Budgeted stream-based active learning via adaptive submodular maximization

Kaito Fujii (UTokyo)

(joint work with Hisashi Kashima (KyotoU))

IBIS Workshop

2017.11.8

Table of Contents

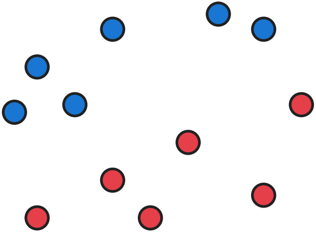
Supervised Classification

Input

A set of labeled instances $\{(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}\}_{i=1,\cdots,n}$

Output

A classifier $\hat{f} : \mathcal{X} \to \mathcal{Y}$



$\mathcal{X} = \mathbb{R}^2$,

$\mathcal{Y} = \{$red, blue$\}$

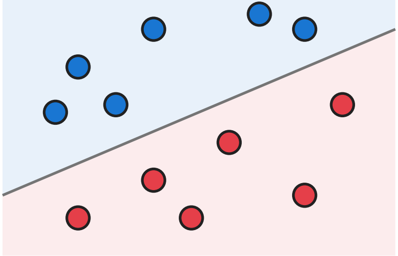
Supervised Classification

Input

A set of labeled instances $\{(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}\}_{i=1,\cdots,n}$

Output

A classifier $\hat{f} : \mathcal{X} \to \mathcal{Y}$



$\mathcal{X} = \mathbb{R}^2$,
$\mathcal{Y} = \{$red, blue$\}$

Motivation for Active Learning

In some real world scenarios,

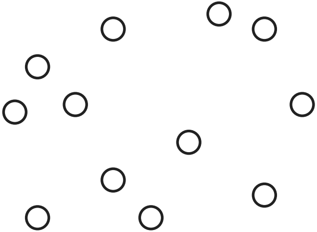
- there are a lot of unlabeled instances, but
- labeling needs a large cost (money or time).

Active Learning

The learner selects which instances to label and can reduce the labeling cost.

Pool-based Active Learning

All unlabeled instances are given in advance



Unlabeled instances

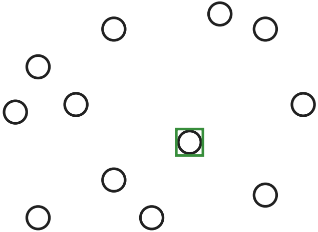
$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

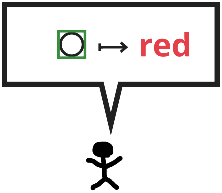
Labeling oracle

$\phi : V \to \mathcal{Y}$

Pool-based Active Learning

All unlabeled instances are given in advance





Unlabeled instances

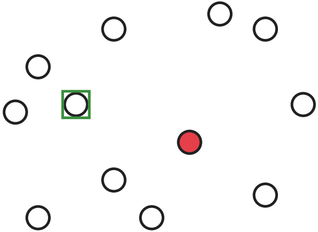
$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

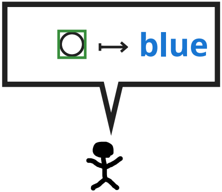
Labeling oracle

$\phi : V \to \mathcal{Y}$

Pool-based Active Learning

All unlabeled instances are given in advance





Unlabeled instances

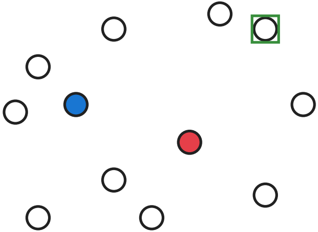
$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

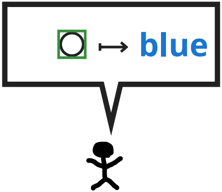
Labeling oracle

$\phi : V \to \mathcal{Y}$

Pool-based Active Learning

All unlabeled instances are given in advance





Unlabeled instances

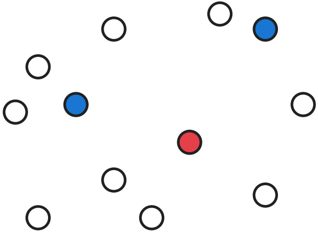
$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

Labeling oracle

$\phi : V \to \mathcal{Y}$

Pool-based Active Learning

All unlabeled instances are given in advance



Unlabeled instances

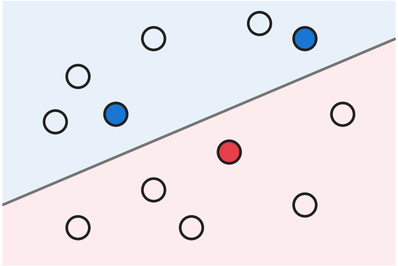
$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

