Multi-task Learning with a Shared Annotator

Haim Cohen

Supervised by Prof. Koby Crammer

Faculty of Electrical Engineering, Technion
Israel Institute of Technology

13.12.2015

Outline

1. Introduction
   - Problem statement
   - Related work
   - Work guidelines

Outline

1. Introduction
   - Problem statement
   - Related work
   - Work guidelines
2. First Order Algorithms
   - Perceptron SHAMPO
   - Aggressive perceptron SHAMPO
   - SHAMPO with prior
   - Mistakes bound

Outline

1. Introduction
   - Problem statement
   - Related work
   - Work guidelines
2. First Order Algorithms
   - Perceptron SHAMPO
   - Aggressive perceptron SHAMPO
   - SHAMPO with prior
   - Mistakes bound
3. Second Order Algorithm

Outline

1. Introduction
   - Problem statement
   - Related work
   - Work guidelines
2. First Order Algorithms
   - Perceptron SHAMPO
   - Aggressive perceptron SHAMPO
   - SHAMPO with prior
   - Mistakes bound
3. Second Order Algorithm
4. Variants

Outline

1. Introduction
   - Problem statement
   - Related work
   - Work guidelines
2. First Order Algorithms
   - Perceptron SHAMPO
   - Aggressive perceptron SHAMPO
   - SHAMPO with prior
   - Mistakes bound
3. Second Order Algorithm
4. Variants
5. Experiments

Outline

1. Introduction
   - Problem statement
   - Related work
   - Work guidelines
2. First Order Algorithms
   - Perceptron SHAMPO
   - Aggressive perceptron SHAMPO
   - SHAMPO with prior
   - Mistakes bound
3. Second Order Algorithm
4. Variants
5. Experiments
6. Conclusion

Online Learning

- Input comes in sequence
- Feedback after prediction
- Uses when :
  - Data comes in sequence
  - Big data
- Examples: stock market, advertisement, content recommendation etc.

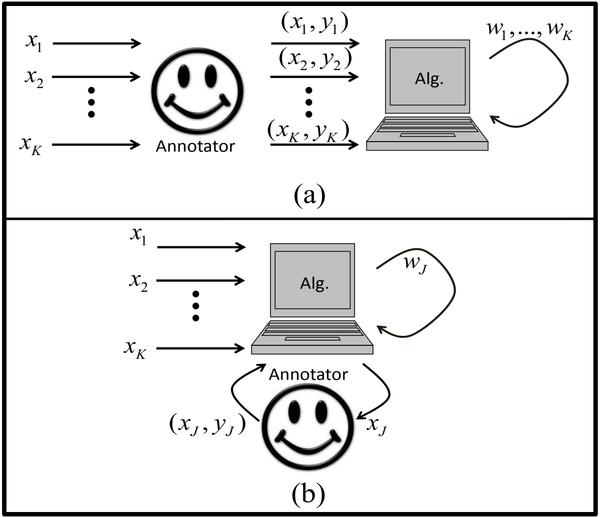
Online Learning

- On each round:
  1. Instance $\mathbf{x}_t$ is observed
  2. Prediction $\hat{y}_t$ is made
  3. Loss ℓ_t is suffered
  4. True value y_t is revealed
  5. An update of the model is made
- Loss types: zero-one, hinge, exponential, quadratic...

Problem statement

- K binary learning tasks in parallel
- Limited resources (limited bandwidth)
- Annotate one task at a time
- Examples: classify data news from many agencies

Problem statement - update



Related Work on multitask learning

- [Evgeniou et al., 2004] [Argyriou et al., 2008]

  Used regularization for multitask learning

- [Collobert et al., 2008]

  Focused on learning neural networks for NLP

Assume relation between tasks

Selective Sampling

Problem

- Labeling is expensive - consume resources (time, money, etc.)

