setting, and bound its performances in the worst-case setting. The algorithms apply an exploration-exploitation approach in order to allocate the limited feedback in the way that reduces the total number of errors. Additionally, we show that our algorithm can be used to solve two bandits problems: contextual bandits, and dueling bandits with context, both allowed to decouple exploration and exploitation. Empirical study with OCR data , vowel prediction (VJ project) and NLP -sentiment analysis data shows that our algorithms outperforms algorithms that use uniform allocation, and essentially makes more (accuracy) for the same labour of the annotator.

Abbreviations and Notations

$SHAMPO$	—	SHared Annotator for Multiple PrOblems
$\|\mathbf{u}\|$	—	ℓ_2-norm of the vector $\mathbf{u}$

Chapter 1

Introduction

Machine learning is a field of computer science that concerned with data processing and the ability of the computer to learn from this data. One main objective of this field is the development of algorithms capable of inference based on observable data, such as text documents, pictures, audio, video etc...

In the *Supervised Learning* setting, the input of the learning algorithm are input-label pairs. The goal of the algorithm is to learn the underlying connection between the inputs and their labels, thus being able to *predict* a label for a previously unseen input. When the possible label values are from a discrete finite set, this learning problem is called *classification*. The basic classification problem is the *binary classification*, i.e. classifying each data instance into one of only two possible classes. In contradiction to the multiclass classification task, in the binary case is more simple because, eliminating one class, gives you the correct one straightforward. However, this is more difficult in the *multiclass classification*, when even when we eliminate one possible class, we yet have some more classes the we need to decide which of these classes is the correct one.

1.1 Online learning

One of the main features of the classification problem is the way how the data is been collected. In some application, the labeled data is been collected first, such that we have an access to entire training dataset at once. Then we use this whole examples collection as an input to a *Batch learning* algorithm and learn a classification model about this problem. However, in a lot of real life application

this is not the case. In some problems like spam filtering, there is a flow of data that is transmitted in sequence and it takes time to collect the a large amount of data to learn from it, so we don't want to wait too long before we can have a decent prediction about the continuing incoming examples. For those application we use *Online Learning* based algorithms. In this setting, at any time we keep the learned model in memory, and update it when a new labeled example is coming in. Unlike the *Batch learning*, in *Online Learning* at any time, we the learner perception about the classification task become stronger when the time is pass by.

The *Online Learning* is performed in rounds, where in each round t, the algorithm gets an input instance $\mathbf{x}_t$ in some domain $\mathcal{X}$ and predicts a correspond measure, $\hat{p}_t$ based on the algorithm decision rule. This measure can be in the label domain, $\mathcal{Y}$, or can be mapped into $\hat{y}_t$ which is the predicted label in $\mathcal{Y}$. After predicting the label, the true label (y_t in the labels domain, $\mathcal{Y}$) is revealed and the learner suffers a non negative loss of $l\left(\hat{p}_t, y_t\right)$ that measures how much the prediction is compatible with the true label. The desired property of such function is to generate low values when the prediction is close to the actual label in some sense, and high values when the opposite is true. Then, the algorithm update its decision rule based on the past known data and the revealed label.

1.2 Selective Sampling

Usually, in an online binary learning task setting, we improve the prediction over the time, which means that the algorithm have less and lees prediction mistakes when it updates its model. Sometimes, annotating the data consume expensive resources, like time, money or manpower, and we would like to avoid using this resources when we can. In other words, we would like to avoid querying labels for the input examples when it is possible. For example, if we update the model only when there is a prediction mistake (as in Perceptron), we actually don't really use the information about the correct label when there is no need to update. In such cases, it will be helpful to assess every time how much we sure about our prediction, and no update should be done, so no query should be issued, or if we not sure about the prediction, hence we should issue a query and update the model using the update rule an the correct label. This approach, that queries labels only for selected examples is called *Selective sampling*.

to do - background in multi-armed bandits. and show the simple perceptron

1.3 Multi Task with Shared Annotator

In supervised learning setting, the main bottleneck is the need to annotate data. A common protocol is problem centric: first collect data or inputs automatically (with low cost), and then pass it on to a user or an expert to be annotated. Annotation can be outsourced to the crowed by a service like Mechanical Turk, or performed by experts as in the Linguistic data Consortium. Then, this data may be used to build models, either for a single task or many tasks. This approach is not making optimal use of the main resource - the annotator - as some tasks are harder than others, yet we need to give the annotator the (amount of) data to be annotated for each task a-priori.

Another aspect of this problem is the need to adapt systems to individual users, to this end, such systems may query the user for the label of some input, yet, if few systems will do so independently, the user will be flooded with queries, and will avoid interaction with those systems. For example, sometimes there is a need to annotate news items from few agencies. One person cannot handle all of them, and only some items can be annotated, which ones? Our setting is designed to handle exactly this problem, and specifically, how to make best usage of annotation time. This settings can also handle with the case when we want to limit the updates number, for example if we have a lot of of clients that generate data, but only one server with a limited computation power is allocated to process the received data and we want to limit the amount of updates for all tasks.

We propose a new framework of online multi-task learning with a shared annotator. Here, algorithms are learning few tasks simultaneously, yet they receive feedback using a central mechanism that trades off the amount of feedback (or labels) each task receives. We derive a specific algorithm based on the good-old Perceptron algorithm, called SHAMPO (SHared Annotator for Multiple PrOblems) for binary classification and analyze it in the mistake bound model, showing that our algorithm may perform well compared with methods that observe all annotated data. We then show how to reduce few contextual bandit problems into our framework, and provide specific bounds for such settings. We evaluate our algorithm with four different datasets for OCR, vowel prediction (VJ) and document classification, and show that it can improve performance either on average over all tasks, or even if their output is combined towards a single shared task, such as multi-class prediction. We conclude with discussion of related work, and few of the many routes to extend this work.

Chapter 2

The Shared Annotator Setting

In our setting, there are K binary classification tasks to be learned simultaneously. In opposed to the common multi-task classification settings, here, no dependency between the tasks is assumed during the analysis, but the tasks can be dependent as well. The model learning is performed in rounds as an online learning algorithm, as following: On each round t, there are K input-label pairs $(\mathbf{x}_{i,t}, y_{i,t})$, one for each classification task, where $i = 1, 2, \ldots, K$ is the task index and t is the step index. The inputs $\mathbf{x}_{i,t} \in \mathbb{R}^{d_i}$ are vectors, and the labels $y_{i,t} \in \{-1, +1\}$ are binary. In the general case, the input-spaces for each problem may be different, and inputs may have different number of elements. Yet, in order to simplify the notation and without loss of generality, from now on, in our analysis we assume that for all of the tasks, $\mathbf{x}_{i,t} \in \mathbb{R}^d$. i.e. all the tasks are in the same dimension and $d_i = d$ holds for all tasks. In practice, since the proposed algorithms use the margin that is affected by the vector norm, there is a need to scale all the vectors into a ball.

On round t, the learning algorithm receives K input vectors $\mathbf{x}_{i,t}$ for $i = 1, \ldots, K$ tasks and produce K binary-labels output $\hat{y}_{i,t}$, where $\hat{y}_{i,t} \in \{-1, +1\}$ is the label predicted for the input $\mathbf{x}_{i,t}$ of task i. The algorithm then chooses a task $J_t \in \{1, \ldots, K\}$ and asks from an annotator its true-label $y_{J_t,t}$ for that task J_t. Unlike the usual online multi-task setting, and due to the limitation in our case, it does not observe any other label. Then, the algorithm updates its models, using the received feedback, and proceeds to the next round (and inputs). For the ease of the calculations below, we denote by K indicators $Z_t = (Z_{1,t}, \ldots, Z_{K,t})$, the identity of the task which was queried by the algorithm on round t, and set $Z_{J_t,t} = 1$ and $Z_{i,t} = 0$ for $i \neq J_t$. Clearly from the definition, the condition, $\sum_i Z_{i,t} = 1, \forall i, t$, always holds. In order to use it in the analysis below, we define

the notation $\mathrm{E}_{t-1}[x]$ as well, to be the conditional expectation $\mathrm{E}[x|Z_1, \ldots Z_{t-1}]$ given all previous choices of the tasks to be queried.

Step	1	2	3	4	5
Task 1	Q	NQ	Q	NQ	NQ
Task 2	NQ	NQ	Q	Q	NQ
Task 3	Q	NQ	Q	NQ	NQ
Task 4	Q	Q	NQ	NQ	Q

Table 2.1: In selective sampling we focus on when to issue a query for a single task (a row).

Step	1	2	3	4	5
Task 1	Q	NQ	NQ	NQ	NQ
Task 2	NQ	NQ	NQ	Q	NQ
Task 3	NQ	NQ	NQ	NQ	Q
Task 4	NQ	Q	NQ	NQ	NQ

Table 2.2: In the SHAMPO setting we focus on when to issue a query for a single step (a column).

Schematic illustration of a single iteration of multi-task algorithms is shown in Fig. 2.1. The top panel shows the standard setting of online multi-task algorithms with a shared annotator, that labels all inputs, which are fed to the corresponding algorithms to update corresponding models. The bottom panel shows the SHAMPO algorithm, which couples labeling annotation and learning process, and synchronizes a single annotation per round. At most one task performs an update per round, the one with the annotated input.

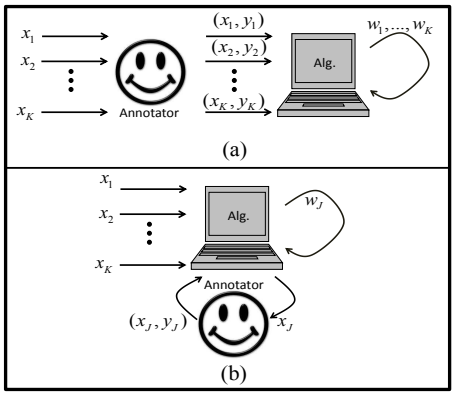

Figure 2.1: Illustration of a single iteration of multi-task algorithms (a) standard setting, shared annotator labels all inputs, and algorithms update models. (b) SHAMPO algorithm couples labeling annotation and learning process, and synchronizes a single annotation per round.

Chapter 3

SHMPO (SHared Annotator for Multiple Problems) Algorithms

We now describe algorithms for multi-task learning with a shared annotator setting, that uses linear models. Two steps are yet to be specified: how to pick a task to be labeled and how to perform an update, once the true label for that task is given. Our first algorithm is based on the perceptron algorithm

3.1 SHAMPO perceptron

We focus now on linear-functions of the form $f(\mathbf{x}) = \text{sign}(p)$ for the quantity $p = \mathbf{w}^\top \mathbf{x}$, $\mathbf{w} \in \mathbb{R}^d$, which often called the *margin*. Specifically, the algorithm maintains a set of K weight vectors, $\mathbf{w}_{i,t},\ i \in \{1, \ldots, K\}$. On round t, the algorithm predicts a label for each one of the tasks, $\hat{y}_{i,t} = \text{sign}(\hat{p}_{i,t})$ where $\hat{p}_{i,t} = \mathbf{w}_{i,t-1}^\top \mathbf{x}_t$. On rounds for which the label of some task J_t is queried, the algorithm update the model of the queried task only, such that for the rest of the tasks, $i \neq J_t$, we have $\mathbf{w}_{i,t} = \mathbf{w}_{i,t-1}$ and no update is made for those unqueried tasks.

We say that the algorithm has a prediction mistake on task i in round t if $y_{i,t} \neq \hat{y}_{i,t}$, and denote this event by the indicator $M_{i,t} = 1$, otherwise, if there is no mistake, we set $M_{i,t} = 0$. The goal of the algorithm is to minimize the cumulative mistakes, $\sum_t \sum_i M_{i,t}$. Models are also evaluated using the *Hinge-loss*. Specifically, let $\mathbf{u}_i \in \mathbb{R}^d$ be some vector associated with problem i. We denote the Hinge-loss $\boxed{\text{show hinge loss plot}}$ of it, with respect to some input-label pair

$(\mathbf{x}_{i,t}, y_{i,t})$, by $\ell_{\gamma,i,t}(\mathbf{u}_i) = \left(\gamma - y_{i,t}\mathbf{u}_i^\top \mathbf{x}_{i,t}\right)_+$, where, $(x)_+ = \max\{x, 0\}$, and $\gamma > 0$ is some parameter. The cumulative loss over all tasks and a sequence of n input steps, is $L_{\gamma,n} = L_\gamma(\{\mathbf{u}_i\}) = \sum_{t=1}^{n}\sum_{i=1}^{K} \ell_{\gamma,i,t}(\mathbf{u}_i)$. We also use the following expected hinge-loss over the random queried choices of the algorithm,

$$\bar{L}_{\gamma,n} = \bar{L}_{\{\mathbf{u}_i\}} = \mathrm{E}\left[\sum_{t}^{n}\sum_{i=1}^{K} M_{i,t} Z_{i,t} \ell_{\gamma,i,t}(\mathbf{u}_i)\right]. \tag{3.1}$$

Now, we proceed by describing our algorithm and specify how to choose a task to query its label, and how to perform an update.

In order to select a task to query on, the algorithm uses the absolute margin $\hat{p}_{i,t}$. We can see $|\hat{p}_{i,t}|$ as the certainty measure of the label prediction. intuitively if $|\hat{p}_{i,t}|$ is small, then there is uncertainty about the labeling of $\mathbf{x}_{i,t}$, and vice-versa for large values of $|\hat{p}_{i,t}|$. Similar argument was used by Tong and Koller [2000] for picking an example to be labeled in batch active learning. Prima facie, under this claim, at each point we should query the true label for the task that corresponds to the smallest margin. Yet, if the model $\mathbf{w}_{i,t-1}$ is not accurate enough, due to small number of observed examples, this estimation may be rough, and may lead to a wrong conclusion. We thus perform an exploration-exploitation strategy, and query tasks randomly, with a bias towards tasks with low margin $|\hat{p}_{i,t}|$.

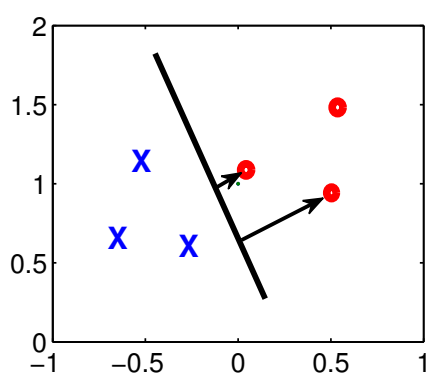

Figure 3.1: Example for the using of the margin as a certainty measure. The circles and crosses represent two different classes and the separate hyperplane is shown. One of the circles examples have lower margin than the other, so its label can easily been replaced with the other one.

