Learnability of Co-Enumerable Classes

LATA 2012

Frank Stephan[1] Gao Ziyuan[2]

[1]Department of Mathematics and Department of Computer Science
National University of Singapore

[2]Department of Mathematics
National University of Singapore

March 8, 2012

1. Introduction
   - Outline of Inductive Inference
   - Examples
2. Goals of the Project
   - Objective 1
   - Objective 2

General Idea of Inductive Inference

- The task is to correctly infer a concept based on finitely but possibly arbitrarily many examples generated by the rule defining the concept.

General Idea of Inductive Inference

- The task is to correctly infer a concept based on finitely but possibly arbitrarily many examples generated by the rule defining the concept.

General Idea of Inductive Inference

- The task is to correctly infer a concept based on finitely but possibly arbitrarily many examples generated by the rule defining the concept.

Data

Rule

General Idea of Inductive Inference

- The task is to correctly infer a concept based on finitely but possibly arbitrarily many examples generated by the rule defining the concept.

Data	Rule
$2, 4, 6, 8, 10, 12, \ldots$	Even numbers

General Idea of Inductive Inference

- The task is to correctly infer a concept based on finitely but possibly arbitrarily many examples generated by the rule defining the concept.

Data	Rule
$2, 4, 6, 8, 10, 12, \ldots$	Even numbers
$1, 1, 2, 3, 5, 8, \ldots$	Fibonacci numbers

General Idea of Inductive Inference

- The task is to correctly infer a concept based on finitely but possibly arbitrarily many examples generated by the rule defining the concept.

Data	Rule
$2, 4, 6, 8, 10, 12, \ldots$	Even numbers
$1, 1, 2, 3, 5, 8, \ldots$	Fibonacci numbers
$2, 1, 3, 4, 7, 11, 18, 29, 47, 76, 123, \ldots$	Lucas numbers

General Idea of Inductive Inference

- The task is to correctly infer a concept based on finitely but possibly arbitrarily many examples generated by the rule defining the concept.

Data	Rule
$2, 4, 6, 8, 10, 12, \ldots$	Even numbers
$1, 1, 2, 3, 5, 8, \ldots$	Fibonacci numbers
$2, 1, 3, 4, 7, 11, 18, 29, 47, 76, 123, \ldots$	Lucas numbers

- Let $\mathcal{C}$ be a class of concepts. Is there a function F mapping finite sequences of concept examples to members of $\mathcal{C}$ such that, if $a_0, a_1, a_2, \ldots$ is any infinite sequence of examples generated by a fixed member $\mathcal{L}$ of $\mathcal{C}$, then $F(a_0, a_1, \ldots, a_n) = \mathcal{L}$ for almost all n?

Formalising the Model Via Computability

- A learning machine M is modelled as a recursive function mapping all finite sequences in $\mathbb{N} \cup \{\#\}$ into $\mathbb{N}$.

Formalising the Model Via Computability

- A learning machine M is modelled as a recursive function mapping all finite sequences in $\mathbb{N} \cup \{\#\}$ into $\mathbb{N}$.
  - A *concept* is modelled as a set of natural numbers; each number in this set encodes a string example of the concept in question.

Formalising the Model Via Computability

- A learning machine M is modelled as a recursive function mapping all finite sequences in $\mathbb{N} \cup \{\#\}$ into $\mathbb{N}$.
  - A *concept* is modelled as a set of natural numbers; each number in this set encodes a string example of the concept in question.
  - $\mathcal{C}$ shall denote a class of concepts; that is, $\mathcal{C}$ is a class of sets of natural numbers.

Formalising the Model Via Computability

- A learning machine M is modelled as a recursive function mapping all finite sequences in $\mathbb{N} \cup \{\#\}$ into $\mathbb{N}$.
  - A *concept* is modelled as a set of natural numbers; each number in this set encodes a string example of the concept in question.
  - $\mathcal{C}$ shall denote a class of concepts; that is, $\mathcal{C}$ is a class of sets of natural numbers.
  - In this project, it is generally assumed that the set of natural numbers representing a concept is either recursively enumerable (r.e.) or co-recursively enumerable (co-r.e.).