Solution

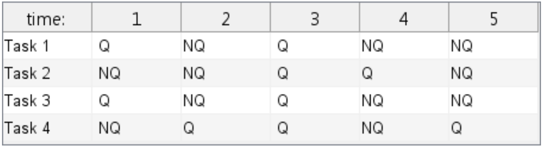
- Only some labels are queried, others remain unknown
- Two questions: When should we query? How to update?

In our problem, the question ”when”, becomes ”which task”

[Cesa-Bianchi et al., 2006, 2009], [Fruend et al., 1997] , [Crammer , 2014]

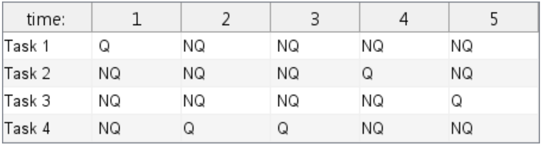
Selective sampling-example

Selective sampling setting:



time:	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	NQ	Q	Q	NQ	Q

Our setting:



time:	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	Q	NQ	NQ

Q=Queried, NQ=Not Queried

In this work

- Propose ways for feedback selection (to answer the question ”which task?”).
- Devise SHAMPO algorithms - SHared Annotator for Multiple PrOblems
- Analyze - mistakes bound
- Empirical study that strengthen the algorithms

Feedback selection - guidelines

How to issue a query?

- Ask when wrong prediction is assumed
- Two possible ways:
  - Similarity to previous examples
  - Prediction is not distinctive

Problem Setting

Problem setup:

- K binary tasks to be learned
- $\mathbf{x}_{i,t} \in \mathbb{R}^{d_i}, \quad i \in \{1, \cdots, K\}$, - instance vector
- $y_{i,t} \in \{\pm 1\}$ - label

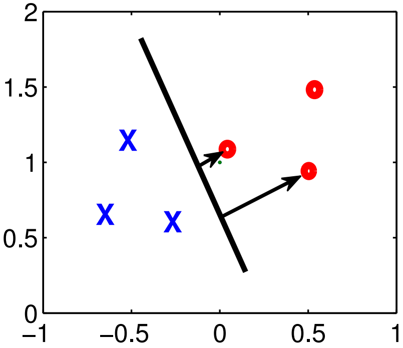
Perceptron SHAMPO - definitions

- Linear classifier $\mathbf{w}_{i,t} \in \mathbb{R}^d$
- Margin $\hat{p}_{i,t} = \mathbf{w}_{i,t-1}^\top \mathbf{x}_{i,t}$
- Predicted label $\hat{y}_{i,t} = \mathrm{sign}(\hat{p}_{i,t})$
- Mistake indicator $M_{i,t} = \mathbb{I}\left[\hat{y}_{i,t} \neq y_{i,t}\right] \in \{0,1\}$
- Query indicator $Z_{i,t} \in \{0,1\}$
  - $\sum_{i=1}^{K} Z_{i,t} = 1 \;\; , \forall t$

Perceptron SHAMPO

Margin can measure certainty

- large $|\hat{p}_{i,t}|$ ⇒ high certainty
- small $|\hat{p}_{i,t}|$ ⇒ low certainty



perceptron SHAMPO

Define J_t - the chosen task in time t

The probability to query the task $j \in \{1 \cdots , K\}$ is:

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

$$\text{for } \; D_t = \sum_{i=1}^{K} \left(b + |\hat{p}_{i,t}| - \min_{m} |\hat{p}_{m,t}|\right)^{-1}, \quad b > 0 \in \mathbb{R}$$

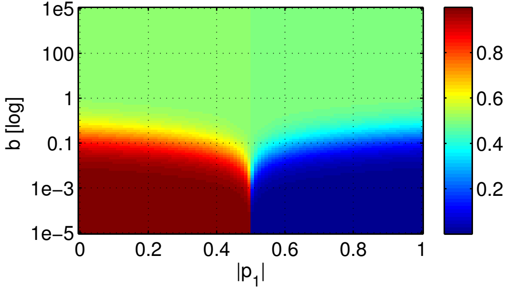
- Large $b \, (b >> 0) \Rightarrow$ uniform distribution - exploration
- Small $b \, (b \rightarrow 0) \Rightarrow$ delta distribution - exploitation
- The data need to be scaled into a ball with the same norm

Probability example

Example of the distribution over 2 tasks.