Labeling oracle

$\phi : V \rightarrow \mathcal{Y}$

Pool-based Active Learning

All unlabeled instances are given in advance



Unlabeled instances

$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

Labeling oracle

$\phi : V \to \mathcal{Y}$

Stream-based Active Learning

Unlabeled instances arrive sequentially

We consider the case of selecting k instances out of n

(n, k known in advance)

Unlabeled instances

$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

Labeling oracle

$\phi : V \rightarrow \mathcal{Y}$

Stream-based Active Learning

Unlabeled instances arrive sequentially

We consider the case of selecting k instances out of n

(n, k known in advance)

Unlabeled instances

$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

Labeling oracle

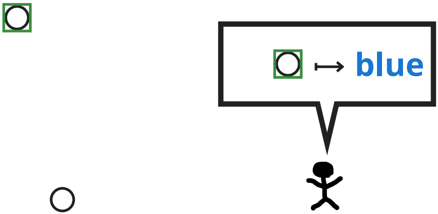
$\phi : V \to \mathcal{Y}$

Stream-based Active Learning

Unlabeled instances arrive sequentially

We consider the case of selecting k instances out of n

(n, k known in advance)



Unlabeled instances

$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

Labeling oracle

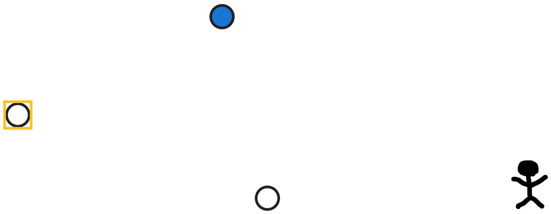
$\phi : V \to \mathcal{Y}$

Stream-based Active Learning

Unlabeled instances arrive sequentially

We consider the case of selecting k instances out of n

(n, k known in advance)



Unlabeled instances

$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

Labeling oracle

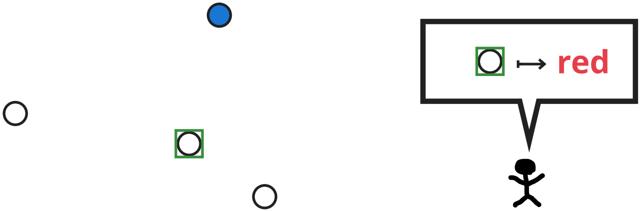
$\phi : V \to \mathcal{Y}$

Stream-based Active Learning

Unlabeled instances arrive sequentially

We consider the case of selecting k instances out of n

(n, k known in advance)



Unlabeled instances

$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

Labeling oracle

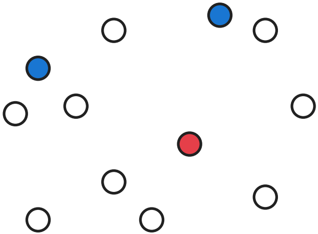
$\phi : V \rightarrow \mathcal{Y}$

Stream-based Active Learning

Unlabeled instances arrive sequentially

We consider the case of selecting k instances out of n

(n, k known in advance)



Unlabeled instances

$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

Labeling oracle

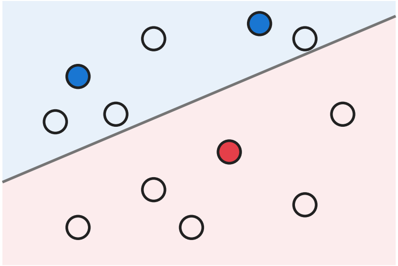
$\phi : V \to \mathcal{Y}$

Stream-based Active Learning

Unlabeled instances arrive sequentially

We consider the case of selecting k instances out of n

(n, k known in advance)



Unlabeled instances

$V = \{x_i\}_{i=1,\cdots,n} \subset \mathcal{X}$

Labeling oracle

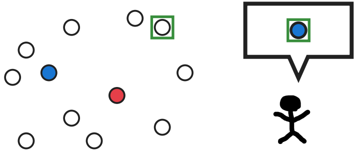
$\phi : V \to \mathcal{Y}$

Overview

A new framework for stream-based active learning

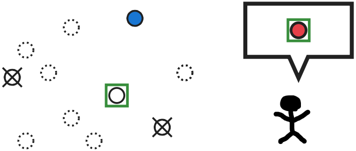
Pool-based active learning

All unlabeled instances are given in advance



Stream-based active learning

Unlabeled instances appear one by one



Adaptive submodular maximization [Golovin–Krause'11] → Proposed framework

Submodular secretary problem [Bateni–Hajiaghayi–Zadimoghaddam'13] → Proposed framework