Formalising the Model Via Computability

- Let $\mathcal{C}$ be a class of co-r.e. sets.

Formalising the Model Via Computability

- Let $\mathcal{C}$ be a class of co-r.e. sets.
- Fix a universal numbering

$$\overline{W}_0, \overline{W}_1, \overline{W}_2, \ldots.$$

of all co-r.e. sets. Then $\mathcal{C} \subseteq \{\overline{W}_0, \overline{W}_1, \overline{W}_2, \ldots\}$.

Formalising the Model Via Computability

- Let $\mathcal{C}$ be a class of co-r.e. sets.
- Fix a universal numbering

$$\overline{W}_0, \overline{W}_1, \overline{W}_2, \ldots.$$

of all co-r.e. sets. Then $\mathcal{C} \subseteq \{\overline{W}_0, \overline{W}_1, \overline{W}_2, \ldots\}$.

- We shall take $\{\overline{W}_0, \overline{W}_1, \overline{W}_2, \ldots\}$ as the default hypothesis space.

Formalising the Model Via Computability

- Let $\mathcal{C}$ be a class of co-r.e. sets.
- Fix a universal numbering

$$\overline{W}_0, \overline{W}_1, \overline{W}_2, \ldots.$$

  of all co-r.e. sets. Then $\mathcal{C} \subseteq \{\overline{W}_0, \overline{W}_1, \overline{W}_2, \ldots\}$.
- We shall take $\{\overline{W}_0, \overline{W}_1, \overline{W}_2, \ldots\}$ as the default hypothesis space.
- So our learner M specifies its conjecture $\mathcal{L}$ by outputting an index of a set in $\{\overline{W}_0, \overline{W}_1, \overline{W}_2, \ldots\}$ equal to $\mathcal{L}$.

Formalising the Model Via Computability

- Let $\mathcal{L}$ be a member of $\mathcal{C}$. A *text* $T_{\mathcal{L}}$ for $\mathcal{L}$ is an infinite sequence in $\mathcal{L} \cup \{\#\}$ that contains every element of $\mathcal{L}$. M is fed piecewise with prefixes of $T_{\mathcal{L}}$.

Formalising the Model Via Computability

- Let $\mathcal{L}$ be a member of $\mathcal{C}$. A *text* $T_{\mathcal{L}}$ for $\mathcal{L}$ is an infinite sequence in $\mathcal{L} \cup \{\#\}$ that contains every element of $\mathcal{L}$. M is fed piecewise with prefixes of $T_{\mathcal{L}}$.
- A possible text for the set of Fibonacci numbers could start with

  $$3, 8, 5, 13, \#, 2, 1, 3, 5, \#, \ldots$$

  The text may be arranged in any order; elements of the text may be repeated infinitely often, and $\#$ is a pause symbol denoting that no new data is available at the present stage.

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture
3	1

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture
3	1
$3, 8$	5

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture
3	1
3, 8	5
3, 8, 5	8

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture
3	1
$3, 8$	5
$3, 8, 5$	8
$3, 8, 5, 13$	0

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture
3	1
$3, 8$	5
$3, 8, 5$	8
$3, 8, 5, 13$	0
$3, 8, 5, 13, \#$	0

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture
3	1
$3, 8$	5
$3, 8, 5$	8
$3, 8, 5, 13$	0
$3, 8, 5, 13, \#$	0
$3, 8, 5, 13, \#, 2$	1

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture
3	1
$3, 8$	5
$3, 8, 5$	8
$3, 8, 5, 13$	0
$3, 8, 5, 13, \#$	0
$3, 8, 5, 13, \#, 2$	1
$3, 8, 5, 13, \#, 2, 1$	0