To the best of our knowledge, exploration-exploitation usage in this context of choosing an examples to be labeled (e.g. in settings such as semi-supervised learning or selective sampling) is novel and new. In order to get this property, we induced a distribution over the tasks in the time step t, such that the probability to issue a query on the task j ()$j = 1, \ldots, K$) is:

$$\begin{aligned} \Pr\left[J_t = j\right] &= \frac{\left(b + |\hat{p}_{j,t}| - \min_{m=1}^{K} |\hat{p}_{m,t}|\right)^{-1}}{D_t} \\ \text{for } \; D_t &= \sum_{i=1}^{K} \left(b + |\hat{p}_{i,t}| - \min_{m} |\hat{p}_{m,t}|\right)^{-1}. \end{aligned} \tag{3.2}$$

where $b \geq 0, b \in \mathbb{R}$ is a tradeoff parameter, between exploration and exploitation. Clearly, it is a distribution, since $\Pr\left[J_t = j\right] \geq 0$ and $\sum_j \Pr\left[J_t = j\right] = 1$. When we examine the extreme cases of b we see that for $b \to 0$ we have $\Pr\left[J_t = j\right] \to 1$ and for the task with minimal margin, $J_t = \arg\min_{i=1}^{K} |\hat{p}_{i,t}|$, and $\Pr\left[J_t = j\right] \to 0$ for all the rest. In this case , pure exploitation is being made. The pure exploration happens when $b \to \infty$, then the distribution is uniform, $\Pr\left[J_t = j\right] = 1/K, \;\; \forall j$. Fig. **??** shows an example of this distribution for the case of two tasks ($K = 2$) at an arbitrary step t^*. For visualization purpose, we fix $|\hat{p}_{2,t^*}| = 0.5$ and see how the probability of task 1 to be chosen, is affected by varying b and $|\hat{p}_{1,t^*}|$ values. Three different vertical areas can be easily seen in the graph. The upper (green) area, where $b >> \max\left\{\hat{p}_{1,t^*}, \hat{p}_{2,t^*}\right\}$, shows uniform distribution ($\Pr\left[J_{t^*} = 2\right] = \Pr\left[J_{t^*} = 1\right] = 1/K = 0.5$) which represents the exploration over the tasks. In the lower area, the probability is compatible with the exploitation method and is changing between probability 1 in the left, and probability 0 to the right with a sharp threshold at $|\hat{p}_{1,t^*}| = 0.5$, which is very close to the delta distribution $\Pr\left[J_{t^*} = 1\right] = \mathbb{I}\left[\hat{p}_{1,t^*} < \hat{p}_{2,t^*}\right]$. Whereas, the intermediate area, is the exploration-exploitation area that is given by a distribution that is biased toward the lowest margin task.

As noted above we denote by $Z_{i,t} = 1$ iff $i = J_t$. The update of the algorithm is performed with the perceptron rule, that is $\mathbf{w}_{J_t,t} = \mathbf{w}_{J_t,t-1} + M_{J_t,t}\, y_{J_t,t}\, \mathbf{x}_{J_t,t}$ and $\mathbf{w}_{i,t} = \mathbf{w}_{i,t-1}$ for $i \neq J_t$. For simplicity of presentation we write the update for all of the tasks in one term as, $\mathbf{w}_{i,t} = \mathbf{w}_{i,t-1} + Z_{i,t}\, M_{i,t}\, y_{i,t}\, \mathbf{x}_{i,t}$. One can notice that although this notation uses labels for all tasks, only the label of the task J_t is used in practice, as for other tasks $Z_{i,t} = 0$.

We call this algorithm *SHAMPO* for SHared Annotator for Multiple PrOblems.

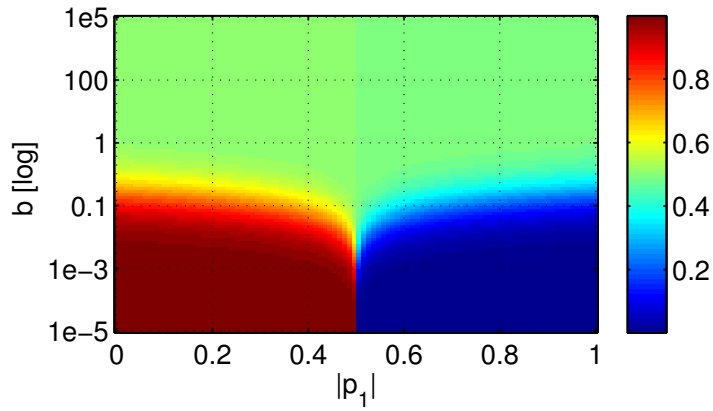

Figure 3.2: Example of the distribution over 2 tasks.

The pseudo-code of this algorithm appears in Fig. 3.4.

We conclude this section by noting that the algorithm can be incorporated with Mercer-kernels as all operations depend implicitly on inner-product between inputs.

The following theorem states that the expected cumulative number of mistakes that the algorithm makes, may not be higher than for an algorithm that observes the labels of all inputs.

Theorem 1 *If SHAMPO algorithm runs on K tasks with K parallel example pair sequences $(\mathbf{x}_{i,1}, y_{i,1}), ...(\mathbf{x}_{i,n}, y_{i,n}) \in \mathbb{R}^d \times \{-1, 1\}$, $i = 1, ..., K$ with input parameter $b > 0$, then for all $\gamma > 0$, all $\mathbf{u}_i \in \mathbb{R}^d$ and all $n \geq 1$, there exists $0 < \delta \leq K$, such that,*

$$\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right] \leq \frac{\delta}{\gamma}\left[\left(1+\frac{X^2}{2b}\right)\bar{L}_{\gamma,n} + \frac{\left(2b+X^2\right)^2 \tilde{U}^2}{8\gamma b}\right] ,$$

where $X = \max_{i,t} \|\mathbf{x}_{i,t}\|$, $\tilde{U}^2 = \sum_{i=1}^{K} \|\mathbf{u}_i\|^2$ and the expectation is over the random choices of the algorithm.

Proof: Fix n examples sequences, $(\mathbf{x}_{i,1}, y_{i,1}), ..., (\mathbf{x}_{i,n}, y_{i,n})$ for each one of the K tasks. Let t be certain trial and i to be an update task on this trial, such that

Parameters: $b \in \mathbb{R} > 0$.
Initialize: $\mathbf{w}_{i,0} = \mathbf{0}$ for $i = 1, \ldots, K$
For $t = 1, 2, ..., n$

1. Observe K instance vectors, $\mathbf{x}_{i,t}$, $(i = 1, \ldots, K)$.
2. Compute margins $\hat{p}_{i,t} = \mathbf{w}_{i,t-1}^{\top} \mathbf{x}_{i,t}$.
3. Predict K labels, $\hat{y}_{i,t} = \mathrm{sign}(\hat{p}_{i,t})$.
4. Draw task J_t with the distribution:

$$\Pr\left[J_t = j\right] = \frac{\left(b + |\hat{p}_{j,t}| - \min_{m=1}^{K} |\hat{p}_{m,t}|\right)^{-1}}{D_t}, Eq.\ (\mathbf{??})$$
$$D_t = \sum_i \left(b + |\hat{p}_{i,t}| - \min_{m=1}^{K} |\hat{p}_{m,t}|\right)^{-1}.$$

5. Query the true label ,$y_{J_t,t} \in \{-1, 1\}$.
6. Set the indicator $M_{J_t,t} = 1$ iff $y_{J_t,t} \neq \hat{y}_{J_t,t}$.
7. Update with the perceptron rule:

$$\begin{aligned} \mathbf{w}_{J_t,t} &= \mathbf{w}_{J_t,t-1} + M_{J_t,t}\, y_{J_t,t}\, \mathbf{x}_{J_t,t} \\ \mathbf{w}_{i,t} &= \mathbf{w}_{i,t-1} \text{ for } i \neq J_t \end{aligned} \tag{3.3}$$

End for

Output : $\mathbf{w}_{i,n}$ for $i = 1, \ldots, K$.

Figure 3.3: SHAMPO: SHared Annotator for Multiple PrOblems.

$M_{i,t} = 1$. We can write,

$$\begin{aligned} \gamma - \ell_{\gamma,i,t}(\mathbf{u}_i) &= \gamma - \left(\gamma - y_{i,t}\mathbf{u}_i^T \mathbf{x}_{i,t}\right)_+ \\ &\leq\ y_{i,t}\mathbf{u}_i^T \mathbf{x}_{i,t} \\ &= y_{i,t}\left(\mathbf{u}_i + \mathbf{w}_{i,t-1} - \mathbf{w}_{i,t-1}\right)^T \mathbf{x}_{i,t} \\ &= y_{i,t}\mathbf{w}_{i,t-1}^T \mathbf{x}_{i,t} + \left(\mathbf{u}_i - \mathbf{w}_{i,t-1}\right)^T y_{i,t}\mathbf{x}_{i,t} \\ &= y_{i,t}\mathbf{w}_{i,t-1}^T \mathbf{x}_{i,t} + \left(\mathbf{u}_i - \mathbf{w}_{i,t-1}\right)^T \left(\mathbf{w}_{i,t} - \mathbf{w}_{i,t-1}\right) \\ &= y_{i,t}\mathbf{w}_{i,t-1}^T \mathbf{x}_{i,t} + \frac{1}{2}\|\mathbf{u}_i - \mathbf{w}_{i,t-1}\|^2 - \frac{1}{2}\|\mathbf{u}_i - \mathbf{w}_{i,t}\|^2 + \frac{1}{2}\|\mathbf{w}_{i,t-1} - \mathbf{w}_{i,t}\|^2 \end{aligned}$$

$$= y_{i,t}\hat{p}_{i,t} + \frac{1}{2}\|\mathbf{u}_i - \mathbf{w}_{i,t-1}\|^2 - \frac{1}{2}\|\mathbf{u}_i - \mathbf{w}_{i,t}\|^2 + \frac{1}{2}\|\mathbf{w}_{i,t-1} - \mathbf{w}_{i,t}\|^2 .$$

The last inequality holds for all $\gamma > 0$ and for all $\mathbf{u}_i \in \mathbb{R}^d$, so we can replace γ and $\mathbf{u}_i$ by their scaling $\alpha\gamma$ and $\alpha\mathbf{u}_i$ respectively, where $\alpha > 0$ is a scaling parameter that will be determined shortly. Since the inequality is computed on an update task, $y_{i,t}\hat{p}_{i,t} \leq 0$ holds , i.e. $y_{i,t}\hat{p}_{i,t} = -\left|\hat{p}_{i,t}\right|$, and we get

$$\alpha\gamma + |\hat{p}_{i,t}| \leq \alpha\ell_{\gamma,i,t}(\mathbf{u}_i) + \frac{1}{2}\|\alpha\mathbf{u}_i - \mathbf{w}_{i,t-1}\|^2 - \frac{1}{2}\|\alpha\mathbf{u}_i - \mathbf{w}_{i,t}\|^2 + \frac{1}{2}\|\mathbf{w}_{i,t-1} - \mathbf{w}_{i,t}\|^2.$$

Now, in order to generalize the inequality, we multiply it by the indicator $M_{i,t}Z_{i,t}$, which means that the inequality holds for trials and tasks where there is an update, as stated above.

$$M_{i,t}Z_{i,t}(\alpha\gamma - |\hat{p}_{i,t}|) \leq M_{i,t}Z_{i,t}\alpha\ell_{\gamma,i,t}(\mathbf{u}_i) + \frac{M_{i,t}Z_{i,t}}{2}\|\alpha\mathbf{u}_i - \mathbf{w}_{i,t-1}\|^2 - \frac{M_{i,t}Z_{i,t}}{2}\|\alpha\mathbf{u}_i - \mathbf{w}_{i,t}\|^2 + \frac{M_{i,t}Z_{i,t}}{2}\|\mathbf{w}_{i,t-1} - \mathbf{w}_{i,t}\|^2.$$

Yet, in the other cases, when no update is performed and $M_{i,t}Z_{i,t} = 0$, the equality $\mathbf{w}_{i,t} = \mathbf{w}_{i,t-1}$ holds as well, so we can rid off the indicator for any one of the last three terms and we have

$$M_{i,t}Z_{i,t}(\alpha\gamma + |\hat{p}_{i,t}|) \leq M_{i,t}Z_{i,t}\alpha\ell_{\gamma,i,t}(\mathbf{u}_i) + \frac{1}{2}\|\alpha\mathbf{u}_i - \mathbf{w}_{i,t-1}\|^2 - \frac{1}{2}\|\alpha\mathbf{u}_i - \mathbf{w}_{i,t}\|^2 + \frac{M_{i,t}Z_{i,t}}{2}\|\mathbf{w}_{i,t-1} - \mathbf{w}_{i,t}\|^2 . \tag{3.4}$$

Next, we sum the inequality above, over t, recall the facts that $\mathbf{w}_{i,0} = 0$ and $\|\mathbf{w}_{i,t-1} - \mathbf{w}_{i,t}\|^2 \leq X^2$ for $X = \max_{i,t}\|\mathbf{x}_{i,t}\|$ to get,

$$\sum_{t=1}^{n} M_{i,t}Z_{i,t}\left(\alpha\gamma + |\hat{p}_{i,t}| - \frac{X^2}{2}\right) \leq \alpha\sum_{t=1}^{n} M_{i,t}Z_{i,t}\ell_{\gamma,i,t}(\mathbf{u}_i) + \frac{\alpha^2}{2}\|\mathbf{u}_i\|^2 . \tag{3.5}$$

Substituting $\alpha = (2b + X^2)/2\gamma$ (where $b \in \mathbb{R}, \;\; b > 0$) in Eq. (3.6), we get

$$\sum_{t=1}^{n} M_{i,t} Z_{i,t}(b + |\hat{p}_{i,t}|) \leq \frac{2b + X^2}{2\gamma} \sum_{t=1}^{n} M_{i,t} Z_{i,t} \ell_{\gamma,i,t}(\mathbf{u}_i) + \frac{\left(2b + X^2\right)^2}{8\gamma^2} \|\mathbf{u}_i\|^2 .$$

