Online Discriminative Learning
Theory and Applications

Nicolò Cesa-Bianchi

Università degli Studi di Milano

Summary

1. The online protocol
2. Structured classification
3. Active learning
4. Multiview learning

Summary

1. The online protocol
2. Structured classification
3. Active learning
4. Multiview learning

Theory of repeated games

James Hannan

David Blackwell

Learning to play a game (1956)

Play a game repeatedly against a possibly suboptimal opponent

Zero-sum 2-person games played more than once

	1	2	$\dots$	M
1	$\ell(1,1)$	$\ell(1,2)$	$\dots$	
2	$\ell(2,1)$	$\ell(2,2)$	$\dots$	
$\vdots$	$\vdots$	$\vdots$	$\ddots$	
N				

$N \times M$ known loss matrix

- Row player (player) has N actions
- Column player (opponent) has M actions

For each game round $t = 1, 2, \dots$

- Player chooses action i_t and opponent chooses action y_t
- The player suffers loss $\ell(i_t, y_t)$ (= gain of opponent)

Zero-sum 2-person games played more than once

$$\begin{array}{c|cccc|}
 & 1 & 2 & \dots & M \\
\hline
1 & \ell(1,1) & \ell(1,2) & \dots & \\
2 & \ell(2,1) & \ell(2,2) & \dots & \\
\vdots & \vdots & \vdots & \ddots & \\
N & & & & \\
\hline
\end{array}$$

$N \times M$ known loss matrix

- Row player (player) has N actions
- Column player (opponent) has M actions

For each game round $t = 1, 2, \dots$

- Player chooses action i_t and opponent chooses action y_t
- The player suffers loss $\ell(i_t, y_t)$ (= gain of opponent)

Player can learn from opponent's history of past choices $y_1, \dots, y_{t-1}$

Prediction with expert advice, 1989

Volodya Vovk

Manfred Warmuth

- Opponent's moves $y_1, y_2, \ldots$ define a nonstochastic sequential prediction problem
- Loss matrix can change with time
- Design a player's strategy that predicts any sequence $y_1, y_2, \ldots$ nearly as well as the single best action for that sequence

Exponentially weighted forecaster

At time t pick action i with probability proportional to

$$\exp\left(-\eta \, \mathrm{Loss}_{i,t}\right)$$

where $\mathrm{Loss}_{i,t}$ is total loss of action i up to now

Theorem [C-B, 1997]

The average per-round expected loss of the forecaster converges to that of the best action for the observed sequence at rate

$$\sqrt{\frac{2}{T} \ln N}$$

where N is number of experts and T is the number of time steps

Exponentially weighted forecaster

At time t pick action i with probability proportional to

$$\exp\left(-\eta \, \mathrm{Loss}_{i,t}\right)$$

where $\mathrm{Loss}_{i,t}$ is total loss of action i up to now

Theorem [C-B, 1997]

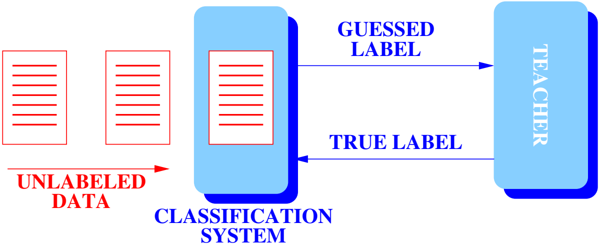
The average per-round expected loss of the forecaster converges to that of the best action for the observed sequence at rate

$$\sqrt{\frac{2}{T} \ln N}$$

where N is number of experts and T is the number of time steps

No dependence on number of opponent's actions!

From game theory to online learning

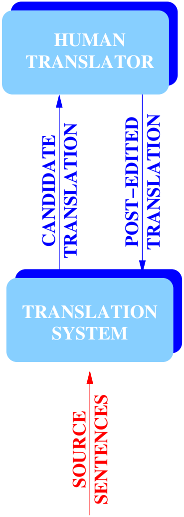


- Add side info to opponent's moves $(x_1, y_1), (x_2, y_2), \ldots \quad x_t \in \mathbb{R}^d$
- Linear classifiers $\boldsymbol{w}$ which predict using $\quad \boldsymbol{w}^\top \boldsymbol{x}_t$
- A repeated game between the player choosing action $\boldsymbol{w}_t \in \mathbb{R}^d$ and the opponent choosing action (x_t, y_t)
- Convergence to performance of best linear classifier under no statistical assumptions on data source