Fix $|\hat{p}_{2,t^*}| = 0.5$

The probability to choose task 1 is:



perceptron SHAMPO - probability

Advantages of random selection:

- A few tasks has similar margin
- To cope with adversary
- To get exploration-exploitation

perceptron SHAMPO

Initialize: $\mathbf{w}_{i,0} = \mathbf{0}$, $b \in \mathbb{R} > 0$

On each round t, the algorithm:

- Observes K instances $\mathbf{x}_{i,t}$
- Predicts K labels $\hat{y}_{i,t} = \mathrm{sign}(\mathbf{w}_{i,t-1}^{\top}\mathbf{x}_{i,t})$
- Chooses a task to query with probability $\Pr\left[J_t = j\right]$
- Query the label $y_{J_t,t}$
- Sets $M_{J_t} = 1 \quad$ iff $\quad \hat{y}_{J_t,t} \neq y_{J_t,t}$
- Updates : $\quad \mathbf{w}_{J_t,t} = \mathbf{w}_{J_t,t-1} + M_{J_t,t}\, y_{J_t,t}\, \mathbf{x}_{J_t,t}$

Aggressive perceptron SHAMPO

- Aggressive update: correct prediction but low margin.
- $\lambda \in \mathbb{R} > 0$, - aggressiveness threshold
- Aggressive update indicator $G_{i,t} = \mathbb{I}\left[|\hat{p}_{i,t}| < \lambda, M_{i,t} = 0\right] \in \{0, 1\}$
- Update indicator $U_{i,t} = M_{i,t} + G_{i,t} \in \{0, 1\}$

SHAMPO with prior

If we have a prior knowledge about the tasks, we prefer to change the distribution to:

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

with the appropriate normalization factor D_t

- The ”prior” parameters $a_j \geq 1$ (only for the analysis)
- Large $b \Rightarrow$ prior distribution
- Small $b \Rightarrow$ delta distribution

Perceptron SHAMPO loss

- $\mathbf{u}_i \in \mathbb{R}^d$ is arbitrary hyperplane
- Hinge loss function $\ell_{\gamma,i,t}(\mathbf{u}_i) = \left(\gamma - y_{i,t}\mathbf{u}_i^\top \mathbf{x}_{i,t}\right)_+ \;,\quad \gamma > 0$
- Expected loss over updates up to time T

$$\bar{L}_{\gamma,T} = \mathbb{E}\left[\sum_{t=1}^{T}\sum_{i=1}^{K} Z_{i,t}U_{i,t}\ell_{\gamma,i,t}(\mathbf{u}_i)\right]$$

- $\tilde{U}^2 = \sum_{i=1}^{K} \left\|\mathbf{u}_i\right\|^2 \;,\quad X = \max_{i,t}\left\|\mathbf{x}_{i,t}\right\|$

Perceptron SHAMPO bound

Expected mistakes bound

There exist $0 < \delta \leq \sum_{i=1}^{K} a_i$ such that the expected number of mistakes of the perceptron SHAMPO up to time T can be bounded as follows:

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

- A good choice $\lambda < b/2$
- When $\lambda \to 0 \Longrightarrow \mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} G_{i,t}\right] \to 0$

Second order SHAMPO

Adapting the RLS (Regularized Least squares) estimator from regression to binary classification, where:

- $A_{i,0} = I_{d\times d}$
- $A_{i,t} = \left(A_{i,t-1} + U_{i,t} Z_{i,t} \mathbf{x}_{i,t} \mathbf{x}_{i,t}^{\top}\right) \in R^{d\times d}$
- $\mathbf{w}_{i,t} = \mathbf{w}_{i,t-1} + U_{i,t} Z_{i,t} y_{i_t} \mathbf{x}_{i,t} \in R^{d}$