Now, we subtract a non negative quantity $\sum_{t=1}^{n} M_{i,t} Z_{i,t} \min_j |\hat{p}_{j,t}|$ from the l.h.s. and get,

$$\begin{aligned} &\sum_{t=1}^{n} M_{i,t} Z_{i,t} \left(b + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}| \right) \leq \\ &\qquad \frac{2b + X^2}{2\gamma} \sum_{t=1}^{n} M_{i,t} Z_{i,t} \ell_{\gamma,i,t}(\mathbf{u}_i) + \frac{\left(2b + X^2\right)^2}{8\gamma^2} \|\mathbf{u}_i\|^2 . \end{aligned} \tag{3.6}$$

At this point we take the expectation of all the terms. Recall that the conditional expectation of $Z_{i,t}$ is $(b + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|)^{-1}/D_t$ and that $M_{i,t}$ and $\hat{p}_{i,t}$ are measurable with respect to the σ-algebra that generated by $Z_1, ...Z_{t-1}$. We start with the left term,

$$\begin{aligned} &\mathbb{E}\left[\sum_{t=1}^{n} M_{i,t} Z_{i,t} \left(b + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}| \right) \right] = \\ &\mathbb{E}\left[\mathbb{E}_{t-1}\left[\sum_{t=1}^{n} M_{i,t} Z_{i,t} \left(b + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}| \right) \right] \right] = \\ &\mathbb{E}\left[\sum_{t=1}^{n} M_{i,t} \left(b + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}| \right) \mathbb{E}_{t-1}\left[Z_{i,t} \right] \right] = \\ &\mathbb{E}\left[\sum_{t=1}^{n} \frac{M_{i,t}}{D_t} \right] . \end{aligned}$$

Substituting the last term in the expectation of Eq. (3.6),

$$\mathbb{E}\left[\sum_{t=1}^{n} \frac{M_{i,t}}{D_t} \right] \leq \frac{2b + X^2}{2\gamma} \bar{L}_{\gamma,i,n}(u_i) + \frac{\left(2b + X^2\right)^2}{8\gamma^2} \|\mathbf{u}_i\|^2 . \tag{3.7}$$

Since that the normalization factor is a sum of positive components, hence D_t is maximal when each of the components gets its maximal value. This happens when $|\hat{p}_{m,t}| = |\hat{p}_{j,t}| \quad \forall m, j \in \{1, .., K\}$. This allows us to bound the normalization

factor as follows,

$$D_t = \sum_{m=1}^{K} \left(b + |\hat{p}_{m,t}| - \min_j |\hat{p}_{j,t}| \right)^{-1} \leq \sum_{m=1}^{K} b^{-1} = \frac{K}{b} \,. \tag{3.8}$$

Since $M_{i,t} > 0 \; \forall i, t$, thus there exists $\delta_i \in \mathbb{R} > 0$ such that

$$\mathbb{E}\left[\sum_{t=1}^{n} \frac{M_{i,t}}{D_t} \right] = \frac{b}{\delta_i} \mathbb{E}\left[\sum_{t=1}^{n} M_{i,t} \right] . \tag{3.9}$$

and

$$\frac{b}{\delta_i} \geq \min \frac{1}{D_t} = \frac{1}{\max D_t} \geq \frac{b}{K} \,.$$

Which implies that $0 < \delta_i \leq K$.
Plugging Eq. (3.9) in Eq. (3.7) we get,

$$\frac{b}{\delta_i} \mathbb{E}\left[\sum_{t=1}^{n} M_{i,t} \right] \leq \frac{2b + X^2}{2\gamma} \bar{L}_{\gamma,i,n}(\mathbf{u}_i) + \frac{\left(2b + X^2\right)^2}{8\gamma^2} \|\mathbf{u}_i\|^2. \tag{3.10}$$

Summing up the last inequality over all K tasks and setting $\delta = \max_i \delta_i$ yields, $\boxed{\text{try to replace with } \min_i \delta_i \leq \delta \leq \max_i \delta_i}$

$$\frac{1}{\delta} \mathbb{E}\left[\sum_{i=1}^{K} \sum_{t=1}^{n} M_{i,t} \right] \leq \frac{1}{\gamma} \left(1 + \frac{X^2}{2b} \right) \bar{L}_{\gamma,n} + \frac{\left(2b + X^2\right)^2}{8\gamma^2 b} \sum_{i=1}^{K} \|\mathbf{u}_i\|^2 \,, \tag{3.11}$$

which concludes the proof. ■

One can use the bound to tune the algorithm for an optimal value of b. However, this may not be possible as unfortunately $\bar{L}_{\gamma,n}$ depends implicitly on the value of b[1]. Alternatively, we can take a loose estimate of $\bar{L}_{\gamma,n}$, and replace it with $L_{\gamma,n}$ (which is $\sim K$ times larger). The optimal value of b can be now calculated,

$$b = \frac{X^2}{2} \sqrt{1 + \frac{4\gamma L_{\gamma,n}}{U^2 X^2}} .$$

Substituting this value in the bound of Eq. (**??**) with $L_{\gamma,n}$ leads to the following

[1] Similar issue appears also after the discussion of Theorem 1 in a different context Cesa-Bianchi et al. [2006].

bound,

$$\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right] \leq \frac{\delta}{\gamma}\left[L_{\gamma,n} + \frac{U^2X^2}{2\gamma} + \frac{U^2}{2\gamma}\sqrt{1+\frac{4\gamma L_{\gamma,n}}{U^2X^2}}\right]$$

which has the same dependency in the number of inputs n as algorithm that observes all of them. $\boxed{\text{show this statement}}$

We continue now with an analysis of an extension of our algorithm that allows more than a single query per round. In this setting, the ability of the annotator to annotate data instances is less limited. Here, we allow the algorithm to query κ labels at a time ($\kappa < K$), instead of one. On each iteration t, the modified algorithm samples without repetitions κ labels to be annotated, and perform the same update as of Eq. (**??**). Formally, on each round we have $\sum_i Z_{i,t} = \kappa$ for $Z_{i,t} \in \{0,1\}$ where the first task-index to be queried is drawn according to Eq. (3.2). The second task is drawn from the same distribution, eliminating the first choice and adjusting the normalization factor, and so on. Once κ problems are drawn, the algorithm receives κ labels for the κ corresponding inputs, and updates the κ models according to Eq. (**??**).

Corollary 2 *If SHAMPO algorithm gets feedback for κ tasks on each round, instead of only a single problem, the expected cumulative weighted mistakes is bounded as follows*

$$\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right] \leq \frac{1}{C(\kappa,K)}\left(\frac{1}{\gamma}\left(1+\frac{X^2}{2b}\right)\bar{L}^{\kappa}_{\gamma,n} + \kappa\frac{\left(2b+X^2\right)^2}{8\gamma^2 b}\tilde{U}^2\right) ,$$

where $C(\kappa,K) = \sum_{j=K-\kappa+1}^{K}\frac{1}{j}$, and $\bar{L}^{\kappa}_{\gamma,n}$ is the expected loss of K models $\{\mathbf{u}_i\}$ over the κ instances that are annotated per round t.

$\boxed{\text{show full proof}}$

Proof: We follow the proof of Theorem **??** until Eq. (3.11). We repeat the process κ times, and get the equivalent inequality for sampling κ problems without repetitions,

$$\left(\sum_{j=K-\kappa+1}^{K}\frac{1}{j}\right)\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right] \leq \frac{1}{\gamma}\left(1+\frac{X^2}{2b}\right)\bar{L}^{\kappa}_{\gamma,n} + \kappa\frac{\left(2b+X^2\right)^2}{8\gamma^2 b}U^2 , \tag{3.12}$$

where all expectations are now with respect to the sampling repetitions, and specifically $\bar{L}^{\kappa}_{\gamma,n}$ is the expected loss of a set of linear models $\{\mathbf{u}_i\}$ where κ problems are sampled rather than a single one. ■

It can be shown that this bound is a generalized form of Theorem **??**. For $\kappa = 1$ we get the bound of Theorem **??**, while for $\kappa = K$ we have $C(\kappa, K) \approx \log(K)$, thus recovering the bound of K Perceptron trained in parallel, with a sightly $(logK)$ worse dependency in the number of tasks.

3.2 Aggressive version

So far we used the margin to build the distribution over the tasks, but in the update stage, we used the same regular perceptron update rule, that makes an update only when there is a prediction mistake. However, we already said that the margin can tells us about the uncertainty in the prediction, so it may be helpful to proceed with this approach in the update stage as well and make the sharp update threshold a little bit softer.

In this version of SHAMPO algorithm, the update is been preformed not only when there is a mistake, but also in the case of correct prediction with low margin, i.e. low certainty. We define this margin to be $\lambda \in \mathbb{R} > 0$. In addition to the previous mistake indicator, $M_{i,t}$, we introduce one more indicator, $A_{i,t}$. We keep the notation of $M_{i,t} = 1$ when there is wrong prediction on the task i in round t and $M_{i,t} = 0$ otherwise, and set $A_{i,t} = 1$ when there is no prediction mistake on the task i in round t, but the margin is lower than our threshold, i.e. $0 < \hat{p}_{i,t} < \lambda$, therefore an aggressive update takes place, and $A_{i,t} = 0$ otherwise. An update is performed if either there is a mistake ($M_{J_i,t} = 1$) or the margin is low ($A_{J_i,t} = 1$). Note that these events are mutually exclusive thus, for simplicity, we define the update indicator which is $U_{i,t} = M_{i,t} + A_{i,t}$. This indicator is $U_{i,t} = 1$ on update trials and $U_{i,t} = 1$ when no update has been made.

Theorem 3 *If Aggressive SHAMPO algorithm runs on K problems with K parallel example pair sequences $(\mathbf{x}_{i,1}, y_{i,1}), ...(\mathbf{x}_{i,n}, y_{i,n}) \in \mathbb{R}^d \times \{-1, 1\}$, $i = 1, ..., K$ with input parameters $b > 0$, and $0 \leq \lambda \leq b/2$ then for all $\gamma > 0$, all $\mathbf{u}_i \in \mathbb{R}^d$ and all $n \geq 1$, there exists $0 < \delta \leq K$, such that,*

Parameters: $b, \lambda \in \mathbb{R} > 0$.
Initialize: $\mathbf{w}_{i,0} = \mathbf{0}$ for $i = 1, \ldots, K$
For $t = 1, 2, ..., n$

1. Observe K instance vectors, $\mathbf{x}_{i,t}$, $(i = 1, \ldots, K)$.
2. Compute margins $\hat{p}_{i,t} = \mathbf{w}_{i,t-1}^{\top} \mathbf{x}_{i,t}$.
3. Predict K labels, $\hat{y}_{i,t} = \mathrm{sign}(\hat{p}_{i,t})$.
4. Draw problem J_t with the distribution:

$$\Pr\left[J_t = j\right] = \frac{\left(b + |\hat{p}_{j,t}| - \min_{m=1}^{K} |\hat{p}_{m,t}|\right)^{-1}}{D_t}, Eq. (\mathbf{??})$$
$$D_t = \sum_i \left(b + |\hat{p}_{i,t}| - \min_{m=1}^{K} |\hat{p}_{m,t}|\right)^{-1}.$$

5. Query the true label ,$y_{J_t,t} \in \{-1, 1\}$.
6. Set the indicator $M_{J_t,t} = 1$ iff $y_{J_t,t} \neq \hat{y}_{J_t,t}$.
7. Iff $M_{J_t,t} = 0$ and $|p_{J_t,t}| < \lambda$, set the indicator $A_{J_t,t} = 1$
8. For $U_{i,t} = M_{i,t} + A_{i,t}$ update with the perceptron rule:

$$\begin{aligned} \mathbf{w}_{J_t,t} &= \mathbf{w}_{J_t,t-1} + U_{J_t,t}\, y_{J_t,t}\, \mathbf{x}_{J_t,t} \\ \mathbf{w}_{i,t} &= \mathbf{w}_{i,t-1} \text{ for } i \neq J_t \end{aligned} \tag{3.13}$$

End for

Output : $\mathbf{w}_{i,n}$ for $i = 1, \ldots, K$.

Figure 3.4: SHAMPO aggressive perceptron.

$$\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right] \leq \frac{\delta}{\gamma}\left[\left(1 + \frac{X^2}{2b}\right)\bar{L}_{\gamma,n} + \frac{\left(2b + X^2\right)^2 \tilde{U}^2}{8\gamma b} + \left(2\frac{\lambda}{b} - 1\right)\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} A_{i,t}\right]\right],$$

where $X = \max_{i,t} \|\mathbf{x}_{i,t}\|$, $\tilde{U}^2 = \sum_{i=1}^{K} \|\mathbf{u}_i\|^2$ *and the expectation is over the random choices of the algorithm.*

The theorem above, states that when the update is aggressive, the bound on the expected number of errors can be tighter than the non aggressive algorithm because we have added a new term term to the bound which can be negative.

In order to achieve tighter bound on the expected cumulative mistakes than we had in the regular SHAMPO perceptron algorithm (Thm. 1), we need to restrict the aggressive margin λ to be $\lambda \leq b/2$. Indeed, one can say that this bound shows that the best choice of λ is $\lambda = 0$, i.e. it seems that aggressive update is not a good idea. However, in this case, $A_{i,t} = 0,\ \forall i, t$ which means that this would not be the best choice. This discrimination actually shows that there is a sweet point where $0 < \lambda < b/2$ such that the bound will be the tightest as this bound allows .

Proof: We follow the proof of $Thm.$ 3.1 on an update trial, reminding that now the update indicator is $U_{i,t} = 1$, until we get,

$$\gamma - \ell_{\gamma,i,t}(\mathbf{u}_i) \leq \\ y_{i,t}\hat{p}_{i,t} + \frac{1}{2}\|\mathbf{u}_i - \mathbf{w}_{i,t-1}\|^2 - \frac{1}{2}\|\mathbf{u}_i - \mathbf{w}_{i,t}\|^2 + \frac{1}{2}\|\mathbf{w}_{i,t-1} - \mathbf{w}_{i,t}\|^2 .$$

Whereas in the non aggressive version proof we claimed that $y_{i,t}\hat{p}_{i,t} = -|\hat{p}_{i,t}|$ in an update trial, here, it is not true when an aggressive update takes place since we update when the prediction is correct as well. For this reason, we remain with the term $y_{i,t}\hat{p}_{i,t}$ as is for now and we will deal with it later. In addition, in this proof, we change the update indicator from $M_{i,t}$ to $U_{i,t}$. Considering these two changes, we proceed the proof until $Eq.$ (3.6). Now we have,

$$\sum_{t=1}^{n} U_{i,t} Z_{i,t}\left(b - y_{i,t}\hat{p}_{i,t} - \min_j |\hat{p}_{j,t}|\right) \leq \\ \frac{2b + X^2}{2\gamma}\sum_{t=1}^{n} U_{i,t} Z_{i,t}\ell_{\gamma,i,t}(\mathbf{u}_i) + \frac{\left(2b + X^2\right)^2}{8\gamma^2}\|\mathbf{u}_i\|^2 . \tag{3.14}$$

Since the inequality is computed on an update task, there is a need to discriminate between the two possible options: either $y_{i,t}\hat{p}_{i,t} \leq 0$ holds when there is prediction mistake ($M_{i,t} = 1$), i.e. $y_{i,t}\hat{p}_{i,t} = -|\hat{p}_{i,t}|$, or $0 \leq y_{i,t}\hat{p}_{i,t} \leq \lambda$ holds when the prediction is correct, but an aggressive update occurs ($A_{i,t} = 1$) and in that case $y_{i,t}\hat{p}_{i,t} = |\hat{p}_{i,t}|$.