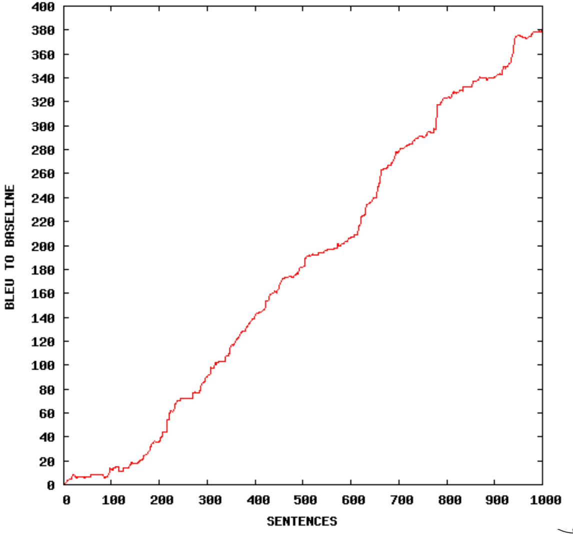
Advantages of online algorithms

- Simple algorithms for learning linear models
- Scalable
- Robust: game-theoretic performance guarantees
- Versatile:
    - structured classification
    - active learning
    - matrix learning
    - tracking
    - bounded memory learning

Online protocol naturally exploits interaction



Computer Assisted Translation



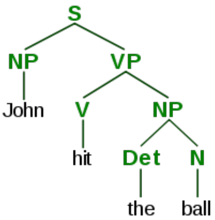
Summary

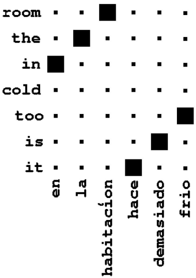
1. The online protocol
2. Structured classification
3. Active learning
4. Multiview learning

Structured Classification

A combinatorial label space (sequences, trees)

- POS tagging: sentence → sequence of POS tags
- Parsing: sentence → parse tree
- Bilingual alignment: sentence pair → alignment (matching)
- Letter to phoneme: word → phoneme sequence
- Phrase-based translation: source sentence → target sentence



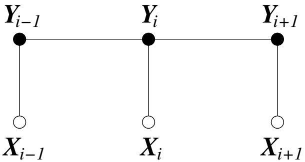


Conditional Random Fields [Lafferty, McCallum and Pereira, 2001]

- A graphical model $\mathbb{P}(\boldsymbol{y} \mid \boldsymbol{x})$ for predicting a set of hidden variables $\boldsymbol{y} = (y_1, \dots, y_m)$ given observation $\boldsymbol{x}$
- The conditional dependency of $\boldsymbol{y}$ on $\boldsymbol{x}$ is defined through feature functions $f_j(\boldsymbol{x}, \boldsymbol{y}_i) \in \mathbb{R}$
- $\boldsymbol{y}_1 \dots, \boldsymbol{y}_m$ are subparts of $\boldsymbol{y}$ (e.g., single nodes, linked nodes)

Some simple examples

- (n-gram, POS tag)
- NP $\longrightarrow$ DT NN
- (le chat, the cat)



Conditional Random Fields – Inference

Loglinear model for the joint distribution of labels

$$\mathbb{P}(\boldsymbol{y} \mid \boldsymbol{x}) = \exp\left(\sum_{i,j} w_j \, f_j(\boldsymbol{x}, \boldsymbol{y}_i) - \ln Z_{\boldsymbol{x}}\right) = \exp\left(\boldsymbol{w}^\top \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{y}) - \ln Z_{\boldsymbol{x}}\right)$$

Decoding

$$\widehat{\boldsymbol{y}} = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}(\boldsymbol{x})} \mathbb{P}(\boldsymbol{y} \mid \boldsymbol{x}) = \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}(\boldsymbol{x})} \exp\left(\boldsymbol{w}^\top \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{y})\right)$$

$$= \operatorname*{argmax}_{\boldsymbol{y} \in \mathcal{Y}(\boldsymbol{x})} \sum_{i,j} w_j \, f_j(\boldsymbol{x}, \boldsymbol{y}_i)$$