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture
3	1
$3, 8$	5
$3, 8, 5$	8
$3, 8, 5, 13$	0
$3, 8, 5, 13, \#$	0
$3, 8, 5, 13, \#, 2$	1
$3, 8, 5, 13, \#, 2, 1$	0
$3, 8, 5, 13, \#, 2, 1, 3$	0

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture
3	1
$3, 8$	5
$3, 8, 5$	8
$3, 8, 5, 13$	0
$3, 8, 5, 13, \#$	0
$3, 8, 5, 13, \#, 2$	1
$3, 8, 5, 13, \#, 2, 1$	0
$3, 8, 5, 13, \#, 2, 1, 3$	0
$3, 8, 5, 13, \#, 2, 1, 3, 5$	0

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture
3	1
$3, 8$	5
$3, 8, 5$	8
$3, 8, 5, 13$	0
$3, 8, 5, 13, \#$	0
$3, 8, 5, 13, \#, 2$	1
$3, 8, 5, 13, \#, 2, 1$	0
$3, 8, 5, 13, \#, 2, 1, 3$	0
$3, 8, 5, 13, \#, 2, 1, 3, 5$	0
$3, 8, 5, 13, \#, 2, 1, 3, 5, \#$	0

Formalising the Model Via Computability

- When M is fed with a prefix σ of a text $T_{\mathcal{L}}$ for some $\mathcal{L}$ in $\mathcal{C}$, it must guess a co-r.e. index for the set generating the data seen.
- Suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture
3	1
$3, 8$	5
$3, 8, 5$	8
$3, 8, 5, 13$	0
$3, 8, 5, 13, \#$	0
$3, 8, 5, 13, \#, 2$	1
$3, 8, 5, 13, \#, 2, 1$	0
$3, 8, 5, 13, \#, 2, 1, 3$	0
$3, 8, 5, 13, \#, 2, 1, 3, 5$	0
$3, 8, 5, 13, \#, 2, 1, 3, 5, \#$	0

Formalising the Model Via Computability

- Two general types of success criteria:

Formalising the Model Via Computability

- Two general types of success criteria:
  1. **Explanatory learning ($Ex_{co-r.e.}$) [Gold]**: M is said to $Ex_{co-r.e.}$ learn $\mathcal{C}$ if, for each $\mathcal{L}$ in $\mathcal{C}$, and any corresponding text $T_{\mathcal{L}}$ for $\mathcal{L}$, M outputs only finitely many new hypotheses, and the last one is correct.

Formalising the Model Via Computability

- Two general types of success criteria:
  1. **Explanatory learning ($Ex_{co-r.e.}$) [Gold]**: M is said to $Ex_{co-r.e.}$ learn $\mathcal{C}$ if, for each $\mathcal{L}$ in $\mathcal{C}$, and any corresponding text $T_{\mathcal{L}}$ for $\mathcal{L}$, M outputs only finitely many new hypotheses, and the last one is correct.
  2. **Behaviourally Correct learning ($BC_{co-r.e.}$) [Case, Lynes]**: M is said to $BC_{co-r.e.}$ learn $\mathcal{C}$ if, for each $\mathcal{L}$ in $\mathcal{C}$, and any corresponding text $T_{\mathcal{L}}$ for $\mathcal{L}$, M always correctly conjectures a set equal to $\mathcal{L}$ when it has seen a sufficiently long prefix of $T_{\mathcal{L}}$.

Formalising the Model Via Computability

- Two general types of success criteria:
  1. **Explanatory learning ($Ex_{co-r.e.}$) [Gold]**: M is said to $Ex_{co-r.e.}$ learn $\mathcal{C}$ if, for each $\mathcal{L}$ in $\mathcal{C}$, and any corresponding text $T_{\mathcal{L}}$ for $\mathcal{L}$, M outputs only finitely many new hypotheses, and the last one is correct.
  2. **Behaviourally Correct learning ($BC_{co-r.e.}$) [Case, Lynes]**: M is said to $BC_{co-r.e.}$ learn $\mathcal{C}$ if, for each $\mathcal{L}$ in $\mathcal{C}$, and any corresponding text $T_{\mathcal{L}}$ for $\mathcal{L}$, M always correctly conjectures a set equal to $\mathcal{L}$ when it has seen a sufficiently long prefix of $T_{\mathcal{L}}$.