A_t can be viewed as covariance or ”confidence ” matrix

[Cesa-Bianchi et al. 2006, Crammer 2014]

Second order SHAMPO

Initialize: $\mathbf{w}_{i,0} = \mathbf{0}$, $A_0 = I, \ \ b \in \mathbb{R} > 0$

On each round t

- Observe K instances $\mathbf{x}_{i,t}$
- Compute K margins $\hat{p}_{i,t} = \mathbf{x}_{i,t}^T \left(A_{i,t-1} + \mathbf{x}_{i,t}\mathbf{x}_{i,t}^T \right)^{-1} \mathbf{w}_{i,t-1}$
- Predict K labels $\hat{y}_{i,t} = \mathrm{sign}\left(\hat{p}_{i,t}\right)$
- Query the label $y_{J_t,t}$ with same probability $\Pr\left[J_t = j\right]$ as in first order,
- Update : $\mathbf{w}_{J_t,t} \ \ , \ \ A_{J_t,t}$

Second order SHAMPO

Expected mistake bound

There exists $0 < \delta \leq K$, such that the expected number of mistakes of the second order perceptron SHAMPO up to time T can be bounded as follows:

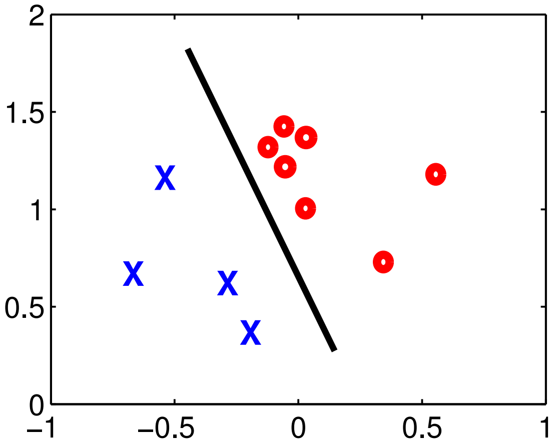
$$\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{n} M_{i,t}\right]$$

$$\leq \frac{\delta}{\gamma}\bar{L}_{\gamma,n}(\mathbf{u}_i) + \frac{\delta b}{2\gamma^2}\sum_{i=1}^{K}\mathbf{u}_i^T\mathbb{E}\left[A_{i,n}\right]\mathbf{u}_i + \frac{\delta}{2b}\sum_{i=1}^{K}\sum_{k=1}^{d}\mathbb{E}\left[\ln\left(1+\lambda_{i,k}\right)\right]$$

- $\lambda_{i,k}$ is the i^th eigenvalue of the matrix $A_{i,n}$

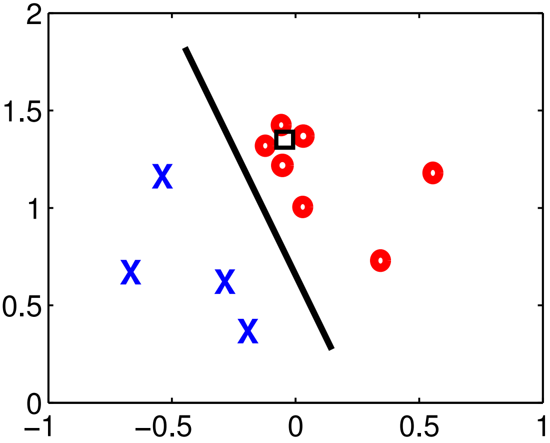
Second order SHAMPO - Aggressive

This is the state in time t



Second order SHAMPO - Aggressive

On $t+1$ we get a new example (black square). What will be it's label?