Table of Contents

Submodular Maximization

Selection of a "good" subset of given finite set V

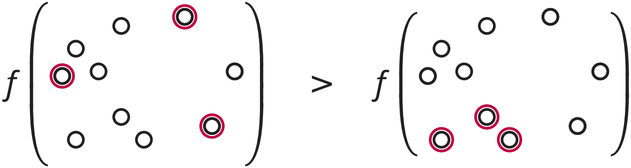
Maximize $f(S)$

subject to $|S| \leq k$

$f : 2^V \to \mathbb{R}$
submodular

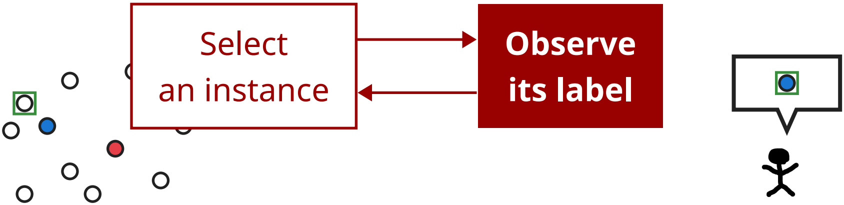
Data Summarization [Badanidiyuru+'14]

Select a small summary for given large dataset V



Adaptive Submodular Maximization

[Golovin–Krause'11]



The learner can select the next instance to label according to the labels observed so far

Adaptive Submodularity

An extention of submodularity to this adaptive setting

Table of Contents

Classical Secretary Problem [folklore'60s]

Problem

n candidates arrive **in random order** (n is given),
and decide whether to hire at each arrival

Classical Secretary Algorithm

pass the first $\lfloor n/\mathrm{e} \rfloor$ ones, and after that,
if the coming one is the best so far, hire him
→ the best one can be hired with prob. $\geq 1/\mathrm{e}$

Classical Secretary Problem [folklore'60s]

Problem

n candidates arrive **in random order** (n is given),
and decide whether to hire at each arrival

Classical Secretary Algorithm

pass the first $\lfloor n/e \rfloor$ ones, and after that,
if the coming one is the best so far, hire him
$\longrightarrow$ the best one can be hired with prob. $\geq 1/e$

$n = 10$

3

Classical Secretary Problem [folklore'60s]

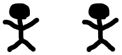
Problem

n candidates arrive **in random order** (n is given),
and decide whether to hire at each arrival

Classical Secretary Algorithm

pass the first $\lfloor n/e \rfloor$ ones, and after that,
if the coming one is the best so far, hire him
→ the best one can be hired with prob. $\geq 1/e$

$n = 10$



3 4

Classical Secretary Problem [folklore'60s]

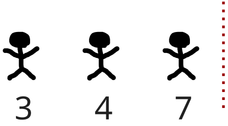
Problem

n candidates arrive **in random order** (n is given),
and decide whether to hire at each arrival

Classical Secretary Algorithm

pass the first $\lfloor n/e \rfloor$ ones, and after that,
if the coming one is the best so far, hire him
→ the best one can be hired with prob. $\geq 1/e$

$n = 10$



Classical Secretary Problem [folklore'60s]

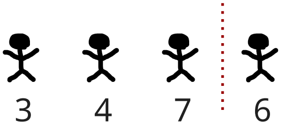
Problem

n candidates arrive **in random order** (n is given),
and decide whether to hire at each arrival

Classical Secretary Algorithm

pass the first $\lfloor n/e \rfloor$ ones, and after that,
if the coming one is the best so far, hire him
➡ the best one can be hired with prob. $\geq 1/e$

$n = 10$



Classical Secretary Problem [folklore'60s]

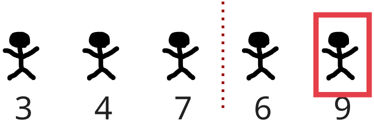
Problem

n candidates arrive **in random order** (n is given),
and decide whether to hire at each arrival

