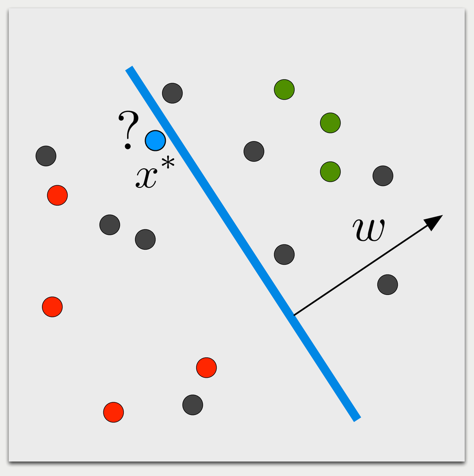
Near-optimal Batch Mode Active Learning and Stochastic Optimization

Yuxin Chen and Andreas Krause
Department of Computer Science, ETH Zurich, Switzerland
Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

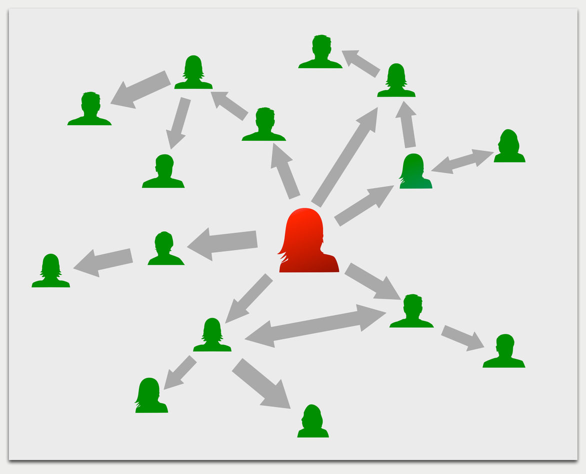
Contributions

- A general approach for information parallel learning and decision making
- Strong performance guarantees for a simple BatchGreedy algorithm
- Practical for batch-mode active learning and influence maximization
- Demonstrate the effectiveness of the algorithms for both applications

Adaptive Stochastic Minimum Cost Cover (Golovin & Krause '11)



[A]ctive learning



[I]nfluence maximization

A: f: # of hypotheses eliminated
$\mathbf{y}_\mathcal{V}$: labels of data points
P: hypotheses prior

I: f: # of nodes influenced
$\mathbf{y}_\mathcal{V}$: Influences of nodes
P: Joint distribution of $\mathbf{y}_\mathcal{V}$

$$\min_{\pi\in\Pi} \text{cost}(\pi) \text{ s.t. } f(\mathcal{S}(\pi,\mathbf{y}_\mathcal{V}))\geq Q \text{ for all } \mathbf{y}_\mathcal{V} \text{ with } P(\mathbf{y}_\mathcal{V})>0$$

The Min-Cost Cover subjective

Normalized We derive no utility from knowing nothing: $f(\emptyset)=0$

Monotonic Adding labels never hurts:

$$\mathcal{S}\subseteq\mathcal{S}'\subseteq\mathcal{V}\times O\Rightarrow f(\mathcal{S})\leq f(\mathcal{S}')$$

Submodular Adding a label helps more if we have observed less labels:

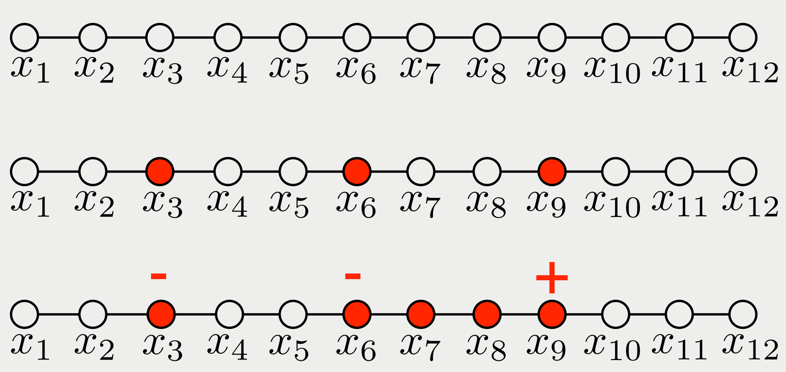
$$\left.\begin{array}{l}\mathcal{S}\subseteq\mathcal{S}'\subseteq\mathcal{V}\times O\\(j,y)\in\mathcal{V}\times O\setminus\mathcal{S}'\end{array}\right\}\Rightarrow f(\mathcal{S}\cup\{(j,y)\})-f(\mathcal{S})\geq f(\mathcal{S}'\cup\{(j,y)\})-f(\mathcal{S}')$$