Second order SHAMPO - Aggressive

Define $r_{i,t} = \mathbf{x}_{i,t}^T A_{i,t-1}^{-1} \mathbf{x}_{i,t}$

- $r_{i,t}$ - the confidence in the prediction of $\hat{y}_{i,t}$
- Large $r_{i,t} \Rightarrow$ low confidence
- Small $r_{i,t} \Rightarrow$ high confidence
- If $\|\mathbf{x}_{i,t}\|^2 = 1, \forall i, t$ then $0 < r_{i,t} \leq 1$

Second order SHAMPO - Aggressive

Define:

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

We build a new distribution:

$$\Pr\left[J_t = j\right] = \frac{1}{D_t}\frac{1}{\left(b + F\left(\left|\hat{p}_{i,t}\right|, r_{i,t}\right)_+\right)} \quad \forall j \in \{1 \cdots, K\}$$

$$\text{Where} \;\; D_t = \sum_{i=1}^{K}\left(b + F\left(\left|\hat{p}_{i,t}\right|, r_{i,t}\right)_+\right)^{-1}$$

- $F\left(\left|\hat{p}_{i,t}\right|, r_{i,t}\right) \leq 0$ iff $\left|\hat{p}_{i,t}\right| \leq \frac{-1+\sqrt{1+r_{i,t}}}{1+r_{i,t}} \leq \frac{-1+\sqrt{2}}{2} \approx 0.3$ (aggressive)

Second order SHAMPO - Aggressive

Initialize: $\mathbf{w}_{i,0}= \mathbf{0},\ A_0 = I,\ \ b \in \mathbb{R} > 0$

On each round t

- Observe K instances $\mathbf{x}_{i,t}$
- Compute K margins $\hat{p}_{i,t} = \mathbf{x}_{i,t}^T \left(A_{i,t-1} + \mathbf{x}_{i,t}\mathbf{x}_{i,t}^T\right)^{-1} \mathbf{w}_{i,t-1}$
- Predict K labels $\hat{y}_{i,t} = \mathrm{sign}\left(\hat{p}_{i,t}\right)$
- Query the label $y_{J_t,t}$ with probability $\Pr\left[J_t = j\right]$ above,
- If $F\left(\left|\hat{p}_{J_t,t}\right|, r_{J_t,t}\right) \leq 0$ or $\hat{y}_{J_t,t} \neq y_{J_t,t}$ set $U_{J_t,t} = 1$
- Update : $\mathbf{w}_{J_t,t}$, $A_{J_t,t}$ iff $U_{iJ_tt} = 1$

Contextual Bandits - decoupling of exploration and exploitation

The problem: predicting a label $\hat{Y}_t \in \{1, \ldots, C\}$ given an input $\mathbf{x}_t$

- Tasks are related
- Query a single binary question and update the model

We consider two forms :

- One vs. one
- One vs. rest

[Kakade et al., 2008], [Hazan et al. 2012]

One vs. rest

- There are $K = C$ binary tasks

On each round t:

- Observe a single input $\mathbf{x}_t$
- Compute K margins $\hat{p}_{i,t}$ for binary tasks
- Predict the multiclass label, $\hat{Y}_t = \arg\max_i \hat{p}_{i,t}$
- Choose the label (task) to query on $\bar{Y}_t = J_t$
- Update

One vs. rest

Expected mistake bound

There exists $0 < \delta \leq \sum_{i=1}^{C} a_i$ such that the expected number of mistakes of the One vs. Rest contextual SHAMPO bandit can be bounded as:

$$\mathbb{E}\left[\sum_t [\![Y_t \neq \hat{Y}_t]\!]\right]$$
$$\leq \frac{\delta}{\gamma}\left[\left(1 + \frac{X^2}{2b}\right)\bar{L}_{\gamma,T} + \frac{\left(2b + X^2\right)^2 \tilde{U}^2}{8\gamma b}\right] + \left(2\frac{\lambda}{b} - 1\right)\mathbb{E}\left[\sum_{i=1}^{K}\sum_{t=1}^{T} a_i G_{i,t}\right],$$

- This bound comes from $\mathbb{I}\left[Y_t \neq \hat{Y}_t\right] \leq \sum_i M_{i,t}$

One vs. one

- There are $K = \binom{C}{2}$ binary tasks.