Classical Secretary Algorithm

pass the first $\lfloor n/e \rfloor$ ones, and after that,
if the coming one is the best so far, hire him
→ the best one can be hired with prob. $\geq 1/e$

$n = 10$



Classical Secretary Problem [folklore'60s]

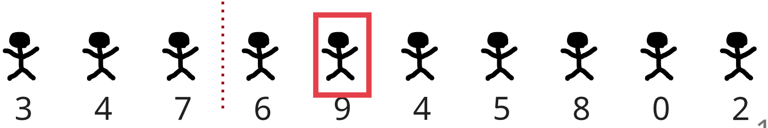
Problem

n candidates arrive **in random order** (n is given),
and decide whether to hire at each arrival

Classical Secretary Algorithm

pass the first $\lfloor n/e \rfloor$ ones, and after that,
if the coming one is the best so far, hire him
$\Rightarrow$ the best one can be hired with prob. $\geq 1/e$

$n = 10$



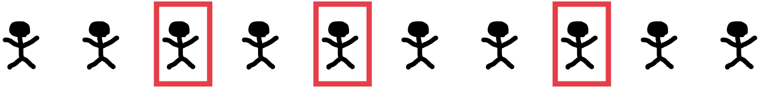
Submodular Secretary Problem

[Bateni–Hajiaghayi–Zadimoghaddam'13]

A generalization of the classical secretary problem

1. multiple candidates can be selected
2. the objective function $f : 2^V \to \mathbb{R}_{\geq 0}$ is submodular

$n = 10, k = 3$



Competitive Ratio

The competitive ratio of an algorithm is $\alpha \in [0, 1]$.

$\Updownarrow$ **def**

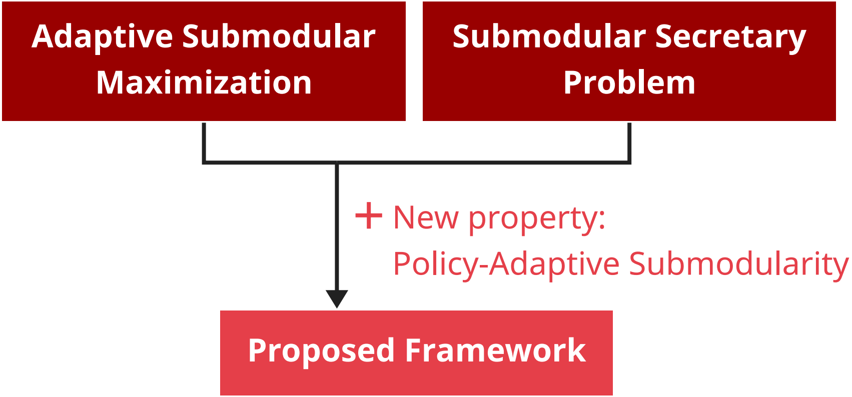
For any problem instance, the output S satisfies:

$$\mathbb{E}[f(S)] \geq \alpha \underbrace{\max_{S^* \subseteq V} f(S^*)}_{\text{the optimal achieved by the clairvoyant}}$$

Table of Contents

Proposed Framework

The proposed framework is a combination of previous frameworks, but it is not straightforward



Policy-Adaptive Submodularity

Policy-adaptive submodularity is also a natural extension of submodularity to the adaptive setting

Submodularity

Diminishing return of each instance

to the adaptive setting

Adaptive Submodulairty

Policy-Adaptive Submodularity

Policy-adaptive submodularity is also a natural extension of submodularity to the adaptive setting