$\mathcal{Y}(\boldsymbol{x})$ is the set of feasible labels for observation $\boldsymbol{x}$

Recall

Prediction via decoding $\quad \widehat{y} = \operatorname*{argmax}_{y \in \mathcal{Y}(x)} w^\top \mathbf{f}(x, y)$

Stochastic gradient ascent

Estimate w by maximizing the log-likelihood of a training set $(x_1, y_1), \ldots, (x_T, y_T)$

$$w \leftarrow w + \underbrace{\mathbf{f}(x_t, y_t) - \mathbb{E}_Y\big[\mathbf{f}(x_t, Y)\big]}_{\nabla \ln \mathbb{P}(y_t \mid x_t)} \qquad \text{for } t = 1, \ldots, T$$

Gradient ascent and the Perceptron [Collins, 2002]

Recall

Prediction via decoding $\widehat{y} = \underset{y \in \mathcal{Y}(x)}{\operatorname{argmax}} \, w^\top f(x, y)$

Stochastic gradient ascent

Estimate w by maximizing the log-likelihood of a training set $(x_1, y_1), \ldots, (x_T, y_T)$

$$w \leftarrow w + \underbrace{f(x_t, y_t) - \mathbb{E}_Y\big[f(x_t, Y)\big]}_{\nabla \ln \mathbb{P}(y_t \mid x_t)} \qquad \text{for } t = 1, \ldots, T$$

Structured Perceptron

A Viterbi approximation $f(x_t, \widehat{y}_t)$ of the expectation $\mathbb{E}_Y\big[f(x_t, Y)\big]$

$$w \leftarrow w + f(x_t, y_t) - f(x_t, \widehat{y}_t)$$

Remarks

Structured Perceptron

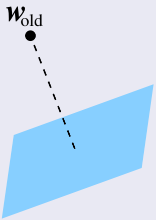
$$\widehat{y}_t = \underset{y \in \mathcal{Y}(x)}{\operatorname{argmax}} \, w_t^\top f(x_t, y_t) \qquad\qquad w_{t+1} = w_t + f(x_t, y_t) - f(x_t, \widehat{y}_t)$$

- Simplest example of online linear algorithm
- Generates an ensemble $\mathbf{0} = w_1, w_2, \ldots, w_T$ of classifiers by performing multiple epochs over the training set
- Average classifier: $\displaystyle \frac{1}{T} \sum_t w_t$ has typically low risk
- If training set is i.i.d., then the statistical risk can be provably bounded $\rightarrow$ online to batch conversion

Margin optimization associated with the 1-best label for x_t

$$\min_{\boldsymbol{w}} \quad \left(\|\boldsymbol{w} - \boldsymbol{w}_{\text{old}}\|^2 + C\,\xi_t\right)$$

$$\text{s.t.} \quad \underbrace{\boldsymbol{w}^\top \mathbf{f}(x_t, y_t) - \boldsymbol{w}^\top \mathbf{f}(x_t, \widehat{y}_t) \geqslant 1 - \xi_t}_{\text{linear constraint}}$$

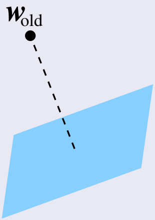


An optimization view [Crammer et al., 2006]

Margin optimization associated with the 1-best label for x_t

$$\min_{\boldsymbol{w}} \quad \left(\|\boldsymbol{w} - \boldsymbol{w}_{\text{old}}\|^2 + C\,\xi_t\right)$$

$$\text{s.t.} \quad \underbrace{\boldsymbol{w}^\top \mathbf{f}(\boldsymbol{x}_t, \boldsymbol{y}_t) - \boldsymbol{w}^\top \mathbf{f}(\boldsymbol{x}_t, \widehat{\boldsymbol{y}}_t) \geqslant 1 - \xi_t}_{\text{linear constraint}}$$