At each round, the algorithm:

- Gets a single input $\mathbf{x}_t$
- Computes K predictions $\hat{y}_{i,t}$ for binary tasks
- Predicts the multiclass label $\hat{Y}_t$, by tournament.
- Chooses pair of labels (task) to query on $\{\bar{Y}_t^+, \bar{Y}_t^-\}$ assigned with J_t
- Updates

One vs. One

Expected mistake bound

There exists $0 < \delta \leq \sum_{i=1}^{\binom{C}{2}} a_i$ such that the expected number of mistakes of the One vs. One contextual SHAMPO bandit is:

$$\mathbb{E}\left[\sum_t [\![Y_t \neq \hat{Y}_t]\!]\right] \leq \frac{2}{(\binom{C}{2}-1)/2+1} \times$$

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

- This bound is follows from $\mathbb{I}\left[Y_t \neq \hat{Y}_t\right] \leq \frac{2}{(\binom{C}{2}-1)/2+1} \sum_{i=1}^{\binom{C}{2}} M_{i,t}$ [Allwein et al., 2000]
- The bound coefficient is upper bounded by 4.

One vs. One

What if the prediction is not a mistake, nor correct , i.e.$y_{J_t,t} = 0$?

- No update - this task will be chosen again
- Random update - not allow zero feedback (only -1 or 1)
- Weak update - increases the margin using $\eta > 0$

$$\mathbf{w}_{J_t,t} = \mathbf{w}_{J_t,t-1} + \mathbb{I}\left[y_{J_t,t} \neq 0\right]\, y_{J_t,t}\, \mathbf{x}_{J_t,t} + \mathbb{I}\left[y_{J_t,t} = 0\right] \eta \hat{y}_{J_t,t} \mathbf{x}_{J_t,t}$$

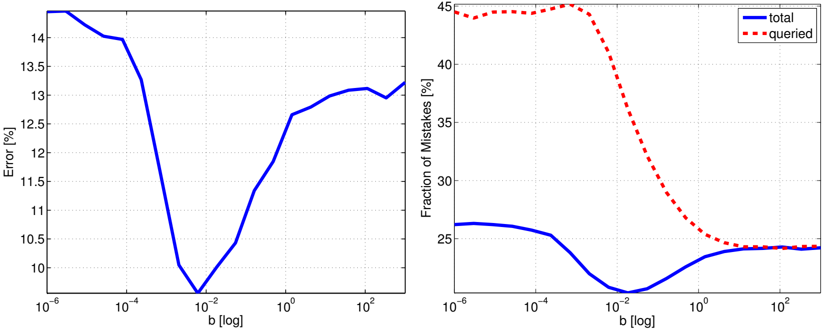
$$\begin{aligned} \left|\mathbf{w}_{J_t,t}^\top \mathbf{x}_{J_t,t}\right| &= \left|\left(\mathbf{w}_{J_t,t-1} + \eta \hat{y}_{J_t,t} \mathbf{x}_{J_t,t}\right)^\top \mathbf{x}_{J_t,t}\right| \\ &= \left|\mathbf{w}_{J_t,t-1}^\top \mathbf{x}_{J_t,t} + \eta \operatorname{sign}(\mathbf{w}_{J_t,t-1}^\top \mathbf{x}_{J_t,t}) \|\mathbf{x}_{J_t,t}\|^2\right| \\ &= \left|\mathbf{w}_{J_t,t-1}^\top \mathbf{x}_{J_t,t}\right| + \eta \|\mathbf{x}_{J_t,t}\|^2 > \left|\mathbf{w}_{J_t,t-1}^\top \mathbf{x}_{J_t,t}\right| \end{aligned}$$

Experiments - Data sets

- OCR - USPS (7,291 train /2,007 test,d=256), MNIST(60,000 train/10,000 test,$d = 784$)
  - One vs. Rest - 10 tasks
  - One vs. One - 45 tasks
- Vowel prediction - Vocal Joystick (572,911 train /236,680 test, $d = 27$)
  - One vs. Rest - 8 tasks
  - One vs. One - 28 tasks
- NLP - sentiment analysis - 36 tasks (266,645 examples, $8,768 \leq d \leq 1,447,866$)

One vs. Rest - USPS dataset

Right values correspond to pure exploration, while left values to pure exploitation. The only thing we see is the red curve. The ”knee” can show the area of the tradeoff b.