At this point we take the expectation of all the terms as before taking into consideration the two cases. Recall that the conditional expectation of $Z_{i,t}$ is $(b + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|)^{-1}/D_t$ and that $M_{i,t}$ and $A_{i,t}$ (therefor also $U_{i,t}$) and $\hat{p}_{i,t}$ are measurable with respect to the σ-algebra that generated by $Z_1, ...Z_{t-1}$. We start with the left term,

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{n} U_{i,t} Z_{i,t}\left(b - y_{i,t}\hat{p}_{i,t} - \min_j |\hat{p}_{j,t}|\right)\right] \\
&= \mathbb{E}\left[\mathbb{E}_{t-1}\left[\sum_{t=1}^{n} U_{i,t} Z_{i,t}\left(b - y_{i,t}\hat{p}_{i,t} - \min_j |\hat{p}_{j,t}|\right)\right]\right] \\
&= \mathbb{E}\left[\sum_{t=1}^{n} U_{i,t}\left(b - y_{i,t}\hat{p}_{i,t} - \min_j |\hat{p}_{j,t}|\right)\mathbb{E}_{t-1}\left[Z_{i,t}\right]\right] \\
&= \mathbb{E}\left[\sum_{t=1}^{n} \frac{(M_{i,t} + A_{i,t})}{D_t}\frac{(b - y_{i,t}\hat{p}_{i,t} - \min_j |\hat{p}_{j,t}|)}{b + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|}\right] \\
&= \mathbb{E}\left[\sum_{t=1}^{n} \frac{1}{D_t}\left(M_{i,t} + \frac{b - |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|}{b + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|} A_{i,t}\right)\right] .
\end{aligned}$$

Substituting the last term in the expectation of Eq. (3.14) we get,

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{n} \frac{1}{D_t}\left(M_{i,t} + \frac{b - |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|}{b + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|} A_{i,t}\right)\right] \leq \\
\frac{2b + X^2}{2\gamma}\bar{L}_{\gamma,i,n}(u_i) + \frac{\left(2b + X^2\right)^2}{8\gamma^2}\|\mathbf{u}_i\|^2 .
\end{aligned} \tag{3.15}$$

Now we bound the factor that is multiplied by $A_{i,t}$ using the fact that $A_{i,t} = 1$ only when $|\hat{p}_{i,t}| < \lambda$,

$$\left(1 - 2\frac{\lambda}{b}\right) = \frac{b - 2\lambda}{b} \leq \frac{b - |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|}{b + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|} \leq 1. \tag{3.16}$$

As stated before, there exists $\delta_i \in \mathbb{R},\ 0 < \delta_i \leq K$ such that Eq. (3.9) holds.

Plugging this in Eq. (3.15) we get,

$$\begin{aligned}\frac{b}{\delta_i}\mathbb{E}\left[\sum_{t=1}^{n} M_{i,t}\right] \leq & \frac{2b+X^2}{2\gamma}\bar{L}_{\gamma,i,n}(\mathbf{u}_i) \\ & + \frac{\left(2b+X^2\right)^2}{8\gamma^2}\|\mathbf{u}_i\|^2 + \frac{b}{\delta_i}\left(2\frac{\lambda}{b}-1\right)\mathbb{E}\left[\sum_{t=1}^{n} A_{i,t}\right].\end{aligned} \tag{3.17}$$

Summing up the last inequality over all K tasks and setting $\delta = \max_i \delta_i$ yields,

$$\begin{aligned}\frac{1}{\delta}\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right] \leq & \frac{1}{\gamma}\left(1+\frac{X^2}{2b}\right)\bar{L}_{\gamma,n} \\ & + \frac{\left(2b+X^2\right)^2}{8\gamma^2 b}\sum_{i=1}^{K}\|\mathbf{u}_i\|^2 + \frac{1}{\delta}\left(2\frac{\lambda}{b}-1\right)\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} A_{i,t}\right]\end{aligned}$$

Which concludes the proof ■

3.3 prior distribution over tasks

We saw before that b is a tradeoff parameter between exploitation and uniform distribution over the tasks (exploration). It is possible that we have prior knowledge about the tasks, which tasks are harder than the others. In this case, we would like to add this prior knowledge to the distribution such that in the exploration will be done with a prior distribution. This is especially important in the first steps of the algorithms when instead starting in pure uniform exploration (since we initialize our model vectors, $\mathbf{w}_{i,0} = \mathbf{0}$) we can start with a reasonable prior distribution.

Try to simulate the algorithm with varying in time a_i prion that uses the distance between the centers of querie

In order to do that we change the probability to be

$$\Pr\left[J_t = j\right] = \frac{1}{D_t}\frac{a_j}{\left(b + |\hat{p}_{j,t}| - \min_{m=1}^{K}|\hat{p}_{m,t}|\right)}$$

where $a_j > 1 \;\; \forall j \in \{1, \cdots, K\}$ and D_t is the normalization factor. We can use this distribution with the SHAMPO perceptron algorithm, but we would like to analyse the bound of the SHAMPO aggressive perceptron algorithm because it's more global.

Theorem 4 *If Aggressive SHAMPO algorithm with prior runs on K problems with K parallel example pair sequences $(\mathbf{x}_{i,1}, y_{i,1}), ...(\mathbf{x}_{i,n}, y_{i,n}) \in \mathbb{R}^d \times \{-1, 1\}$, $i = 1, ..., K$ with input parameters $b > 0$, and $\lambda > 0$ and $a_i > 1$ $\forall i \in \{1, \cdots, K\}$ then for all $\gamma > 0$, all $\mathbf{u}_i \in \mathbb{R}^d$ and all $n \geq 1$,*

$$\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right] \leq \frac{\sum_{j=1}^{K} a_j}{\gamma}\left[\left(1+\frac{X^2}{2b}\right)\bar{L}_{\gamma,n} + \frac{\left(2b+X^2\right)^2 \tilde{U}^2}{8\gamma b}\right] + \left(2\frac{\lambda}{b}-1\right)\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} a_i A_{i,t}\right],$$

where $X = \max_{i,t}\|\mathbf{x}_{i,t}\|$, $\tilde{U}^2 = \sum_{i=1}^{K}\|\mathbf{u}_i\|^2$ and the expectation is over the random choices of the algorithm.

This bound brings an advantage and disadvantage. We can reduce the last term in the bound choosing the right weights for a_i, but then, we suffer the factor of $\sum_{j=1}^{K} a_j$ in rest part of the bound.

Proof: We follow the proof of Thm. 3 until Eq. (3.14), than we take the expectation of this term. First, we take the expectation of the left hand side where now the conditional expectation of $Z_{i,t}$ changed to

$$\mathbb{E}\left[Z_{i,t}\right] = \frac{1}{D_t}\frac{a_i}{\left(b + |\hat{p}_{i,t}| - \min_{m=1}^{K}|\hat{p}_{m,t}|\right)}.$$

Doing that, the expectation becomes

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=1}^{n} U_{i,t} Z_{i,t}\left(b - y_{i,t}\hat{p}_{i,t} - \min_j |\hat{p}_{j,t}|\right)\right] \\
&= \mathbb{E}\left[\mathbb{E}_{t-1}\left[\sum_{t=1}^{n} U_{i,t} Z_{i,t}\left(b - y_{i,t}\hat{p}_{i,t} - \min_j |\hat{p}_{j,t}|\right)\right]\right] \\
&= \mathbb{E}\left[\sum_{t=1}^{n}\frac{a_i}{D_t}\left(M_{i,t} + \frac{b - |\hat{p}_{j,t}| - \min_j |\hat{p}_{j,t}|}{b + |\hat{p}_{j,t}| - \min_j |\hat{p}_{j,t}|} A_{i,t}\right)\right].
\end{aligned}$$

Reminding that $a_i \geq 1 \forall i$, we can bound $M_{i,t} \leq M_{i,t} a_i$ and get

$$\mathbb{E}\left[\sum_{t=1}^{n} \frac{1}{D_t}\left(M_{i,t} + \frac{b - |\hat{p}_{j,t}| - \min_j |\hat{p}_{j,t}|}{b + |\hat{p}_{j,t}| - \min_j |\hat{p}_{j,t}|} A_{i,t} a_i\right)\right] \leq \frac{2b + X^2}{2\gamma} \bar{L}_{\gamma,i,n}(u_i) + \frac{\left(2b + X^2\right)^2}{8\gamma^2} \|\mathbf{u}_i\|^2 . \tag{3.18}$$

We use now the same bound as in Eq. (3.16) and plug it into the left side of the inequality,

$$\mathbb{E}\left[\sum_{t=1}^{n} \frac{1}{D_t}\left(M_{i,t} + \left(1 - 2\frac{\lambda}{b}\right) A_{i,t} a_i\right)\right] \leq \mathbb{E}\left[\sum_{t=1}^{n} \frac{1}{D_t}\left(M_{i,t} + \frac{b - |\hat{p}_{j,t}| - \min_j |\hat{p}_{j,t}|}{b + |\hat{p}_{j,t}| - \min_j |\hat{p}_{j,t}|} A_{i,t} a_i\right)\right] .$$

Since $b/2 \geq \lambda$ we have that $M_{i,t} + \left(1 - 2\frac{\lambda}{b}\right) A_{i,t} a_i \geq 0$, thus there exists δ_i such that,

$$\mathbb{E}\left[\sum_{t=1}^{n} \frac{1}{D_t}\left(M_{i,t} + \left(1 - 2\frac{\lambda}{b}\right) A_{i,t} a_i\right)\right] = \frac{b}{\delta_i} \mathbb{E}\left[\sum_{t=1}^{n}\left(M_{i,t} + \left(1 - 2\frac{\lambda}{b}\right) A_{i,t} a_i\right)\right] , \tag{3.19}$$

and

$$\frac{b}{\delta_i} \geq \min \frac{1}{D_t} = \frac{1}{\max D_t} \geq \frac{b}{\sum_{j=1}^{K} a_j} ,$$

where the last inequality follows from,

$$D_t = \sum_{j=1}^{K} \frac{a_j}{\left(b + |\hat{p}_{j,t}| - \min_{m=1}^{K} |\hat{p}_{m,t}|\right)} \leq \frac{\sum_{j=1}^{K} a_j}{b}. \tag{3.20}$$

The previous bound implies that $0 < \delta_i \leq \sum_{i=1}^{K} a_i$.

Combining Eq. (3.18) and Eq. (3.19),

$$\frac{b}{\delta_i}\mathbb{E}\left[\sum_{t=1}^{n} M_{i,t}\right] \leq \frac{2b+X^2}{2\gamma}\bar{L}_{\gamma,i,n}(\mathbf{u}_i) + \frac{\left(2b+X^2\right)^2}{8\gamma^2}\|\mathbf{u}_i\|^2 + \frac{b}{\delta_i}\left(2\frac{\lambda}{b}-1\right)a_i\mathbb{E}\left[\sum_{t=1}^{n} A_{i,t}\right]. \quad (3.21)$$

Summing up the last inequality over all K tasks and setting $\delta = \max_i \delta_i$ yields,

$$\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right] \leq \frac{\delta}{\gamma}\left[\left(1+\frac{X^2}{2b}\right)\bar{L}_{\gamma,n} + \frac{\left(2b+X^2\right)^2}{8\gamma b}\sum_{i=1}^{K}\|\mathbf{u}_i\|^2\right] + \left(2\frac{\lambda}{b}-1\right)\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} a_i A_{i,t}\right], \quad (3.22)$$

which concludes the proof. ■

We see that when this bound is a global bound for the last two algorithms. If we set $a_i = 1,\ \forall i \in \{1, \cdots, K\}$ we get the same bound as SHAMPO aggressive algorithm and if we update only when a prediction mistake occurs, it means that $A_{i,t} = 0,\ \forall i \in \{a, \cdots, K\}, t > 0$ and we get the bound of the ordinary SHAMPO perceptron algorithm.

3.4 Kernel-based SHAMPO

3.5 SHAMPO perceptron with Adaptive b parameter

As it has been proved in the analysis and will be shown in experiments, the parameter b can tune the distribution over the tasks between exploration and exploitation, yet, there is still a need to supply the algorithm with this input parameter. So far, all of the tasks sheared the same parameter b which was determined at the beginning of the algorithm and remained constant during the whole process. For this reason, once the initial parameters were supplied, the margin was the only factor that had a varying influence on the distribution. Now, we would like to suggest another algorithm that allow us to dominate the distribution using the number of updates for each task. We assume as in Cesa-Bianchi et al Cesa-Bianchi et al. [2006] that the algorithm makes more mistakes on harder task than easier one, so a natural choice at any time, can be one that set higher probability to the tasks that

made more prediction mistakes up to the same time.

The adaptive algorithm is similar to the one in Fig. 3.1 with three variation in order to comply with this intuition. Here we introduce two more variables per task. First, the scalar $N_{i,n} = \sum_{t=1}^{n} Z_{i,t} M_{i,t}$ that holds the number of updates that has been done for the ith task up to time n. For simplicity of analysis we will write it in it's recursive way $N_{i,t} = N_{i,t-1} + 1$. The second variable is $X_{i,t}$ which is holds at the end of each cycle, the maximal norm of the input vectors that were involved in the updates. The algorithm still have a parameter β to be determined, however, it has less influence on the distribution than the parameter b in Fig. 3.1. Using those three new parameter we introduce the adaptive version of b

$$b_{i,t-1} = \beta X_{i,t} \sqrt{1 + N_{i,t-1}}$$

The adaptive version of the SHAMPO perceptron algorithm is described in Fig. 3.5.