Adaptive submodular The gain of an item, in expectation over its unknown label, can never increase as we gather more information:

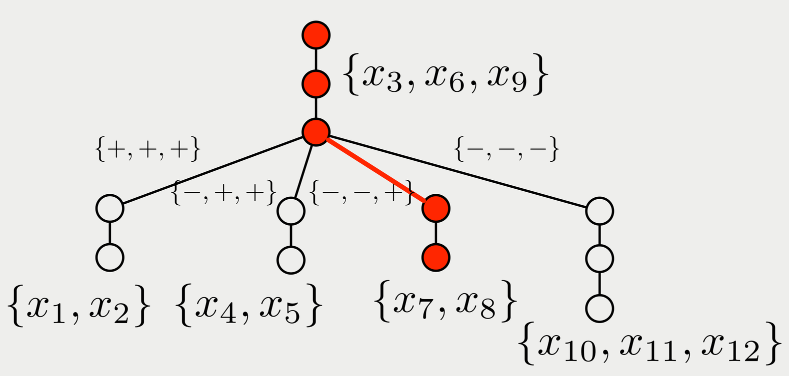
$$\mathcal{S}\subseteq\mathcal{S}'\subseteq\mathcal{V}\times O, P(\mathcal{S}')>0\Rightarrow\Delta_f(j\mid\mathcal{S})\geq\Delta_f(j\mid\mathcal{S}')$$

where $\Delta_f(j\,|\,\mathcal{S})=\sum_y P(Y_j=y\mid\mathcal{S})\Big[f(\mathcal{S}\cup\{(j,y)\})-f(\mathcal{S})\Big]$

Illustration of Batch Mode Active Learning (1-d, binary threshold)



(top) unlabeled data; (middle) 1st selected batch; (bottom) received labels & 2nd selected batch.



The decision tree representing the BatchGreedy policy

Near-optimal Batch Selection Algorithm (BatchGreedy)

Conditional marginal benefit of item s:

$$\Delta_f(s\mid\mathcal{A},\mathbf{y}_\mathcal{B})=\mathbb{E}_{\mathbf{y}_\mathcal{V}}\left[f(\mathbf{y}_{\{s\}\cup\mathcal{A}\cup\mathcal{B}})-f(\mathbf{y}_{\mathcal{A}\cup\mathcal{B}})\mid\mathbf{y}_\mathcal{B}\right].$$

The BatchGreedy policy will greedily select the i-th element in the j-th batch

$$s_{i,j}=\arg\max_{s\in\mathcal{V}}\Delta_f(s\mid\{s_{1,j},\dots,s_{i-1,j}\},\mathbf{y}_\mathcal{B})$$

where $\mathbf{y}_\mathcal{B}$ is the set of observations (labeled examples) from batches up to $j-1$

Comparing BatchGreedy with the Optimal Batch Policy

Let $\text{OPT}_{ac,k}$ be the expected cost and $\text{OPT}_{wc,k}$ be the worst-case cost of an optimal policy selecting batches of size k. Further let $\delta=\min_{\mathbf{y}_\mathcal{V}\in\text{supp}(P)}P(\mathbf{y}_\mathcal{V})$. Then for the cost of the policy π_G implementing BatchGreedy it holds that

$$\text{cost}(\pi_G)\leq\text{OPT}_k\left(\frac{e}{e-1}\right)\Big(\ln Q+1\Big),\text{ and}$$

$$\text{cost}_{wc}(\pi_G)\leq\text{OPT}_{wc,k}\left(\frac{e}{e-1}\right)\left(\ln\frac{Q}{\delta}+1\right).$$