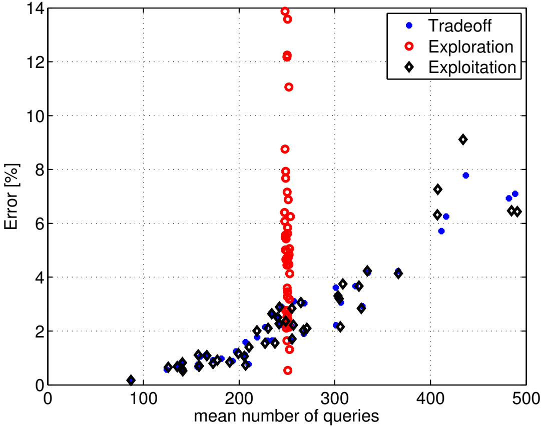
(a) Test error

(b) Train errors

Test error vs. number of queries

MNIST - One vs. One data

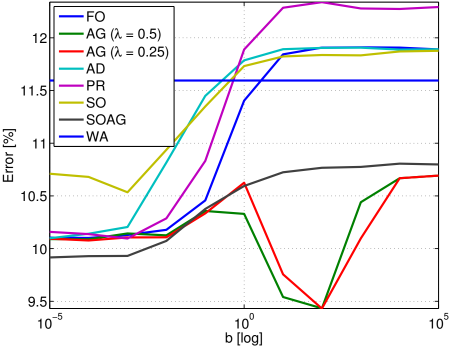
The tradeoff shows less errors for the appropriate queries distribution



Error vs. b - different algorithms

VJ one-vs-one

Comparison between different algorithms. First Order, Aggressive, Adaptive, Prior, Second order, Second order aggressive and ”Watch All”. All algorithms show the same behavior.



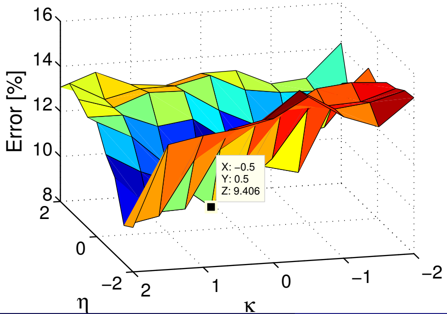
Adaptive b

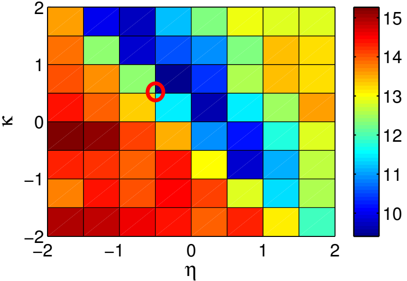
USPS, One vs. One data

The plots show mean error for adaptive b algorithm with $b_{i,t} = (N_{i,t})^{\kappa}(\sum_t Z_{i,t})^{\eta}$

Where $N_{i,t}$ is the number of updates on task i up to time t.

We see that a good choice is $b_{i,t} = \sqrt{(N_{i,t})/(\sum_t Z_{i,t})}$





Conclusion

- We introduced algorithms to solve the multi-task learning with a shared annotator
- We analyzed the algorithms in the mistake bound model
- We showed a variation of our SHAMPO algorithms to contextual bandits - decoupling of exploration and exploitation.
- Experiments that show that SHAMPO algorithms can acheive good results even with partial feedback and focuses on the hard tasks, were presented.

Thanks

- My beloved wife Zohar, for the support
- Prof. Koby Crammer for the guidance and supervising
- My friend from the machine learning group
- Intel for the generous sponsoring

Questions ???