    - M need not converge to a fixed index for $BC_{co-r.e.}$ learning.

Examples

Example [Gold]

The (infinite) class of all finite sets of numbers is $Ex_{co-r.e.}$ learnable.

Example

Every finite class S of co-recursively enumerable sets is $Ex_{co-r.e.}$ learnable.

Example [Gold]

The infinite class of all finite sets of numbers, together with the set $\mathbb{N}$, is not $BC_{co-r.e.}$ learnable.

Examples

Example: Pattern Languages [Angluin]

Let Σ be a finite alphabet containing at least two symbols, and $X = \{x_1, x_2, \ldots\}$ be a countable set of symbols disjoint from Σ. A pattern is any finite string over $\Sigma \cup X$. For any pattern p, the language of p, denoted by L(p), is defined as the class of all strings obtained by substituting strings in $\Sigma \cup X$ for variables occurring in p.

Examples

Example: Pattern Languages [Angluin]

Let Σ be a finite alphabet containing at least two symbols, and $X = \{x_1, x_2, \ldots\}$ be a countable set of symbols disjoint from Σ. A pattern is any finite string over $\Sigma \cup X$. For any pattern p, the language of p, denoted by $L(p)$, is defined as the class of all strings obtained by substituting strings in $\Sigma \cup X$ for variables occurring in p.

- Let $\Sigma = \{0, 1, 2, \ldots, 9\}$, and $p = 8x_1x_28x_5$.

Examples

Example: Pattern Languages [Angluin]

Let Σ be a finite alphabet containing at least two symbols, and $X = \{x_1, x_2, \ldots\}$ be a countable set of symbols disjoint from Σ. A pattern is any finite string over $\Sigma \cup X$. For any pattern p, the language of p, denoted by $L(p)$, is defined as the class of all strings obtained by substituting strings in $\Sigma \cup X$ for variables occurring in p.

- Let $\Sigma = \{0, 1, 2, \ldots, 9\}$, and $p = 8x_1x_28x_5$.
- $L(p)$ contains the elements $80081, 83382, 81185, \ldots$.

Examples

Example: Pattern Languages [Angluin]

Let Σ be a finite alphabet containing at least two symbols, and $X = \{x_1, x_2, \ldots\}$ be a countable set of symbols disjoint from Σ. A pattern is any finite string over $\Sigma \cup X$. For any pattern p, the language of p, denoted by $L(p)$, is defined as the class of all strings obtained by substituting strings in $\Sigma \cup X$ for variables occurring in p.

- Let $\Sigma = \{0, 1, 2, \ldots, 9\}$, and $p = 8x_1x_28x_5$.
- $L(p)$ contains the elements $80081, 83382, 81185, \ldots$.
- Consider an effective enumeration of the pattern languages:

$$L(p_1), L(p_2), L(p_3), \ldots$$

Examples

Example: Pattern Languages [Angluin]

Let Σ be a finite alphabet containing at least two symbols, and $X = \{x_1, x_2, \ldots\}$ be a countable set of symbols disjoint from Σ. A pattern is any finite string over $\Sigma \cup X$. For any pattern p, the language of p, denoted by $L(p)$, is defined as the class of all strings obtained by substituting strings in $\Sigma \cup X$ for variables occurring in p.