Comparing BatchGreedy with the Optimal Sequential Policy

Fix $0<\beta<1$. Let OPT_{wc} be the worst-case cost of an optimal *sequential* policy π^*, constrained to picking a number of items which is a multiple of k. Further suppose that the variables $Y_1,\dots,Y_n$ are independent. Then for the cost of the policy π_G implementing BatchGreedy, run until it achieves $f(\pi_G)\geq Q-\beta$ it holds that

$$\text{cost}_{wc}(\pi_G)\leq\text{OPT}_{wc}\ \Big(e/(e-1)\Big)^2\left(\ln\frac{Q}{\beta}+1\right).$$

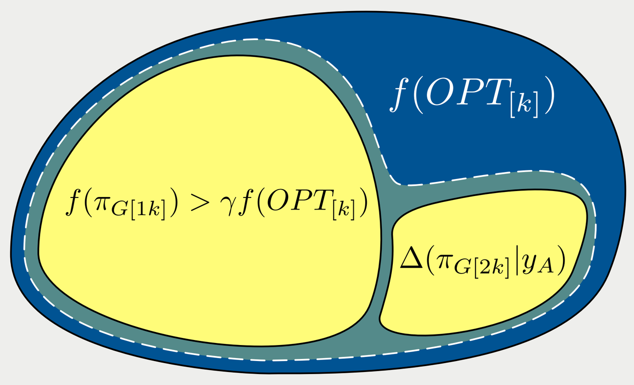
Moreover, it holds that $P\big(f(\mathcal{S}(\pi_G,\mathbf{y}_\mathcal{V}))\geq Q\big)\geq 1-\beta$.

BatchGreedy VS. the Optimal Sequential Policy

Single batch setting

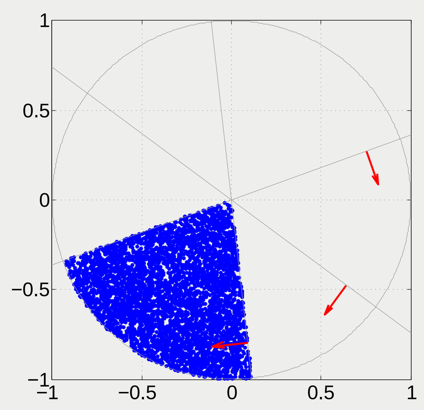
- The expected cost incurred by sequential policy to achieve Q can be *arbitrarily lower* than the cost of single-batch setting (Goemans and Vondrak '06).
- The utility of sequential algorithm (with cardinality cstr.) *cannot be much higher* than that of the best non-adaptive algorithm (Asadpour et al, '08).

Key insight We can string multiple batches together in an adaptive manner, in a way that the overall cost of achieving quota Q is not much higher than that of the fully sequential policy.

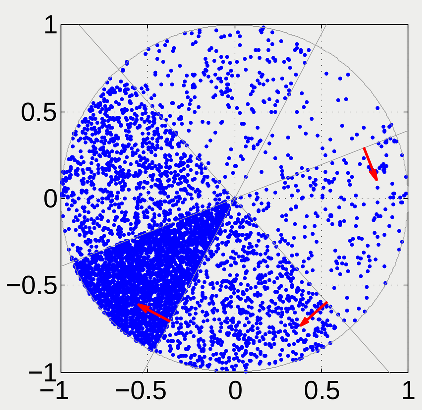


An Approximate Implementation of BatchGreedy for BMAL

Running time of BatchGreedy depends polynomially on $|\mathcal{H}|$.



Noiseless
(red arrows are constraints)



20% observations are noisy

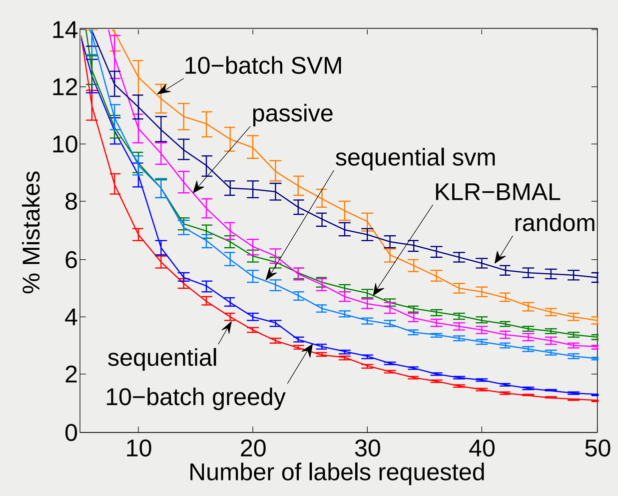
MCMC sampler Hit-and-run hypothesis sampler for linear separators and noisy observations