The solution is the Passive-Aggressive algorithm

$$\boldsymbol{w} = \boldsymbol{w}_{\text{old}} + \eta_t \big(\mathbf{f}(\boldsymbol{x}_t, \boldsymbol{y}_t) - \mathbf{f}(\boldsymbol{x}_t, \widehat{\boldsymbol{y}}_t)\big)$$

the learning rate η_t has a closed form expression

More global updates

- Gradient ascent update tends to enforce all margin constraints
$$\boldsymbol{w}^\top \mathbf{f}(\mathbf{x}_t, \mathbf{y}_t) - \boldsymbol{w}^\top \mathbf{f}(\mathbf{x}_t, \mathbf{y}) \geqslant 1 \quad \text{for all } \mathbf{y} \neq \mathbf{y}_t$$
- Perceptron addresses only $\boldsymbol{w}^\top \mathbf{f}(\mathbf{x}_t, \mathbf{y}_t) - \boldsymbol{w}^\top \mathbf{f}(\mathbf{x}_t, \widehat{\mathbf{y}}_t) \geqslant 1$
- Consider the N-best labels $\mathbf{y}$ in the ranking induced by $\boldsymbol{w}^\top \mathbf{f}(\mathbf{x}_t, \mathbf{y})$

More global updates

- Gradient ascent update tends to enforce all margin constraints

$$\boldsymbol{w}^\top \mathbf{f}(x_t, \boldsymbol{y}_t) - \boldsymbol{w}^\top \mathbf{f}(x_t, \boldsymbol{y}) \geqslant 1 \quad \text{for all } \boldsymbol{y} \neq \boldsymbol{y}_t$$

- Perceptron addresses only $\boldsymbol{w}^\top \mathbf{f}(x_t, \boldsymbol{y}_t) - \boldsymbol{w}^\top \mathbf{f}(x_t, \widehat{\boldsymbol{y}}_t) \geqslant 1$
- Consider the N-best labels $\boldsymbol{y}$ in the ranking induced by $\boldsymbol{w}^\top \mathbf{f}(x_t, \boldsymbol{y})$

MIRA algorithm [Crammer and Singer, 2003]

- Address all the constraints in the N-best list

$$\begin{aligned} \min_{\boldsymbol{w}} \quad & \left(\|\boldsymbol{w} - \boldsymbol{w}_{\text{old}}\|^2 + C\,\xi_t\right) \\ \text{s.t.} \quad & \boldsymbol{w}^\top \mathbf{f}(x_t, \boldsymbol{y}_t) - \boldsymbol{w}^\top \mathbf{f}(x_t, \boldsymbol{y}^{(i)}) \geqslant 1 - \xi_t \quad \text{for} \quad i = 1, \dots, N \end{aligned}$$

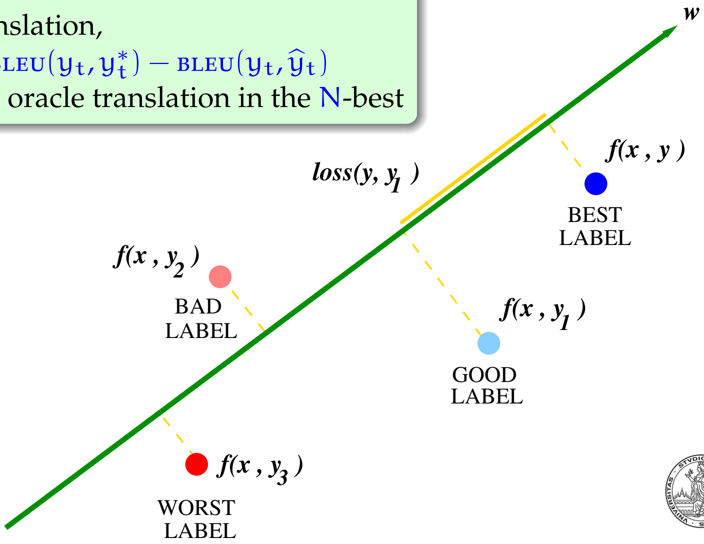
- Use a SVM solver on a pseudo training set with instances

$$\mathbf{f}(x_t, \boldsymbol{y}_t) - \mathbf{f}(x_t, \boldsymbol{y}^{(i)}) \qquad i = 1, \dots, N$$

Cost-sensitive margin-based learning

$$\boldsymbol{w}^\top \mathbf{f}(\mathbf{x}_t, \mathbf{y}_t) - \boldsymbol{w}^\top \mathbf{f}(\mathbf{x}_t, \widehat{\mathbf{y}}_t) \geqslant \text{loss}(y_t, \widehat{y}_t) - \xi_t$$