Theorem 5 *If adaptive SHAMPO algorithm runs on K tasks with K parallel example pair sequences $(\mathbf{x}_{i,1}, y_{i,1}), ...(\mathbf{x}_{i,n}, y_{i,n}) \in \mathbb{R}^d \times \{-1, 1\}$, $i = 1, ..., K$ with input parameter $b > 0$, then for all $\gamma > 0$, all $\mathbf{u}_i \in \mathbb{R}^d$ and all $n \geq 1$, there exists $0 < \delta \leq K$, such that,*

$$\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right] \leq \delta\left[\frac{\delta B^2}{2} + \frac{1}{\gamma}\bar{L}_{\gamma,n} + \frac{KR}{2\beta} + B\sqrt{\frac{\delta B^2}{4} + \frac{1}{\gamma}\bar{L}_{\gamma,n} + \frac{KR}{2\beta}}\right]$$

where $R = \max_i \left(\|u_i\| X_i\right)/\gamma$, $B = \left(R + \frac{2+3R}{2\beta}\right)$ and the expectation is over the random choices of the algorithm.

The proof here is more extended version of the proof in Theorem **??** . We will start here from Eq. (3.6). here, we will replace the constant scaling parameter, formerly α, with a time varying factor $c_{i,t-1}/\gamma$, where $c_{i,t} \geq 0$ is defined as following

$$c_{i,t-1} = \frac{1}{2}\left(\max\left\{X_{i,t-1}, ||\mathbf{x}_{i,t}||\right\}\right) + b_{i,t-1}.$$

Plugging the new scaling factor into the given inequality we get

$$M_{i,t}Z_{i,t}(c_{i,t-1}+|\hat{p}_{i,t}|) \leq M_{i,t}Z_{i,t}\frac{c_{i,t-1}}{\gamma}\ell_{\gamma,i,t}(\mathbf{u}_i)+ \\ \frac{1}{2}\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t-1}\right\|^2-\frac{1}{2}\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t}\right\|^2+\frac{M_{i,t}Z_{i,t}}{2}\left\|\mathbf{w}_{i,t-1}-\mathbf{w}_{i,t}\right\|^2.$$

From the update rule of the algorithm we can bound $\|\mathbf{w}_{i,t-1}-\mathbf{w}_{i,t}\|^2 \leq \|\mathbf{x}_{i,t}\|^2$. This inequality holds for any time step, and particular for an update iteration. Using this bound and dividing the inequality by b_{t-1} yields

$$\begin{aligned} M_{i,t}Z_{i,t}\left(\frac{c_{i,t-1}+|\hat{p}_{i,t}|-\|\mathbf{x}_{i,t}\|^2/2}{b_{i,t-1}}\right) &\leq M_{i,t}Z_{i,t}\frac{c_{i,t-1}}{b_{i,t-1}}\frac{\ell_{\gamma,i,t}(\mathbf{u}_i)}{\gamma} \\ &+\frac{1}{2b_{i,t-1}}\left(\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t-1}\right\|^2-\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t}\right\|^2\right). \end{aligned} \tag{3.23}$$

Now, we decompose the last term of the inequality to two differences,

$$\begin{aligned} &\frac{1}{2b_{i,t-1}}\left(\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t-1}\right\|^2-\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t}\right\|^2\right)= \\ &\frac{1}{2b_{i,t-1}}\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t-1}\right\|^2-\frac{1}{2b_{i,t}}\left\|\frac{c_{i,t}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t}\right\|^2+ \\ &\frac{1}{2b_{i,t}}\left\|\frac{c_{i,t}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t}\right\|^2-\frac{1}{2b_{i,t-1}}\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t}\right\|^2. \end{aligned}$$

At this point expand the last two terms and set

and we get

$$\begin{aligned} &\frac{1}{2b_{i,t}}\left\|\frac{c_{i,t}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t}\right\|^2-\frac{1}{2b_{i,t-1}}\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i-\mathbf{w}_{i,t}\right\|^2 \\ &=\frac{\|\mathbf{u}_i\|^2}{2\gamma^2}\left(\frac{c_{i,t}^2}{b_{i,t}}-\frac{c_{i,t-1}^2}{b_{i,t-1}}\right)+\frac{\mathbf{u}_i^T\mathbf{w}_{i,t}}{\gamma}\left(\frac{c_{i,t-1}}{b_{i,t-1}}-\frac{c_{i,t}}{b_{i,t}}\right) \\ &+\frac{\|\mathbf{w}_{i,t}\|^2}{2}\left(\frac{1}{b_{i,t}}-\frac{1}{b_{i,t-1}}\right) \end{aligned}$$

We know that by definition, $b_{i,t}$ is non decreasing over time such that the last term

is always negative and can be waved. In addition, the definition of $c_{i,t}$ leads to

$$\frac{c_{i,t}}{b_{i,t}} = \frac{1}{2\beta\sqrt{1+N_{i,t}}} + 1$$

which, combining with the fact that $N_{i,t}$ is nondecreasing over time as well, we get the inequality

$$\frac{c_{i,t}}{b_{i,t}} \leq \frac{c_{i,t-1}}{b_{i,t-1}}$$

Combining these facts and using $Cauchy - Schwarz$ the inequality becomes,

$$\begin{aligned}\frac{1}{2b_{i,t}}\left\|\frac{c_{i,t}}{\gamma}\mathbf{u}_i - \mathbf{w}_{i,t}\right\|^2 - \frac{1}{2b_{i,t-1}}\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i - \mathbf{w}_{i,t}\right\|^2 \\ \leq \frac{\|\mathbf{u}_i\|^2}{2\gamma^2}\left(\frac{c_{i,t}^2}{b_{i,t}} - \frac{c_{i,t-1}^2}{b_{i,t-1}}\right) + \frac{\|\mathbf{u}_i\|\,\|\mathbf{w}_{i,t}\|}{\gamma}\left(\frac{c_{i,t-1}}{b_{i,t-1}} - \frac{c_{i,t}}{b_{i,t}}\right).\end{aligned} \tag{3.24}$$

Here, recall that on an update trial $y_{i,t}\mathbf{w}_{i,t-1}^T\mathbf{x}_{i,t} \leq 0$ we define $X_i = \max_t \|\mathbf{x}_{i,t}\|$ and use again the update rule and bound the weight vector norm with the number of updates ,

$$\begin{aligned}\|\mathbf{w}_{i,t}\|^2 &= \|\mathbf{w}_{i,t-1} + M_{i,t}Z_{i,t}y_{i,t}\mathbf{x}_{i,t}\|^2 \\ &= \|\mathbf{w}_{i,t-1}\|^2 + 2M_{i,t}Z_{i,t}y_{i,t}\mathbf{w}_{i,t-1}^T\mathbf{x}_{i,t} + M_{i,t}Z_{i,t}\|\mathbf{x}_{i,t}\|^2 \\ &\leq \|\mathbf{w}_{i,t-1}\|^2 + M_{i,t}Z_{i,t}\|\mathbf{x}_{i,t}\|^2 \\ &\leq \|\mathbf{w}_{i,t-1}\|^2 + M_{i,t}Z_{i,t}X_i^2\end{aligned}$$

Applying this inequality recursively, combined with the initial value $\mathbf{w}_{i,0} = 0$ leads us to

$$\|\mathbf{w}_{i,t}\| \leq X_i\sqrt{N_{i,t}} \tag{3.25}$$

which holds for any i and t. Combining the last inequality into Eq. (3.24) we get

$$\begin{aligned}\frac{1}{2b_{i,t}}\left\|\frac{c_{i,t}}{\gamma}\mathbf{u}_i - \mathbf{w}_{i,t}\right\|^2 - \frac{1}{2b_{i,t-1}}\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i - \mathbf{w}_{i,t}\right\|^2 \leq \\ \frac{\|\mathbf{u}_i\|^2}{2\gamma^2}\left(\frac{c_{i,t}^2}{b_{i,t}} - \frac{c_{i,t-1}^2}{b_{i,t-1}}\right) + \frac{\|\mathbf{u}_i\|\,X_i\sqrt{N_{i,t}}}{\gamma}\left(\frac{c_{i,t-1}}{b_{i,t-1}} - \frac{c_{i,t}}{b_{i,t}}\right).\end{aligned} \tag{3.26}$$

We will write now, a simpler bound on the last term. First we note that on t that $M_{i,t}Z_{i,t} = 1$, i.e. in the case of update, $N_{i,t} = N_{i,t-1} + 1$. Using the bound $\sqrt{1+x} - \sqrt{x} \leq \frac{1}{2\sqrt{x}}$, yields

$$\begin{aligned}
\sqrt{N_{i,t}}\left(\frac{c_{t-1}}{b_{t-1}} - \frac{c_t}{b_t}\right) &= \frac{\sqrt{N_{i,t}}}{2\beta}\left(\frac{1}{\sqrt{1+N_{i,t-1}}} - \frac{1}{\sqrt{1+N_{i,t}}}\right) \\
&= \frac{\sqrt{N_{i,t}}}{2\beta}\left(\frac{1}{\sqrt{N_{i,t}}} - \frac{1}{\sqrt{1+N_{i,t}}}\right) \\
&= \frac{1}{2\beta}\frac{\sqrt{1+N_{i,t}} - \sqrt{N_{i,t}}}{\sqrt{1+N_{i,t}}} \\
&\leq \frac{1}{4\beta}\frac{1}{\sqrt{N_{i,t}}\sqrt{1+N_{i,t}}} \\
&\leq \frac{1}{4\beta}\frac{1}{N_{i,t}}.
\end{aligned}$$

If $M_{i,t}Z_{i,t} = 0$, the left hand side becomes 0 because in such case, $c_{i,t} = c_{i,t-1}$ and $b_{i,t} = b_{i,t-1}$, which means that this bound holds for all i and t and we can write it as

$$\sqrt{N_{i,t}}\left(\frac{c_{t-1}}{b_{t-1}} - \frac{c_t}{b_t}\right) \leq \frac{M_{i,t}Z_{i,t}}{4\beta}\frac{1}{N_{i,t}}.$$

Plugging the last bound and Eq. (3.26) into Eq. (3.23) gives us the next inequality that holds for any $i, t, \gamma > 0$, and $u_i \in \mathbb{R}^d$,

$$\begin{aligned}
M_{i,t}Z_{i,t}\left(\frac{c_{i,t-1} + |\hat{p}_{i,t}| - \|\mathbf{x}_{i,t}\|^2/2}{b_{i,t-1}}\right) &\leq M_{i,t}Z_{i,t}\frac{c_{i,t-1}}{b_{i,t-1}}\frac{\ell_{\gamma,i,t}(\mathbf{u}_i)}{\gamma} \\
&+ \frac{1}{2b_{i,t-1}}\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i - \mathbf{w}_{i,t-1}\right\|^2 - \frac{1}{2b_{i,t}}\left\|\frac{c_{i,t}}{\gamma}\mathbf{u}_i - \mathbf{w}_{i,t}\right\|^2 \\
&+ \frac{\|\mathbf{u}_i\|^2}{2\gamma^2}\left(\frac{c_{i,t}^2}{b_{i,t}} - \frac{c_{i,t-1}^2}{b_{i,t-1}}\right) + \frac{\|\mathbf{u}_i\|\, X_i}{\gamma}\frac{M_{i,t}Z_{i,t}}{4\beta}\frac{1}{N_{i,t}}.
\end{aligned}$$

Recall the definition of $c_{i,t}$, we can write $c_{i,t-1} - \|\mathbf{x}_{i,t}\|^2/2 \geq b_{i,t-1}$ and the

ineqality gets the form

$$
\begin{aligned}
M_{i,t}Z_{i,t}\left(\frac{b_{i,t-1}+|\hat{p}_{i,t}|}{b_{i,t-1}}\right) &\leq M_{i,t}Z_{i,t}\frac{c_{i,t-1}}{b_{i,t-1}}\frac{\ell_{\gamma,i,t}(\mathbf{u}_i)}{\gamma} \\
&+ \frac{1}{2b_{i,t-1}}\left\|\frac{c_{i,t-1}}{\gamma}\mathbf{u}_i - \mathbf{w}_{i,t-1}\right\|^2 - \frac{1}{2b_{i,t}}\left\|\frac{c_{i,t}}{\gamma}\mathbf{u}_i - \mathbf{w}_{i,t}\right\|^2 \\
&+ \frac{\|\mathbf{u}_i\|^2}{2\gamma^2}\left(\frac{c_{i,t}^2}{b_{i,t}} - \frac{c_{i,t-1}^2}{b_{i,t-1}}\right) + \frac{\|\mathbf{u}_i\|\,X_i}{\gamma}\frac{M_{i,t}Z_{i,t}}{4\beta}\frac{1}{N_{i,t}}.
\end{aligned}
$$

Summing both sides over $t = 1, \cdots, n$, and taking into consideration the initialization, $w_0 = 0$ we obtain,

$$
\begin{aligned}
\sum_{t=1}^{n} M_{i,t}Z_{i,t}\left(\frac{b_{i,t-1}+|\hat{p}_{i,t}|}{b_{i,t-1}}\right) &\leq \frac{1}{\gamma}\sum_{t=1}^{n} M_{i,t}Z_{i,t}\frac{c_{i,t-1}}{b_{i,t-1}}\ell_{\gamma,i,t}(\mathbf{u}_i) \\
&+ \frac{\|\mathbf{u}_i\|^2}{2\gamma^2}\frac{c_{i,n}^2}{b_{i,n}} - \frac{1}{2b_{i,n}}\left\|\frac{c_{i,n}}{\gamma}\mathbf{u}_i - \mathbf{w}_{i,n}\right\|^2 + \frac{\|\mathbf{u}_i\|\,X_i}{4\beta\gamma}\sum_{t=1}^{n}\frac{M_{i,t}Z_{i,t}}{N_{i,t}}.
\end{aligned} \tag{3.27}
$$

Now, we will bound the terms in the right hand side. The first term can be bounded as following,

$$
\begin{aligned}
\frac{1}{\gamma}\frac{c_{i,t-1}}{b_{i,t-1}}\ell_{\gamma,i,t}(\mathbf{u}_i) &= \frac{1}{\gamma}\left(\frac{1}{2\beta\sqrt{1+N_{i,t-1}}} + 1\right)\ell_{\gamma,i,t}(\mathbf{u}_i) \\
&\leq \frac{1}{2\gamma\beta\sqrt{1+N_{i,t-1}}}\left(\gamma + \|u_i\|X_i\right) + \frac{\ell_{\gamma,i,t}(\mathbf{u}_i)}{\gamma} \\
&= \frac{1}{2\beta\sqrt{1+N_{i,t-1}}}\left(1 + \frac{\|u_i\|X_i}{\gamma}\right) + \frac{\ell_{\gamma,i,t}(\mathbf{u}_i)}{\gamma}
\end{aligned}
$$