Assumption Hypotheses that violate more constraints induce lower confidence

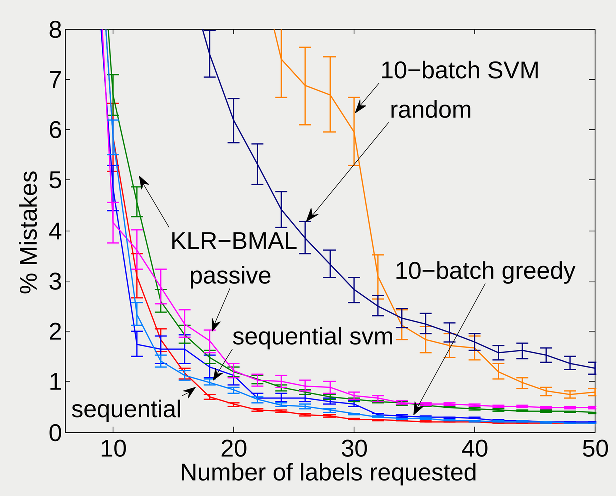
$$s\leftarrow\arg\min_{s'}\sum_{\ell=1}^{N}\Big[\hat{P}(\mathcal{H}(\{(\mathbf{x},h_\ell(\mathbf{x})):\mathbf{x}\in\mathcal{A}\cup\{s'\}\}))\Big]$$

Between batches Update $\hat{P}(\mathcal{H})=\frac{1}{N}\sum_{\ell=1}^{N}\delta_{h_\ell}$ based on current observations.

Pool-based Batch Mode Active Learning



WDBC

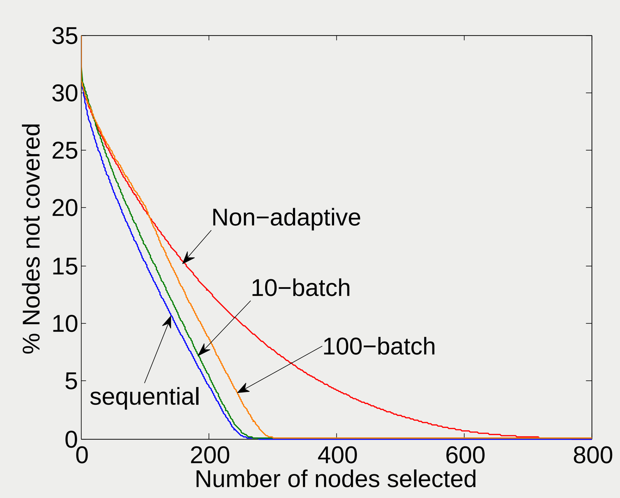


MNIST

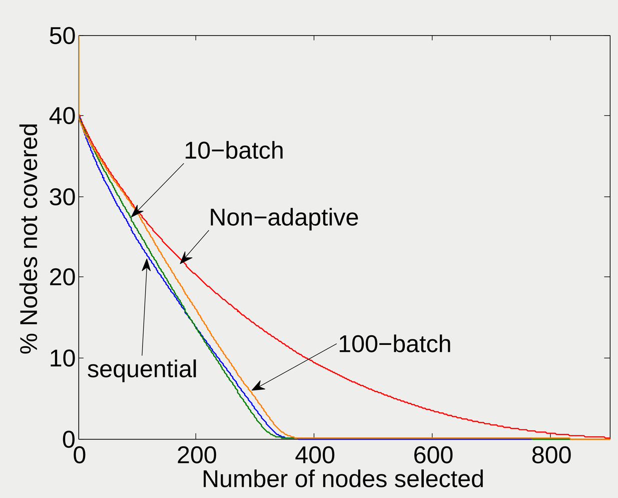
Applications

- Crowdsourcing annotation on Amazon Mechanical Turk
- High-throughput experimental design

Multi-stage Influence Maximization in Social Networks



Epinions



Slashdot

Applications

- Multi-stage marketing campaign
- Resource allocation in computational sustainability
- Vaccination problems in epidemiology