In machine translation,
$\text{loss}(y_t, \widehat{y}_t) = \text{BLEU}(y_t, y_t^*) - \text{BLEU}(y_t, \widehat{y}_t)$
where y_t^* is the oracle translation in the N-best



Using duality

Cost-sensitive passive-aggressive optimization at x_t

$$\begin{aligned} \min_{\boldsymbol{w}} \quad & \left(\|\boldsymbol{w} - \boldsymbol{w}_{\text{old}}\|^2 + C\,\xi_t\right) \\ \text{s.t.} \quad & \boldsymbol{w}^\top \mathbf{f}(\boldsymbol{x}_t, \boldsymbol{y}_t) - \boldsymbol{w}^\top \mathbf{f}(\boldsymbol{x}_t, \widehat{\boldsymbol{y}}_t) \geqslant \text{loss}(y_t, \widehat{y}_t) - \xi_t \end{aligned}$$

Learning rate is the dual-maximizing Lagrange multiplier

The dual Lagrange function $\mathcal{D}_t(\alpha)$ always satisfies

$$\underset{0 \leqslant \alpha \leqslant C}{\operatorname{argmax}}\, \mathcal{D}_t(\alpha) = \eta_t$$

Game-theoretical loss bounds [Shalev-Shwartz and Singer, 2007]

Theorem

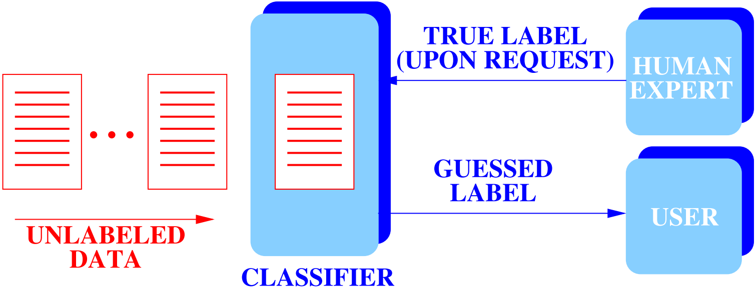
For *any* *sequence* $(\boldsymbol{x}_1, \boldsymbol{y}_1), \dots, (\boldsymbol{x}_T, \boldsymbol{y}_T)$

$$\sum_{t=1}^{T} \mathrm{loss}(y_t, \widehat{y}_t) \leqslant \sum_{t=1}^{T} \mathcal{D}_t(\eta_t) \leqslant \min_{\boldsymbol{w}} \mathcal{P}\Big(\boldsymbol{w},\ (\boldsymbol{x}_1, \boldsymbol{y}_1), \dots, (\boldsymbol{x}_T, \boldsymbol{y}_T)\Big)$$

- $\mathcal{P}$ is a convex SVM-style objective that trades-off $\|\boldsymbol{w}\|^2$ with the overall extent by which the linear margin constraints in the N-best lists are violated by $\boldsymbol{w}$
- Bound extends to any algorithm that optimizes over more constraints (e.g., MIRA)

Summary

1. The online protocol
2. Structured classification
3. Active learning
4. Multiview learning


TRUE LABEL
(UPON REQUEST)
HUMAN
EXPERT
GUESSED
LABEL
USER
UNLABELED
DATA
CLASSIFIER

Learning a binary classifier

Setup

- Data process $\quad x_1, x_2, \dots \in \mathbb{R}^d$
- Label process $\quad y_1, y_2, \dots \in \{-1, +1\}$

Assumptions

- Observing the data process is "cheap"
- Observing the label process is "expensive"
  $\rightarrow$ need to query the human expert

Learning a binary classifier

Setup

- Data process $\quad x_1, x_2, \dots \in \mathbb{R}^d$
- Label process $\quad y_1, y_2, \dots \in \{-1, +1\}$

Assumptions

- Observing the data process is "cheap"
- Observing the label process is "expensive"
  $\rightarrow$ need to query the human expert

Question

How much better can we do by subsampling adaptively the label process?