Here, we used the fact that $\ell_{\gamma,i,t}(\mathbf{u}_i) \leq \gamma + \|u_i\|X_i$. Now we proceed with the two

middle terms of Eq. (3.27) and get the bound

$$\begin{aligned}
\frac{\|\mathbf{u}_i\|^2}{2\gamma^2}\frac{c_{i,n}^2}{b_{i,n}} - \frac{1}{2b_{i,n}}\left\|\frac{c_{i,n}}{\gamma}\mathbf{u}_i - \mathbf{w}_{i,n}\right\|^2 &= \frac{c_{i,n}u_i^T\mathbf{w}_{i,n}}{b_{i,n}\gamma} - \frac{\|\mathbf{w}_{i,n}\|^2}{2b_{i,n}} \\
&\leq \frac{c_{i,n}u_i^T\mathbf{w}_{i,n}}{b_{i,n}\gamma} \\
&\leq \frac{c_{i,n}}{b_{i,n}}\frac{\|u_i\|\ \|\mathbf{w}_{i,n}\|}{\gamma} \\
&\leq \left(\frac{1}{2\beta\sqrt{1+N_{i,n}}}+1\right)\frac{\|u_i\|X_i\sqrt{N_{i,n}}}{\gamma},
\end{aligned}$$

where we used Eq. (3.25) in the last step. Putting all together in Eq. (3.27) gives us

$$\begin{aligned}
&\sum_{t=1}^{n} M_{i,t}Z_{i,t}\left(\frac{b_{i,t-1}+|\hat{p}_{i,t}|}{b_{i,t-1}}\right) \\
&\qquad \leq \frac{1}{2\beta}\left(1+\frac{\|u_i\|X_i}{\gamma}\right)\sum_{t=1}^{n}\frac{M_{i,t}Z_{i,t}}{\sqrt{1+N_{i,t-1}}} + \frac{1}{\gamma}\sum_{t=1}^{n} M_{i,t}Z_{i,t}\ell_{\gamma,i,t}(\mathbf{u}_i) \\
&\qquad + \left(\frac{1}{2\beta\sqrt{1+N_{i,n}}}+1\right)\frac{\|u_i\|X_i\sqrt{N_{i,n}}}{\gamma} + \frac{\|\mathbf{u}_i\|\ X_i}{4\beta\gamma}\sum_{t=1}^{n}\frac{M_{i,t}Z_{i,t}}{N_{i,t}}.
\end{aligned}$$

The first and the last sums in the right hand side can get more elegant form. Recall $M_{i,t}Z_{i,t} = 1$ implies $N_{i,t} = N_{i,t-1} + 1$, we can write

$$\sum_{t=1}^{n}\frac{M_{i,t}Z_{i,t}}{\sqrt{1+N_{i,t-1}}} = \sum_{t=1}^{n}\frac{M_{i,t}Z_{i,t}}{\sqrt{N_{i,t}}} = \sum_{r=1}^{N_{i,n}}\frac{1}{\sqrt{r}} \leq 2\sqrt{N_{i,n}}.$$

Now, we apply the same bound on the second sum, but with more loose bound,

$$\sum_{t=1}^{n}\frac{M_{i,t}Z_{i,t}}{N_{i,t}} \leq \sum_{t=1}^{n}\frac{M_{i,t}Z_{i,t}}{\sqrt{N_{i,t}}} \leq 2\sqrt{N_{i,n}}.$$

Defining $R_i = (\|u_i\| X_i)/\gamma$, and combining the last three bounds together we get

$$\begin{aligned}
&\sum_{t=1}^{n} M_{i,t} Z_{i,t} \left(\frac{b_{i,t-1} + |\hat{p}_{i,t}|}{b_{i,t-1}} \right) \\
&\qquad \leq \frac{1}{\gamma} \sum_{t=1}^{n} M_{i,t} Z_{i,t} \ell_{\gamma,i,t}(\mathbf{u}_i) + \frac{1}{\beta} \left(1 + \frac{3R_i}{2} \right) \sqrt{N_{i,n}} \\
&\qquad + \frac{R_i}{2\beta} + R_i \sqrt{N_{i,n}}.
\end{aligned}$$

Now, we subtract non negative term from the left side, sum the inequality over all tasks and take the expectation.

$$\begin{aligned}
&\mathbb{E} \left[\sum_{i=1}^{K} \sum_{t=1}^{n} M_{i,t} Z_{i,t} \left(\frac{b_{i,t-1} + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|}{b_{i,t-1}} \right) \right] \\
&\leq \frac{1}{\gamma} \mathbb{E} \left[\sum_{i=1}^{K} \sum_{t=1}^{n} M_{i,t} Z_{i,t} \ell_{\gamma,i,t}(\mathbf{u}_i) \right] + \mathbb{E} \left[\sum_{i=1}^{K} \left(R_i + \frac{2 + 3R_i}{2\beta} \right) \sqrt{N_{i,n}} \right] \\
&\quad + \mathbb{E} \left[\sum_{i=1}^{K} \frac{R_i}{2\beta} \right].
\end{aligned}$$

First we look at the left hand side of the inequality.

$$\sum_{i=1}^{K} \mathbb{E} \left[\sum_{t=1}^{n} M_{i,t} Z_{i,t} \left(\frac{b_{i,t-1} + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|}{b_{i,t-1}} \right) \right] = \sum_{i=1}^{K} \mathbb{E} \left[\sum_{t=1}^{n} \frac{M_{i,t}}{D_t} \right]$$

The bound of D_t now is

$$D_t = \sum_{i=1}^{K} \frac{b_{i,t-1}}{b_{i,t-1} + |\hat{p}_{i,t}| - \min_j |\hat{p}_{j,t}|} \leq K$$

Since $M_{i,t} > 0 \forall i, t$, thus there exists $\delta_i \in \mathbb{R} > 0$ such that

$$\sum_{i=1}^{K} \mathbb{E} \left[\sum_{t=1}^{n} \frac{M_{i,t}}{D_t} \right] = \sum_{i=1}^{K} \frac{1}{\delta_i} \mathbb{E} \left[\sum_{t=1}^{n} M_{i,t} \right] \leq \frac{1}{\delta} \mathbb{E} \left[\sum_{i=1}^{K} \sum_{t=1}^{n} M_{i,t} \right]$$

When $\delta = max_i \delta_i$. Next we handle the middle term of the rhs.

$$\mathbb{E}\left[\sum_{i=1}^{K}\left(R_i + \frac{2+3R_i}{2\beta}\right)\sqrt{N_{i,n}}\right] \leq \mathbb{E}\left[\sum_{i=1}^{K}\left(R + \frac{2+3R}{2\beta}\right)\sqrt{N_{i,n}}\right]$$
$$= \left(R + \frac{2+3R}{2\beta}\right)\mathbb{E}\left[\sum_{i=1}^{K}\sqrt{N_{i,n}}\right],$$

Where $R = \max_i \left(\|u_i\| X_i\right)/\gamma$. Now, considering that $N_{i,n}$ is non-negative and and exploiting the concavity of the square root function we can continue as following

$$\mathbb{E}\left[\sum_{i=1}^{K}\sqrt{N_{i,n}}\right] \leq \mathbb{E}\left[\sqrt{K\sum_{i=1}^{K}N_{i,n}}\right] = \sqrt{K}\mathbb{E}\left[\sqrt{\sum_{i=1}^{K}N_{i,n}}\right]$$
$$\leq \sqrt{K}\sqrt{\mathbb{E}\left[\sum_{i=1}^{K}N_{i,n}\right]}.$$

Here we used the bound for sum of squares, $\sum_{i=1}^{K}\sqrt{a_i} \leq \sqrt{K\sum_{i=1}^{K}a_i}$ for $a_i \in \mathbb{R}, a_i \geq 0$ combining with Jensen inequality. Moreover, we can continue, recall that by definition, $N_{i,n} = \sum_{t=1}^{n} Z_{i,t}M_{i,t}$ and $Z_{i,t}M_{i,t} \leq M_{i,t}$ such that

$$\sqrt{K}\sqrt{\mathbb{E}\left[\sum_{i=1}^{K}N_{i,n}\right]} \leq \sqrt{K}\sqrt{\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n}M_{i,t}\right]}.$$

Plugging the result in the inequality, we get

$$\frac{1}{\delta}\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n}M_{i,t}\right]$$
$$\leq \frac{1}{\gamma}\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n}M_{i,t}Z_{i,t}\ell_{\gamma,i,t}(\mathbf{u}_i)\right] + \mathbb{E}\left[\sum_{i=1}^{K}\left(R_i + \frac{2+3R_i}{2\beta}\right)\sqrt{N_{i,n}}\right] + \mathbb{E}\left[\sum_{i=1}^{K}\frac{R_i}{2\beta}\right]$$
$$\leq \frac{1}{\gamma}\bar{L}_{\gamma,n} + \left(R + \frac{2+3R}{2\beta}\right)\sqrt{K}\sqrt{\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n}M_{i,t}\right]} + \frac{KR}{2\beta}$$

which is quadratic inequality in $\sqrt{\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right]}$. Solving this equation gives us the bound for the cumulative mistakes

$$\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right] \leq \delta\left[\frac{\delta B^2}{2} + \frac{1}{\gamma}\bar{L}_{\gamma,n} + \frac{KR}{2\beta} + B\sqrt{\frac{\delta B^2}{4} + \frac{1}{\gamma}\bar{L}_{\gamma,n} + \frac{KR}{2\beta}}\right]$$

$$\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right] \leq \frac{K}{\gamma}\bar{L}_{\gamma,n} + K^{3/2}\left(R + \frac{2+3R}{2\beta}\right)\sqrt{n} + \frac{K^2R}{2\beta}$$

■

3.6 to do

- to write a code that can find one optimal b for all of the tasks. (to find the knee area.
- to write analysis and run a version of the different betas (as in the ss).
- to write algorithm and run for beat as a function of updates and queries, like $F(u,q) = u^{\alpha}q^{\beta}$. to try in powers $(\alpha, \beta = -2, : 0.5 : 2)$ the private case is the upper case where the $\alpha = 0.5$ and $\beta = 0$.
- try to make a quadratic function of the fraction $\frac{\alpha}{\beta}$.

Parameters: $\beta \in \mathbb{R} > 0$.
Initialize: $\mathbf{w}_{i,0} = \mathbf{0}$ for $i = 1, \ldots, K$
For $t = 1, 2, ..., n$

1. Observe K instance vectors, $\mathbf{x}_{i,t}$, $(i = 1, \ldots, K)$.
2. Compute margins $\hat{p}_{i,t} = \mathbf{w}_{i,t-1}^{\top} \mathbf{x}_{i,t}$.
3. Predict K labels, $\hat{y}_{i,t} = \mathrm{sign}(\hat{p}_{i,t})$.
4. Set $X_{i,t} = \max\left(X_{i,t-1}, \|\mathbf{x}_{i,t}\|\right)$
5. Draw task J_t with the distribution:

$$\Pr\left[J_t = j\right] = \frac{1}{D_t} \frac{b_{i,t}}{b_{i,t} + |\hat{p}_{j,t}| - \min_{m=1}^{K} |\hat{p}_{m,t}|},$$
$$D_t = \sum_i \frac{b_{i,t}}{b_{i,t} + |\hat{p}_{j,t}| - \min_{m=1}^{K} |\hat{p}_{m,t}|}.$$

   Where $b_{i,t} = \beta X_{i,t}^2 \sqrt{1 + N_{i,t}}$.
6. Query the true label ,$y_{J_t,t} \in \{-1, 1\}$.
7. Set the indicator $M_{J_t,t} = 1$ iff $y_{J_t,t} \neq \hat{y}_{J_t,t}$.
8. Update with the perceptron rule:

$$\begin{aligned}
\mathbf{w}_{J_t,t} &= \mathbf{w}_{J_t,t-1} + M_{J_t,t}\, y_{J_t,t}\, \mathbf{x}_{J_t,t} \\
\mathbf{w}_{i,t} &= \mathbf{w}_{i,t-1} \text{ for } i \neq J_t \\
N_{J_t,t} &= N_{J_t,t} + M_{J_t,t}
\end{aligned}$$

End for

Output : $\mathbf{w}_{i,n}$ for $i = 1, \ldots, K$.

Figure 3.5: Adaptive SHAMPO algorithm

Chapter 4

Second Order SHAMPO

This algorithm and proof builds on the second order selective sampling by Cesa-Bianchi et al Cesa-Bianchi et al. [2006] define the function

$$\Theta\left(\left|\hat{p}_{i,t}\right|, r_{i,t}\right) = \left(1 + r_{i,t}\right)\hat{p}_{i,t}^2 + 2\left|\hat{p}_{i,t}\right| - \frac{r_{i,t}}{1 + r_{i,t}} \tag{4.1}$$

In order to understand the way that the aggressive update works, there is a need to examine the function $\Theta\left(\left|\hat{p}_{i,t}\right|, r_{i,t}\right)$. One can see that this function is quadratic in $\hat{p}_{i,t}$, such that it becomes negative in a close interval. Solving this quadratic form, shows that

$$\Theta\left(\left|\hat{p}_{i,t}\right|, r_{i,t}\right) \leq 0 \Leftrightarrow \left|\hat{p}_{i,t}\right| \leq \theta(r_{i,t}) = \frac{-1 + \sqrt{1 + r_{i,t}}}{1 + r_{i,t}}. \tag{4.2}$$

The last equation shows that the condition on the function $\Theta\left(\left|\hat{p}_{j,t}\right|, r_{j,t}\right)$ can be translated to threshold on the margin. First, a task is chosen according to the distribution above, then, for margin that is less the threshold, an aggressive update will be issued whereas for margin that is bigger than that, the algorithm will performed update only when there is a prediction error. For all the rest of the task that were not chosen, no update is issued. For now, we would like to analyse the possibilities for one task in a certain iteration. There are two extreme cases: First, when the the algorithm has not get yet input examples the similar to the current one, which cause the maximal uncertainty i.e., $r_{i,t} = 1$. In this case, the aggressiveness threshold is maximal as well, and we get $\max \theta(r_{i,t}) = \frac{-1+\sqrt{2}}{2} \approx 0.21$ and an aggressive update will be issued only when the margin is less than this value. Another interesting

case is when we saw the same example many times before, such the uncertainty in the prediction now is relatively low, which means $r_{i,t} \approx 0$. In this scenario, the aggressive update will be issued only when the margin is zero, $\hat{p}_{i,t} = 0$.