- Let $\Sigma = \{0, 1, 2, \ldots, 9\}$, and $p = 8x_1x_28x_5$.
- $L(p)$ contains the elements $80081, 83382, 81185, \ldots$.
- Consider an effective enumeration of the pattern languages:

$$L(p_1), L(p_2), L(p_3), \ldots$$

Examples

Example: Pattern Languages [Angluin]

Let Σ be a finite alphabet containing at least two symbols, and $X = \{x_1, x_2, \ldots\}$ be a countable set of symbols disjoint from Σ. A pattern is any finite string over $\Sigma \cup X$. For any pattern p, the language of p, denoted by $L(p)$, is defined as the class of all strings obtained by substituting strings in $\Sigma \cup X$ for variables occurring in p.

- Let $\Sigma = \{0, 1, 2, \ldots, 9\}$, and $p = 8x_1x_28x_5$.
- $L(p)$ contains the elements $80081, 83382, 81185, \ldots$.
- Consider an effective enumeration of the pattern languages:

$$L(p_1), L(p_2), L(p_3), \ldots$$

The class of all pattern languages is $Ex_{r.e.}$ learnable with respect to this indexing.

Goals of the Project

Objective 1

- Try to structurally characterise classes that are co-r.e. learnable according to specific learning criteria.

Locking Sequences

Existence of Locking Sequences

[Blum and Blum] Let M be a recursive learner and $\mathcal{L}$ a co-r.e. set learnt by M. Then there is a finite sequence σ of elements in $\mathcal{L} \cup \{\#\}$ such that

- $\overline{W}_{M(\sigma)} = \mathcal{L}$;
- For all finite sequences τ of elements in $\mathcal{L} \cup \{\#\}$, $M(\sigma\tau) = M(\sigma)$. That is, when M is fed with any sequence in $\mathcal{L} \cup \{\#\}$ extending σ, it does not update its hypothesis.
- This σ is called a *locking sequence* for $\mathcal{L}$.

Uniformly Indexed Families of Sets

- We look at uniformly indexed families:

Uniformly Indexed Families of Sets

- We look at uniformly indexed families:
  - An indexed family of sets $L_0, L_1, L_2, \ldots$ is uniformly recursive iff the set $\{\langle i, x \rangle : x \in L_i\}$ is recursive. Each member of this family is a recursive set.

Uniformly Indexed Families of Sets

- We look at uniformly indexed families:
  - An indexed family of sets $L_0, L_1, L_2, \ldots$ is uniformly recursive iff the set $\{\langle i, x\rangle : x \in L_i\}$ is recursive. Each member of this family is a recursive set.
  - An indexed family of sets $L_0, L_1, L_2, \ldots$ is uniformly recursively enumerable iff the set $\{\langle i, x\rangle : x \in L_i\}$ is recursively enumerable. Each member of this family is an r.e. set.

Uniformly Indexed Families of Sets

- We look at uniformly indexed families:
  - An indexed family of sets $L_0, L_1, L_2, \ldots$ is uniformly recursive iff the set $\{\langle i, x\rangle : x \in L_i\}$ is recursive. Each member of this family is a recursive set.
  - An indexed family of sets $L_0, L_1, L_2, \ldots$ is uniformly recursively enumerable iff the set $\{\langle i, x\rangle : x \in L_i\}$ is recursively enumerable. Each member of this family is an r.e. set.
  - An indexed family of sets $L_0, L_1, L_2, \ldots$ is uniformly co-recursively enumerable iff the set $\{\langle i, x\rangle : x \notin L_i\}$ is recursively enumerable. Each member of this family is a co-r.e. set.

Uniformly Indexed Families of Sets

- We look at uniformly indexed families:
  - An indexed family of sets $L_0, L_1, L_2, \ldots$ is uniformly recursive iff the set $\{\langle i, x\rangle : x \in L_i\}$ is recursive. Each member of this family is a recursive set.
  - An indexed family of sets $L_0, L_1, L_2, \ldots$ is uniformly recursively enumerable iff the set $\{\langle i, x\rangle : x \in L_i\}$ is recursively enumerable. Each member of this family is an r.e. set.
  - An indexed family of sets $L_0, L_1, L_2, \ldots$ is uniformly co-recursively enumerable iff the set $\{\langle i, x\rangle : x \notin L_i\}$ is recursively enumerable. Each member of this family is a co-r.e. set.
- A criterion for a uniformly recursive class of sets to be learnable may be formulated as a "tell-tale" condition.