Related work

- Adaptive design in Statistics [Zacks, 2009]

- Uncertainty sampling [Cohn, Atlas and Ladner, 1990]

- Query by committee [Freund, Seung, Shamir and Tishby, 1997]
  Modified Perceptron [Dasgupta, Kalai and Monteleoni, 2005]
  $\longrightarrow$ exponential advantage in the noise-free case

- General strategies for the noisy case
  [Balcan, Beygelzimer and Langford, 2006]
  [Balcan, Broder and Zhang, 2007]
  [Dasgupta, Hsu and Monteleoni, 2008]

A parametric classifier

Regularized least squares (slightly modified)

- $\widehat{f}_t(x) = w^\top x$ where

$$w = \operatorname*{argmin}_{w \in \mathbb{R}^d} \left(\left\| S^\top w - y \right\|^2 + \|w\|^2 + \left(w^\top x\right)^2 \right)$$

- $S = \left[x_{t_1}, \dots, x_{t_N}\right]$ is matrix of queried data points
- $y = \left(y_{t_1}, \dots, y_{t_N}\right)$ is vector of queried labels

Remarks

- Involves only inner products
  $\rightarrow$ replaceable by kernel functions
- Time quadratic (in number of queries) for incremental update

Query strategy

[Cavallanti, C-B and Gentile, 2009]

- View $\widehat{f}_t(\boldsymbol{x})$ as a biased estimator of $f^*(\boldsymbol{x})$
- Use large deviation analysis to obtain confidence interval

$$\widehat{f}_t(\boldsymbol{x}_t) \pm c\sqrt{\frac{\ln t}{N_t}} \qquad \text{for confidence level } 1 - t^{-1}$$

- N_t is number of queries up to time t

Query strategy

[Cavallanti, C-B and Gentile, 2009]

- View $\widehat{f}_t(\boldsymbol{x})$ as a biased estimator of $f^*(\boldsymbol{x})$
- Use large deviation analysis to obtain confidence interval

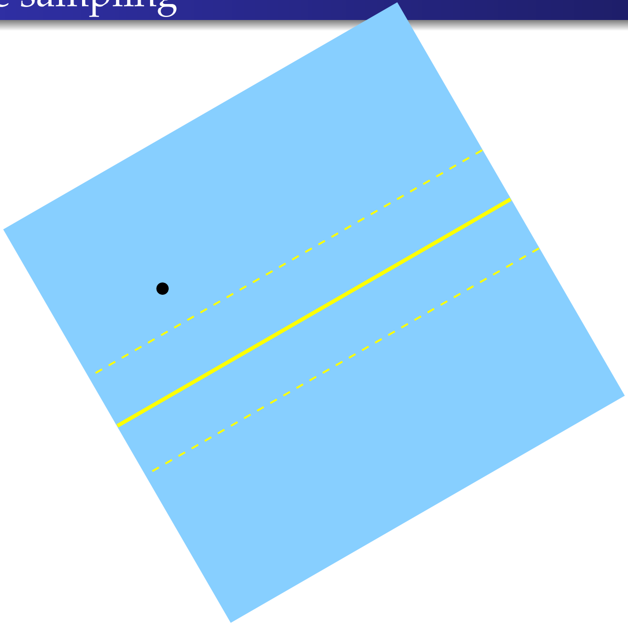
$$\widehat{f}_t(\boldsymbol{x}_t) \pm c\sqrt{\frac{\ln t}{N_t}} \qquad \text{for confidence level } 1 - t^{-1}$$

- N_t is number of queries up to time t

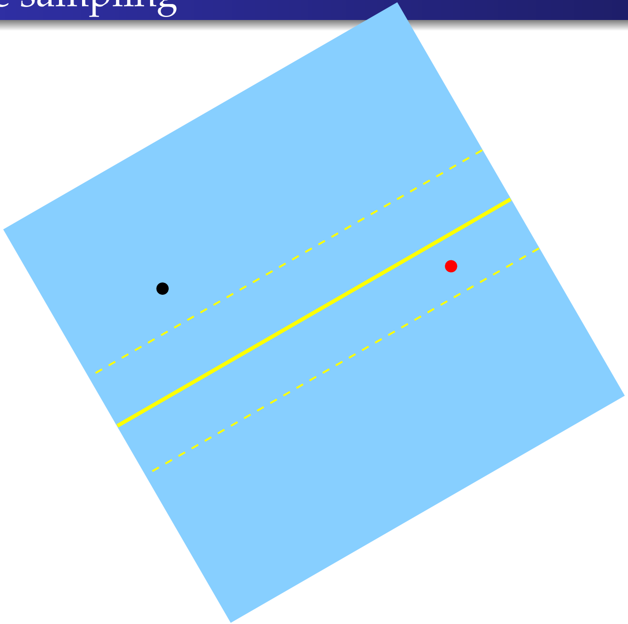
Query if *margin* of current point is small