Submodularity

Diminishing return of each instance

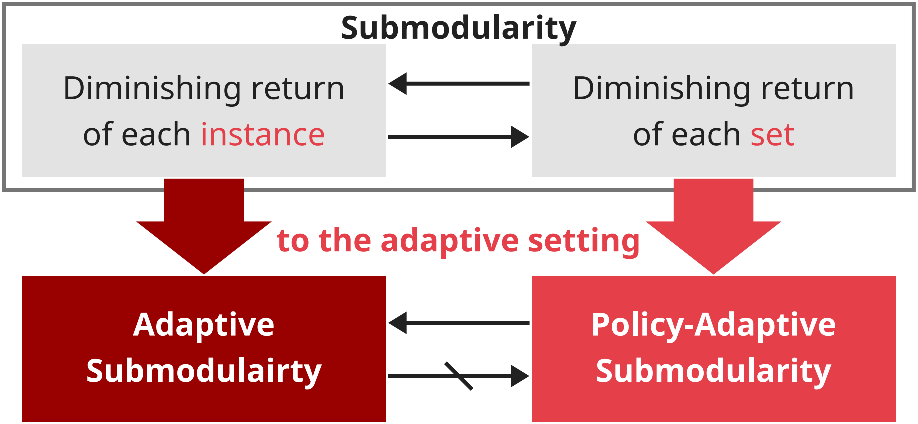
Diminishing return of each set

to the adaptive setting

Adaptive Submodulairty

Policy-Adaptive Submodularity

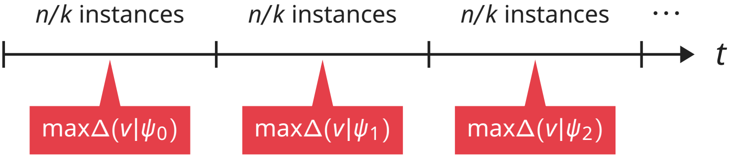
Policy-adaptive submodularity is also a natural extension of submodularity to the adaptive setting



Adaptive Stream Algorithm

Stream setting | A limited memory can be used

Partition the whole stream into k segments,
and select the “best” instance from each segment



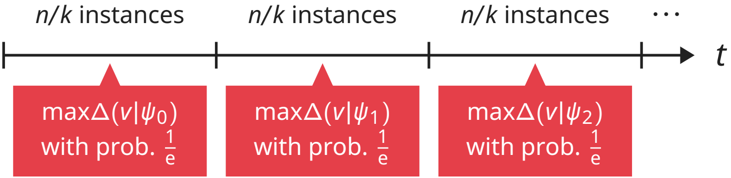
Theorem **[Fujii–Kashima’16]**

The competitive ratio is $(2-\sqrt{3})(1-1/e) \approx 0.16$

Adaptive Secretary Algorithm

Secretary setting	immediate decision at each arrival

Apply the classical secretary algo. to each segment, and select the "best" instance with probability $1/e$



Theorem [Fujii–Kashima'16]

The competitive ratio is $\dfrac{1-1/e}{2e\sqrt{1+2/e}} \approx 0.08$

Table of Contents

Experimental Settings

Datasets

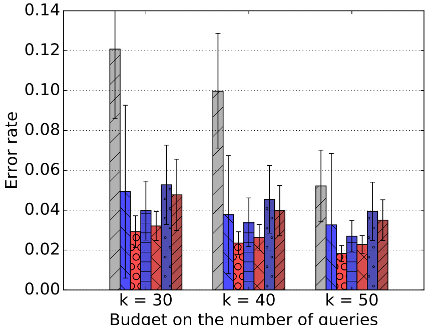
- WDBC ($n = 596$，32-d)
- MNIST ($n = 14780$，reduced to 10-d by PCA)

Benchmarks

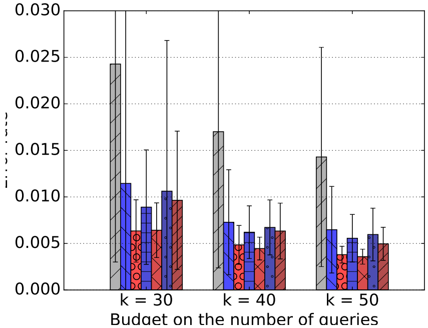
- Uncertainty sampling
- Random

The proposed method is based on ALuMA [Gonen+13]

Experimental Results

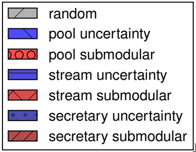


WDBC



MNIST

The proposed method outperforms uncertainty sampling in each setting

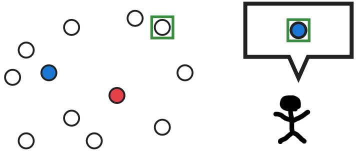


Overview

A new framework for stream-based active learning

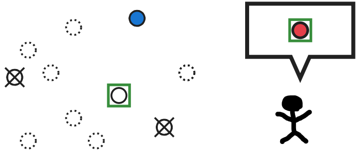
Pool-based active learning

All unlabeled instances are given in advance



Stream-based active learning

Unlabeled instances appear one by one



Adaptive submodular maximization
[Golovin–Krause’11]

Proposed framework

Submodular secretary problem
[Bateni–Hajiaghayi–Zadimoghaddam’13]