Tell-tale Characterisation of $Cnsv_{co-r.e.}$, $BC_{co-r.e.}$, or $Ex_{r.e.}$ Learnable Indexed Families of Recursive Languages

Theorem (Extension of Angluin's Theorem)

A necessary and sufficient condition for a uniformly recursive family $\mathcal{C} = \{L_0, L_1, L_2, \ldots\}$ of sets to be $Cnsv_{co-r.e.}$, $BC_{co-r.e.}$, or $Ex_{r.e.}$ learnable is that there is a uniformly r.e. family of finite sets $\{H_i\}_{i \in \mathbb{N}}$ such that, for every $i, j \in \mathbb{N}$,

- $H_i \subseteq L_i$;
- *if* $H_i \subseteq L_j \subseteq L_i$, *then* $L_j = L_i$.

$\{H_i\}_{i \in \mathbb{N}}$ is said to be a family of tell-tale sets for $\mathcal{C}$, and for each L_i, H_i is said to be a tell-tale set for L_i.

Tell-tale Characterisation of $Cnsv_{co-r.e.}$, $BC_{co-r.e.}$, or $Ex_{r.e.}$ Learnable Indexed Families of Recursive Languages

Theorem (Extension of Angluin's Theorem)

A necessary and sufficient condition for a uniformly recursive family $\mathcal{C} = \{L_0, L_1, L_2, \ldots\}$ of sets to be $Cnsv_{co-r.e.}$, $BC_{co-r.e.}$, or $Ex_{r.e.}$ learnable is that there is a uniformly r.e. family of finite sets $\{H_i\}_{i\in\mathbb{N}}$ such that, for every $i, j \in \mathbb{N}$,

- $H_i \subseteq L_i$;
- *if* $H_i \subseteq L_j \subseteq L_i$, *then* $L_j = L_i$.

$\{H_i\}_{i\in\mathbb{N}}$ *is said to be a family of tell-tale sets for* $\mathcal{C}$, *and for each* L_i, H_i *is said to be a tell-tale set for* L_i.

- A class of sets for which a family of tell-tale sets exists is said to satisfy *Angluin's tell-tale condition*.

Tell-tale Characterisation of $Ex_{r.e}$ Learnable Indexed Families of R.e. Languages

Theorem (de Jongh and Kanazawa)

An indexed family of r.e. languages $\mathcal{L} = \{L_i : i \in \mathbb{N}\}$ is explanatorily learnable if and only if there is a limiting recursive function δ mapping each i to a finite set D_i and a limiting recursive function ϵ mapping i to an index e_i for an r.e. set $E_i = W_{e_i}$ of nonempty finite sets, such that for every $i, j \in \mathbb{N}$,

- *D_i is a tell-tale set for L_i in $\mathcal{L}$;*
- *for all $D \in E_i$, $D \not\subseteq L_i$;*
- *if $D_i \subseteq L_j$ and $L_j - L_i \neq \emptyset$, then there is a set $D \in E_i$ such that $D \subseteq L_j$.*

Tell-tale Characterisation of $Ex_{r.e}$ Learnable Indexed Families of R.e. Languages

Theorem (de Jongh and Kanazawa)

An indexed family of r.e. languages $\mathcal{L} = \{L_i : i \in \mathbb{N}\}$ is explanatorily learnable if and only if there is a limiting recursive function δ mapping each i to a finite set D_i and a limiting recursive function ϵ mapping i to an index e_i for an r.e. set $E_i = W_{e_i}$ of nonempty finite sets, such that for every $i, j \in \mathbb{N}$,

- *D_i is a tell-tale set for L_i in $\mathcal{L}$;*
- *for all $D \in E_i$, $D \not\subseteq L_i$;*
- *if $D_i \subseteq L_j$ and $L_j - L_i \neq \emptyset$, then there is a set $D \in E_i$ such that $D \subseteq L_j$.*

- de Jongh and Kanazawa call E_i a collection of “warning sets” for L_i in $\{L_j : D_i \subseteq L_j\}$.