$$\text{If} \quad \left|\widehat{f}_t(\boldsymbol{x}_t)\right| \leqslant c\sqrt{\frac{\ln t}{N}} \quad \text{then query } \boldsymbol{x}_t$$

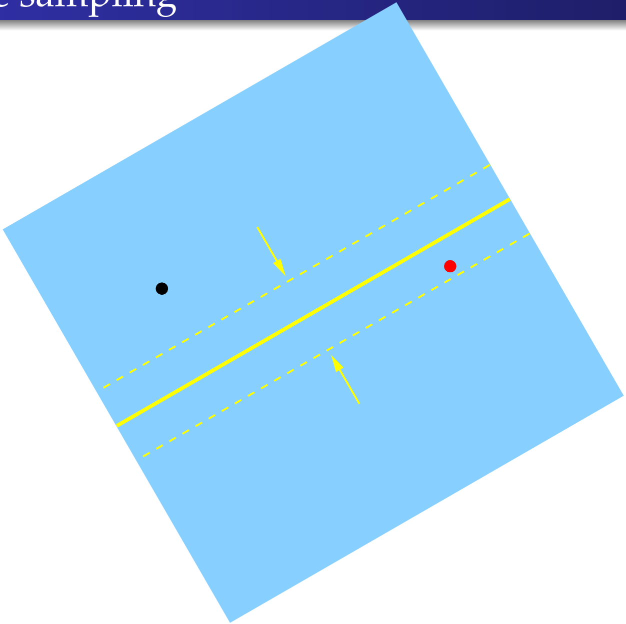
Selective sampling



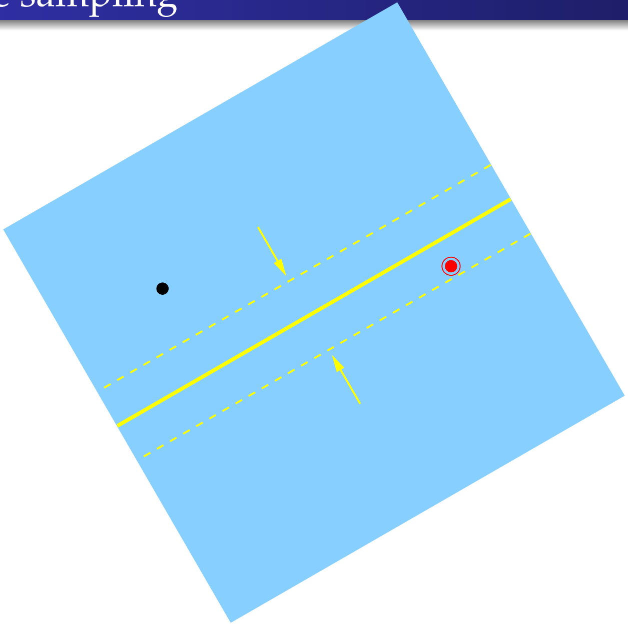
Selective sampling



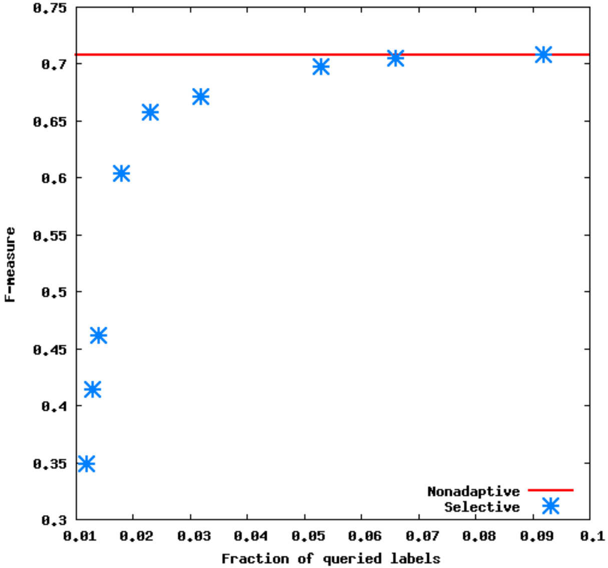
Selective sampling



Selective sampling



Results on text categorization



Summary

1. The online protocol
2. Structured classification
3. Active learning
4. Multiview learning

Online matrix learning

Some applications

- Learn a CD-HMM (emission densities are Gaussian mixtures) [Cheng, Sha and Saul, 2009]
- Learn the best subspace projection (online PCA) [Warmuth and Kuzmin, 2008]
- Multitask/multiview learning [Cavallanti, C-B and Gentile, 2008]

Multiview binary classification

Definitions

- Each data element (instance) is described by a set of tuples $\boldsymbol{x}^{(1)}, \dots, \boldsymbol{x}^{(K)} \in \mathbb{R}^d$ (views) each obtained from different measurements $\quad \rightarrow$ e.g., multimodal learning
- Build $d \times K$ matrix of views
- Run online linear classification algorithms in the linear space with inner product $\langle W, X \rangle = \operatorname{tr}\left(W^\top X\right)$
- Binary prediction $\textsc{sgn}\left(\langle W, X_t \rangle\right)$

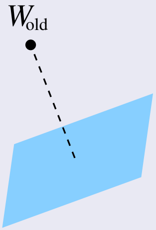
- How to do provably better than combining all views in a vector?
- Transfer information across views in order to reduce the learning complexity

Multiview binary classification

Margin optimization (binary case) for X_t