Proof:
Before proofing the Thm. we will use the next inequality. define $x \in [0, 1]$

$$\sqrt{1-x} + \sqrt{1+x} \leq 2. \tag{4.3}$$

From the concavity of $\sqrt{1+x}$ we see that $\sqrt{1+x} \leq 1 + \frac{1}{2}x$. using this inequality twice, we get $\sqrt{1-x} + \sqrt{1+x} \leq 1 - \frac{1}{2}x + 1 + \frac{1}{2}x = 2$.

Define

$$\Phi_{i,t}(\mathbf{u}_i) = \frac{1}{2}\|\mathbf{u}_i\|^2 + \frac{1}{2}\sum_{s=1}^{t} Z_{i,t}U_{i,t}\left(y_{i,t} - \mathbf{u}_i^T\mathbf{x}_{i,t}\right)^2$$

Simlar to the proof of Thm .3 of Cesa-Bianchi et al Cesa-Bianchi et al. [2006] and Forster ????,

$$\begin{aligned}
\frac{1}{2}Z_{i,t}U_{i,t}\left(y_{i,t} - \hat{p}_{i,t}\right)^2 =& \inf_{\mathbf{u}_i}\Phi_{i,t+1}(\mathbf{u}_i) - \inf_{\mathbf{u}_i}\Phi_{i,t}(\mathbf{u}_i) + \frac{Z_{i,t}U_{i,t}}{2}\mathbf{x}_{i,t}^T A_{i,t}^{-1}\mathbf{x}_{i,t} \\
&- \frac{Z_{i,t}U_{i,t}}{2}\mathbf{x}_{i,t}^T A_{i,t-1}^{-1}\mathbf{x}_{i,t}\hat{p}_{i,t}^2 \\
=& \inf_{\mathbf{u}_i}\Phi_{i,t+1}(\mathbf{u}_i) - \inf_{\mathbf{u}_i}\Phi_{i,t}(\mathbf{u}_i) + \frac{Z_{i,t}U_{i,t}}{2}\frac{r_{i,t}}{1+r_{i,t}} - \frac{Z_{i,t}U_{i,t}}{2}r_{i,t}\hat{p}_{i,t}^2
\end{aligned}$$

Now, we sum up the equation over t,

$$\begin{aligned}
\frac{1}{2}\sum_{t=1}^{n} Z_{i,t}U_{i,t}\left(y_{i,t} - \hat{p}_{i,t}\right)^2 =& \inf_{\mathbf{u}_i}\Phi_{i,n+1}(\mathbf{u}_i) + \sum_{t=1}^{n}\frac{Z_{i,t}U_{i,t}}{2}\frac{r_{i,t}}{1+r_{i,t}} - \sum_{t=1}^{n}\frac{Z_{i,t}U_{i,t}}{2}r_{i,t}\hat{p}_{i,t}^2 \\
\leq& \frac{1}{2}\|\mathbf{u}_i\|^2 + \frac{1}{2}\sum_{t=1}^{n} Z_{i,t}U_{i,t}\left(y_{i,t} - \mathbf{u}_i^T\mathbf{x}_{i,t}\right)^2 + \sum_{t=1}^{n}\frac{Z_{i,t}U_{i,t}}{2}\frac{r_{i,t}}{1+r_{i,t}} \\
&- \sum_{t=1}^{n}\frac{Z_{i,t}U_{i,t}}{2}r_{i,t}\hat{p}_{i,t}^2.
\end{aligned}$$

For simplification we will define

$$A_{i,n} = I + \sum_{t=1}^{n} Z_{i,t} U_{i,t} \mathbf{x}_{i,t} \mathbf{x}_{i,t}^T.$$

Expanding the squares we get,

$$\begin{aligned}
\frac{1}{2} \sum_{t=1}^{n} Z_{i,t} U_{i,t} \left(\hat{p}_{i,t}^2 - 2 y_{i,t} \hat{p}_{i,t} - \frac{r_{i,t}}{1 + r_{i,t}} + r_{i,t} \hat{p}_{i,t}^2 \right) \\
\leq & \frac{1}{2} \|\mathbf{u}_i\|^2 + \sum_{t=1}^{n} Z_{i,t} U_{i,t} \mathbf{u}_i^T \mathbf{x}_{i,t} \mathbf{x}_{i,t}^T \mathbf{u}_i - \sum_{t=1}^{n} Z_{i,t} U_{i,t} \mathbf{u}_i^T \mathbf{x}_{i,t} y_{i,t} \\
= & \frac{1}{2} \mathbf{u}_i^T \left(I + \sum_{t=1}^{n} Z_{i,t} U_{i,t} \mathbf{x}_{i,t} \mathbf{x}_{i,t}^T \right) \mathbf{u}_i - \sum_{t=1}^{n} Z_{i,t} U_{i,t} \mathbf{u}_i^T \mathbf{x}_{i,t} y_{i,t} \\
= & \frac{1}{2} \mathbf{u}_i^T A_{i,n} \mathbf{u}_i - \sum_{t=1}^{n} Z_{i,t} U_{i,t} \mathbf{u}_i^T \mathbf{x}_{i,t} y_{i,t}.
\end{aligned} \tag{4.4}$$

The vectors $\mathbf{u}_i$ can be replaced with their scaled version, $c\mathbf{u}_i$. introducing the trivial inequality, $1 - x \leq \max\{1 - x, 0\}$ we get

$$\begin{aligned}
c Z_{i,t} U_{i,t} \left(1 - \mathbf{u}_i^T \mathbf{x}_{i,t} y_{i,t}\right) \leq c Z_{i,t} U_{i,t} \max\left\{1 - \mathbf{u}_i^T \mathbf{x}_{i,t} y_{i,t}, 0\right\} \\
-c Z_{i,t} U_{i,t} \mathbf{u}_i^T \mathbf{x}_{i,t} y_{i,t} \leq -c Z_{i,t} U_{i,t} + c\ell\left(\mathbf{u}_i^T \mathbf{x}_{i,t} y_{i,t}\right).
\end{aligned} \tag{4.5}$$

Rearranging an plugging Eq. (4.6) and Eq. (4.4)

$$\begin{aligned}
\frac{1}{2} \sum_{t=1}^{n} Z_{i,t} U_{i,t} \left(\hat{p}_{i,t}^2 - 2 y_{i,t} \hat{p}_{i,t} - \frac{r_{i,t}}{1 + r_{i,t}} + r_{i,t} \hat{p}_{i,t}^2 + 2c \right) \\
\leq \frac{c^2}{2} \mathbf{u}_i^T A_{i,n} \mathbf{u}_i + c \sum_{t=1}^{n} Z_{i,t} U_{i,t} \ell\left(\mathbf{u}_i^T \mathbf{x}_{i,t} y_{i,t}\right)
\end{aligned} \tag{4.6}$$

Recall that $U_{i,t} = M_{i,t} + G_{i,t}$ we will split the inequality into two different cases. First we will take into consideration the cases when an error update was performed, i.e.$M_{i,t} = 1$, in which we have $-y_{i,t}\hat{p}_{i,t} = |\hat{p}_{i,t}|$. In this case we need to consider also two subcases, when $\Theta\left(|\hat{p}_{i,t}|, r_{i,t}\right) \geq 0$ and when $\Theta\left(|\hat{p}_{i,t}|, r_{i,t}\right) < 0$. Beginning with the former subcase, recall that for this case $\mathbb{E}_{t-1}\left[Z_{i,t}\right] =$

$\frac{1}{D_t}\frac{2c}{2c+(\Theta(|\hat{p}_{j,t}|,r_{j,t}))}$, we get

$$\begin{aligned}
&\mathbb{E}\left[Z_{i,t}U_{i,t}\left(\hat{p}_{i,t}^2 - 2y_{i,t}\hat{p}_{i,t} - \frac{r_{i,t}}{1+r_{i,t}} + r_{i,t}\hat{p}_{i,t}^2 + 2c\right)\right] \\
&= \mathbb{E}\left[\mathbb{E}_{t-1}\left[Z_{i,t}\right]U_{i,t}\left(\hat{p}_{i,t}^2 - 2y_{i,t}\hat{p}_{i,t} - \frac{r_{i,t}}{1+r_{i,t}} + r_{i,t}\hat{p}_{i,t}^2 + 2c\right)\right] \\
&= 2c\mathbb{E}\left[\frac{1}{D_t}U_{i,t}\right].
\end{aligned}$$

When $\Theta\left(|\hat{p}_{i,t}|, r_{i,t}\right) < 0$ the conditional expectation becomes $\mathbb{E}_{t-1}\left[Z_{i,t}\right] = \frac{1}{D_t}$ and the thus,

$$\begin{aligned}
&\mathbb{E}\left[Z_{i,t}U_{i,t}\left(\hat{p}_{i,t}^2 - 2y_{i,t}\hat{p}_{i,t} - \frac{r_{i,t}}{1+r_{i,t}} + r_{i,t}\hat{p}_{i,t}^2 + 2c\right)\right] \\
&= \mathbb{E}\left[\mathbb{E}_{t-1}\left[Z_{i,t}\right]U_{i,t}\left(\hat{p}_{i,t}^2 + 2\left|\hat{p}_{i,t}\right| - \frac{r_{i,t}}{1+r_{i,t}} + r_{i,t}\hat{p}_{i,t}^2 + 2c\right)\right] \\
&\geq \mathbb{E}\left[\mathbb{E}_{t-1}\left[Z_{i,t}\right]U_{i,t}\left(-\frac{r_{i,t}}{1+r_{i,t}} + 2c\right)\right] \\
&\geq 2c\mathbb{E}\left[\frac{1}{D_t}U_{i,t}\right] - \frac{r_{i,t}}{1+r_{i,t}}\mathbb{E}\left[\frac{1}{D_t}\right].
\end{aligned}$$

Now we examine the case where an update was performed, but there was no mistake. In this case, $0 \leq y_{i,t}\hat{p}_{i,t}$ and the aggressive update was performed. Recall the bound on the margin for such case and using Eq. (4.6) , we bound the margin as follows

$$0 \leq y_{i,t}\hat{p}_{i,t} \leq \theta(r_{i,t}) = \frac{-1+\sqrt{1+r_{i,t}}}{1+r_{i,t}}.$$

We can bound now,

$$\begin{aligned}
&\hat{p}_{i,t}^2 - 2y_{i,t}\hat{p}_{i,t} - \frac{r_{i,t}}{1+r_{i,t}} + r_{i,t}\hat{p}_{i,t}^2 + 2c \\
&\quad = (1+r_{i,t})\hat{p}_{i,t}^2 - 2y_{i,t}\hat{p}_{i,t} + \frac{r_{i,t}}{1+r_{i,t}} - 2\frac{r_{i,t}}{1+r_{i,t}} + 2c \\
&\quad = f(y_{i,t}\hat{p}_{i,t}) - 2\frac{r_{i,t}}{1+r_{i,t}} + 2c
\end{aligned}$$

where $f(y_{i,t}\hat{p}_{i,t}) = (1+r_{i,t})\hat{p}_{i,t}^2 - 2y_{i,t}\hat{p}_{i,t} + \frac{r_{i,t}}{1+r_{i,t}}$ is a quadratic convex function with two non-negative roots $\frac{1\pm\sqrt{1-r_{i,t}}}{1+r_{i,t}}$ and we know that the margin is lower than

the smaller root , $y_{i,t}\hat{p}_{i,t} \leq \frac{1-\sqrt{1-r_{i,t}}}{1+r_{i,t}}$ which leads to the inequality $f(y_{i,t}\hat{p}_{i,t}) \geq 0$ so we bound

$$\begin{aligned}
&\mathbb{E}\left[Z_{i,t}U_{i,t}\left(\hat{p}_{i,t}^2 - 2y_{i,t}\hat{p}_{i,t} - \frac{r_{i,t}}{1+r_{i,t}} + r_{i,t}\hat{p}_{i,t}^2 + 2c\right)\right] \\
&= \mathbb{E}\left[\mathbb{E}_{t-1}\left[Z_{i,t}\right]U_{i,t}\left(f(y_{i,t}\hat{p}_{i,t}) - 2\frac{r_{i,t}}{1+r_{i,t}} + 2c\right)\right] \\
&\geq \mathbb{E}\left[\mathbb{E}_{t-1}\left[Z_{i,t}\right]U_{i,t}\left(-2\frac{r_{i,t}}{1+r_{i,t}} + 2c\right)\right] \\
&\geq 2c\mathbb{E}\left[\frac{1}{D_t}U_{i,t}\right] - \frac{2r_{i,t}}{1+r_{i,t}}\mathbb{E}\left[\frac{1}{D_t}\frac{2r_{i,t}}{1+r_{i,t}}\right].
\end{aligned}$$

Summarize the results we get,

$$\begin{aligned}
&\frac{1}{2}\sum_{t=1}^{n}\mathbb{E}\left[Z_{i,t}U_{i,t}\left(\hat{p}_{i,t}^2 - 2y_{i,t}\hat{p}_{i,t} - \frac{r_{i,t}}{1+r_{i,t}} + r_{i,t}\hat{p}_{i,t}^2 + 2c\right)\right] \\
&\quad\geq c\sum_{t\in\mathcal{M}}\mathbb{E}\left[\frac{1}{D_t}U_{i,t}\right] + c\sum_{t\in\mathcal{G}}\mathbb{E}\left[\frac{1}{D_t}U_{i,t}\right] \\
&\quad- \frac{1}{2}\sum_{t\in\mathcal{A}\cap\mathcal{M}}\frac{r_{i,t}}{1+r_{i,t}}\mathbb{E}\left[\frac{1}{D_t}\right] - \sum_{t\in\mathcal{A}\cap\mathcal{G}}\frac{r_{i,t}}{1+r_{i,t}}\mathbb{E}\left[\frac{1}{D_t}\right]
\end{aligned} \tag{4.7}$$

Combining the result of the last inequality with the expectation of the Eq. (4.6), recall that $\sum_{t\in\mathcal{M}} U_{i,t} = M_i$, and $\sum_{t\in\mathcal{G}} U_{i,t} = G_i$ we get,

$$\begin{aligned}
&c\sum_{t\in\mathcal{M}}\mathbb{E}\left[\frac{1}{D_t}U_{i,t}\right] + c\sum_{t\in\mathcal{G}}\mathbb{E}\left[\frac{1}{D_t}U_{i,t}\right] - \frac{1}{2c}\sum_{t\in\mathcal{A}\cap\mathcal{M}}\frac{r_{i,t}}{1+r_{i,t}}\mathbb{E}\left[\frac{1}{D_t}\right] \\
&-\frac{1}{c}\sum_{t\in\mathcal{A}\cap\mathcal{U}}\frac{r_{i,t}}{1+r_{i,t}}\mathbb{E}\left[\frac{1}{D_t}\right] \leq \frac{c}{2}\mathbf{u}_i^T\mathbb{E}\left[A_{i,n}\right]\mathbf{u}_i + \sum_{t=1}^{n}\mathbb{E}\left[Z_{i,t}U_{i,t}\ell\left(\mathbf{u}_i^T\mathbf{x}_{i,t}y_{i,t}\right)\right]
\end{aligned}$$