$$\min_{W} \quad \left(\|W - W_{\text{old}}\|^2 + C\,\xi_t\right)$$

$$\text{s.t.} \quad \underbrace{y_t \,\langle W, X_t \rangle \geqslant 1 - \xi_t}_{\text{linear constraint}}$$

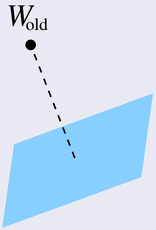


Multiview binary classification

Margin optimization (binary case) for X_t

$$\min_{W} \quad \left(\|W - W_{\text{old}}\|^2 + C\,\xi_t \right)$$

$$\text{s.t.} \quad \underbrace{y_t\ \langle W, X_t \rangle \geqslant 1 - \xi_t}_{\text{linear constraint}}$$



Spectral co-regularization

- $\|\cdot\| =$ Frobenius norm: columns of W updated independently
  $\rightarrow$ Perceptron/Passive-Aggressive on combined views
- $\|\cdot\| =$ any unitarialy invariant norm: columns may interact

Matrix p-norm Perceptron

- Schatten p-norm of W is the p-norm of the SVD vector
- Prediction is sign of $\quad (V^\top V)^{p-2} V^\top X$
- V is updated using the Perceptron rule $\quad V \leftarrow V + y\,X$

Matrix p-norm Perceptron

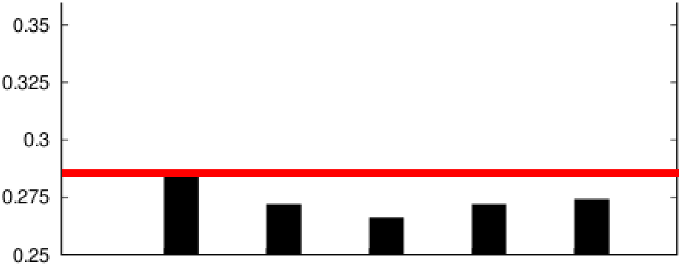
- Schatten p-norm of W is the p-norm of the SVD vector
- Prediction is sign of $\left(V^\top V\right)^{p-2} V^\top X$
- V is updated using the Perceptron rule $V \leftarrow V + y\,X$

Theory

1. If the best model W is simple $\mathrm{rank}(W) \ll K$
2. and views X_t are informative $\mathrm{rank}(X_t) \approx K$

$\rightarrow$ improve over combined views by a factor of K

Experimental results



- Gene function classification using six views
- Values $p = 2, 4, 6, 8, 10$
- Red line ($p = 2$) is error of Perceptron on combined views

THANK YOU