The normalization can also been bound by

$$D_t = 2c\sum_{i=1}^{K}\left(2c + \left(\Theta\left(\left|\hat{p}_{i,t}\right|, r_{i,t}\right)\right)_+\right)^{-1} \leq 2c\sum_{m=1}^{K}\frac{1}{2c} = K$$

which leads to

$$\mathbb{E}\left[M_i\right] + \mathbb{E}\left[G_i\right] - \frac{1}{2c}\mathbb{E}\sum_{t\in\mathcal{A}\cap\mathcal{M}}\left[\frac{r_{i,t}}{1+r_{i,t}}\right] - \frac{1}{c}\mathbb{E}\sum_{t\in\mathcal{A}\cap\mathcal{U}}\left[\frac{r_{i,t}}{1+r_{i,t}}\right]$$
$$\leq \frac{Kc}{2}\mathbf{u}_i^T\mathbb{E}\left[A_{i,n}\right]\mathbf{u}_i + K\sum_{t=1}^{n}\mathbb{E}\left[Z_{i,t}U_{i,t}\ell\left(\mathbf{u}_i^T\mathbf{x}_{i,t}y_{i,t}\right)\right]$$

$$\begin{aligned}\mathbb{E}\left[M_i\right] \leq & \frac{Kc}{2}\mathbf{u}_i^T\mathbb{E}\left[A_{i,n}\right]\mathbf{u}_i + K\sum_{t=1}^{n}\mathbb{E}\left[Z_{i,t}U_{i,t}\ell\left(\mathbf{u}_i^T\mathbf{x}_{i,t}y_{i,t}\right)\right] - \mathbb{E}\left[G_i\right] \\ & + \frac{1}{2c}\mathbb{E}\sum_{t\in\mathcal{A}\cap\mathcal{M}}\left[\frac{r_{i,t}}{1+r_{i,t}}\right] + \frac{1}{c}\mathbb{E}\sum_{t\in\mathcal{A}\cap\mathcal{U}}\left[\frac{r_{i,t}}{1+r_{i,t}}\right] \\ \leq & \frac{Kc}{2}\mathbf{u}_i^T\mathbb{E}\left[A_{i,n}\right]\mathbf{u}_i + K\mathbb{E}\left[\sum_{t=1}^{n}Z_{i,t}U_{i,t}\ell\left(\mathbf{u}_i^T\mathbf{x}_{i,t}y_{i,t}\right)\right] - \mathbb{E}\left[G_i\right] \\ & + \frac{1}{c}\sum_{t\in\mathcal{A}}\left[\frac{r_{i,t}}{1+r_{i,t}}\right]\end{aligned}$$

Now, we can summarize the inequality over all of the tasks,

$$\begin{aligned}\mathbb{E}\left[M\right] \leq & \frac{cK}{2}\sum_{i=1}^{K}\mathbf{u}_i^T\mathbb{E}\left[A_{i,n}\right]\mathbf{u}_i + K\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n}Z_{i,t}U_{i,t}\ell\left(\mathbf{u}_i^T\mathbf{x}_{i,t}y_{i,t}\right)\right] - \mathbb{E}\left[G\right] \\ & + \frac{1}{c}\mathbb{E}\sum_{i=1}^{K}\sum_{t\in\mathcal{A}}\left[\frac{r_{i,t}}{1+r_{i,t}}\right]\end{aligned}$$

$$\begin{aligned}\mathbb{E}\left[M\right] \leq & \frac{cK}{2}\sum_{i=1}^{K}\mathbf{u}_i^T\mathbb{E}\left[A_{i,n}\right]\mathbf{u}_i + K\bar{L}_{1,n} - \mathbb{E}\left[G\right] \\ & + \frac{1}{c}\mathbb{E}\sum_{i=1}^{K}\sum_{t\in\mathcal{A}}\left[\frac{r_{i,t}}{1+r_{i,t}}\right]\end{aligned}$$

4.1 From Multi-task to Contextual Bandits

Although our algorithm is designed for many binary-classification tasks that can be independent, it can also be applied in two settings of contextual bandits, when

one decoupling exploration and exploitation is allowed in every cycle as in Yu and Mannor [2009], Avner et al. [2012]. The problem of this setting is predicting a label $\hat{Y}_t \in \{1, \ldots, C\}$ given an input $\mathbf{x}_t$. As before, the algorithm works in rounds. On round t the algorithm receives an input $\mathbf{x}_t$ and outputs multicalss label $\hat{Y}_t \in \{1, \ldots, C\}$. Then, it queries for some information about the label via a single binary "yes-no" question, and uses the feedback to update its model. We consider here two forms of binary questions.

4.1.1 One-vs-Rest

In the first setting, termed *one-vs-rest*, the algorithm asks if the true label is some label $\bar{Y}_t \in \{1, \ldots, C\}$, possibly not the predicted label, i.e. it may be the case that $\bar{Y}_t \neq \hat{Y}_t$. Given the response whether $\bar{Y}_t$ is indeed the true label Y_t, the algorithm updates its models. The reduction we perform is by introducing K tasks to be done in parallel, one per class, such that $K = C$. The ith task is to decide whether the true label of the current input $\mathbf{x}_t$ is class i or not. Even though, for the most of the tasks, we can not deduce the true labeling $Y_t \in \{1, \ldots, C\}$, considering a single task prediction (actually it can be done only for one specific task at a time), still the binary prediction, gives us the ability to eliminate one of the classes. For example: in the case of numbers OCR, if the true label of an input image is 6, the first task is asking if the predicted label should be 0 or not, the answer of this question is indeed negative, so the true label may not be 0, but we still don't know what it is. Given the output of all (binary) classifiers, the algorithm generates a single multi-class prediction to be the single label for which the output of the corresponding binary classifier is positive. If such class does not exist, or there are more than one such classes, a random prediction is used, that is, given an input $\mathbf{x}_t$ we define $\hat{Y}_t = \arg\max_i \hat{p}_{i,t}$, where ties are broken arbitrarily . Given the feedback whether $\bar{Y}_t$ is indeed the true label Y_t, the algorithm updates its models. The label to be queried is $\bar{Y}_t = J_t$, i.e. the problem index that SHAMPO is querying. We now analyze the performance of this reduction as a multiclass prediction algorithm.

Corollary 6 *Assume the SHAMPO algorithm is executed as above with $K = C$ one-vs-rest tasks, on a sequence $(\mathbf{x}_1, Y_1), ...(\mathbf{x}_n, Y_n) \in \mathbb{R}^d \times \{1, ..., C\}$, and input parameter $b > 0$. Then for all $\gamma > 0$ and all $\mathbf{u}_i \in \mathbb{R}^d$, the expected number of*

multi-class errors is bounded as follows

$$\mathbb{E}\left[\sum_t [\![Y_t \neq \hat{Y}_t]\!]\right] \leq \frac{C}{\gamma}\left[\left(1+\frac{X^2}{2b}\right)\bar{L}_{\gamma,n} + \frac{\left(2b+X^2\right)^2 U^2}{8\gamma b}\right],$$

where $[\![A]\!] = 1$ *if the predicate* A *is true, and zero otherwise.*

The corollary follows directly from Theorem **??** by noting that, $[\![Y_t \neq \hat{Y}_t]\!] \leq \sum_i M_{i,t}$. That is, there is a multiclass mistake if there is at least one prediction mistake of one of the one-vs-rest problems. The closest setting to the presented setting is the one of contextual bandits, yet we allow decoupling of exploration and exploitation. Ignoring this decoupling, the Banditron algorithm Kakade et al. [2008] is the closest to ours, with a regret of $O(T^{2/3})$. Hazan et al Hazan and Kale [2011] proposed an algorithm with $O(\sqrt{T})$ regret but designed for the $\log$ loss, with coefficient that may be very large, and another Crammer and Gentile [2013] algorithm has $O(\sqrt{T})$ regret with respect to prediction mistakes, yet they assumed stochastic labeling, rather than adversarial.

4.1.2 One-vs-One

In the second setting, termed by *one-vs-one*, the algorithm picks two labels $\bar{Y}_t^+, \bar{Y}_t^- \in \{1 \dots C\}$, possibly both not the predicted label. The feedback for the learner is three-fold, it is $y_{J_t,t} = +1$ if the first alternative is the correct label, $\bar{Y}_t^+ = Y_t$. The feedback is $y_{J_t,t} = -1$ if the second alternative is the correct label, $\bar{Y}_t^- = Y_t$, and it is $y_{J_t,t} = 0$ otherwise (in this case there is no error and we set $M_{J_t,t} = 0$). The reduction we perform is by introducing $K = \binom{C}{2}$ problems, one per pair of classes. The goal of the learning algorithm for a problem indexed with two labels (y_1, y_2) is to decide which is the correct label, given it is one of the two. Given the output of all (binary) classifiers the algorithm generates a single multi-class prediction using a tournament in a round-robin approach Fürnkranz [2002]. If there is no clear winner, a random prediction is used. We now analyze the performance of this reduction as a multiclass prediction algorithm,

Corollary 7 *Assume the SHAMPO algorithm is executed as above, with* $K = \binom{C}{2}$ *one-vs-one problems, on a sequence* $(\mathbf{x}_1, Y_1), ...(\mathbf{x}_n, Y_n) \in \mathbb{R}^d \times \{1, ..., C\}$*, and input parameter* $b > 0$*. Then for all* $\gamma > 0$ *and all* $\mathbf{u}_i \in \mathbb{R}^d$*, the expected number*

of multi-class errors can be bounded as follows

$$\mathbb{E}\left[\sum_t [\![Y_t \neq \hat{Y}_t]\!]\right] \leq$$
$$\frac{2\binom{C}{2}}{(\binom{C}{2}-1)/2+1}\frac{1}{\gamma}\left[\left(1+\frac{X^2}{2b}\right)\bar{L}_{\gamma,n}+\frac{\left(2b+X^2\right)^2U^2}{8\gamma b}\right]$$

where $[\![A]\!] = 1$ *if the predicate* A *is true, and zero otherwise.*

The corollary follows directly from Theorem **??** by noting that, $[\![Y_t \neq \hat{Y}_t]\!] \leq \frac{2}{(\binom{C}{2}-1)/2+1}\sum_{i=1}^{\binom{C}{2}} M_{i,t}$. Note, that the bound is essentially independent of C as the coefficient in the bound is upper bounded by 6 for $C \geq 3$.

We conclude this section with two algorithmic modifications, which we employ in this setting. Currently, when the feedback is zero, there is no update of the weights for those problems, because there is no error. This causes the algorithm to effectively ignore such (many) examples, as in these cases the algorithm is not modifying any model, and furthermore, if such example is repeated, a problem with possibly "0" feedback may be queried again. We fix this issue with one of the two modifications. In the first alternative, if the feedback is zero, we modify the model to reduce the chance that the chosen problem, J_t, would be chosen again for the same input (i.e. not to make the same wrong-choice of choosing irrelevant problem again). To this end, we modify the weights a bit, to increase the confidence (absolute margin) of the model for the same input, and replace Eq. (3.3) with, $\mathbf{w}_{J_t,t} = \mathbf{w}_{J_t,t-1} + [\![y_{J_t,t} \neq 0]\!]\, y_{J_t,t}\, \mathbf{x}_{J_t,t} + [\![y_{J_t,t} = 0]\!]\eta\hat{y}_{J_t,t}\mathbf{x}_{J_t,t}$, for some $\eta > 0$. In other words, if there is a possible error (i.e. $y_{J_t,t} \neq 0$) the update follows the Perceptron's rule. Otherwise, the weights are updated such that the absolute margin will increase, as $|\mathbf{w}_{J_t,t}^\top \mathbf{x}_{J_t,t}| = |(\mathbf{w}_{J_t,t-1} + \eta\hat{y}_{J_t,t}\mathbf{x}_{J_t,t})^\top \mathbf{x}_{J_t,t}| = |\mathbf{w}_{J_t,t-1}^\top \mathbf{x}_{J_t,t} + \eta\text{sign}(\mathbf{w}_{J_t,t-1}^\top \mathbf{x}_{J_t,t})\|\mathbf{x}_{J_t,t}\|^2| = |\mathbf{w}_{J_t,t-1}^\top \mathbf{x}_{J_t,t}| + \eta\|\mathbf{x}_{J_t,t}\|^2 > |\mathbf{w}_{J_t,t-1}^\top \mathbf{x}_{J_t,t}|$. We call this method *one-vs-one-weak*, as it performs weak updates when there is zero feedback. The second alternative is not to allow a zeroed value feedback, and if this is the case, to set the label to be either $+1$ or -1, randomly. Both alternates are evaluated below. We call this method *one-vs-one-random*.

Bibliography

Orly Avner, Shie Mannor, and Ohad Shamir. Decoupling exploration and exploitation in multi-armed bandits. In *ICML*, 2012.

Nicolò Cesa-Bianchi, Claudio Gentile, and Luca Zaniboni. Worst-case analysis of selective sampling for linear classification. *The Journal of Machine Learning Research*, 7:1205–1230, 2006.

Koby Crammer and Claudio Gentile. Multiclass classification with bandit feedback using adaptive regularization. *Machine Learning*, 90(3):347–383, 2013.

Johannes Fürnkranz. Round robin classification. *Journal of Machine Learning Research*, 2:721–747, 2002.

Elad Hazan and Satyen Kale. Newtron: an efficient bandit algorithm for online multiclass prediction. *Advances in Neural Information Processing Systems (NIPS)*, 2011.

Sham M Kakade, Shai Shalev-Shwartz, and Ambuj Tewari. Efficient bandit algorithms for online multiclass prediction. In *Proceedings of the 25th international conference on Machine learning*, pages 440–447. ACM, 2008.

Simon Tong and Daphne Koller. Support vector machine active learning with application sto text classification. In *ICML*, pages 999–1006, 2000.

Jia Yuan Yu and Shie Mannor. Piecewise-stationary bandit problems with side observations. In *ICML*, 2009.