Conservative Learning of Co-r.e. Classes

[Angluin] A recursive learner M is said to *conservatively* ($Cnsv_{co-r.e.}$) learn $\mathcal{C}$ if it $Ex_{co-r.e.}$ learns $\mathcal{C}$ and if, given any two finite sequences $\sigma, \tau \in (\mathbb{N} \cup \{\#\})^*$ such that $M(\sigma) \neq M(\sigma\tau)$, there is a number x with $x \in \text{range}(\sigma\tau) - \overline{W}_{M(\sigma)}$.

Conservative Learning of Co-r.e. Classes

[Angluin] A recursive learner M is said to *conservatively* ($Cnsv_{co-r.e.}$) learn $\mathcal{C}$ if it $Ex_{co-r.e.}$ learns $\mathcal{C}$ and if, given any two finite sequences $\sigma, \tau \in (\mathbb{N} \cup \{\#\})^*$ such that $M(\sigma) \neq M(\sigma\tau)$, there is a number x with $x \in \text{range}(\sigma\tau) - \overline{W}_{M(\sigma)}$.

Data	M's conjecture	Mind change?

Conservative Learning of Co-r.e. Classes

[Angluin] A recursive learner M is said to *conservatively* ($Cnsv_{co-r.e.}$) learn $\mathcal{C}$ if it $Ex_{co-r.e.}$ learns $\mathcal{C}$ and if, given any two finite sequences $\sigma, \tau \in (\mathbb{N} \cup \{\#\})^*$ such that $M(\sigma) \neq M(\sigma\tau)$, there is a number x with $x \in \text{range}(\sigma\tau) - \overline{W}_{M(\sigma)}$.

Data	M's conjecture	Mind change?
3	$\overline{W}_1 = \emptyset$	$*$

Conservative Learning of Co-r.e. Classes

[Angluin] A recursive learner M is said to *conservatively* ($Cnsv_{co-r.e.}$) learn $\mathcal{C}$ if it $Ex_{co-r.e.}$ learns $\mathcal{C}$ and if, given any two finite sequences $\sigma, \tau \in (\mathbb{N} \cup \{\#\})^*$ such that $M(\sigma) \neq M(\sigma\tau)$, there is a number x with $x \in \text{range}(\sigma\tau) - \overline{W}_{M(\sigma)}$.

Data	M's conjecture	Mind change?
3	$\overline{W}_1 = \emptyset$	$*$
$3, 8$	$\overline{W}_1 = \emptyset$	No

Conservative Learning of Co-r.e. Classes

[Angluin] A recursive learner M is said to *conservatively* ($Cnsv_{co-r.e.}$) learn $\mathcal{C}$ if it $Ex_{co-r.e.}$ learns $\mathcal{C}$ and if, given any two finite sequences $\sigma, \tau \in (\mathbb{N} \cup \{\#\})^*$ such that $M(\sigma) \neq M(\sigma\tau)$, there is a number x with $x \in \text{range}(\sigma\tau) - \overline{W}_{M(\sigma)}$.

Data	M's conjecture	Mind change?
3	$\overline{W}_1 = \emptyset$	$*$
$3, 8$	$\overline{W}_1 = \emptyset$	No
$3, 8, 5$	$\overline{W}_2 = \{3, 8\}$	Yes

Conservative Learning of Co-r.e. Classes

[Angluin] A recursive learner M is said to *conservatively* ($Cnsv_{co-r.e.}$) learn $\mathcal{C}$ if it $Ex_{co-r.e.}$ learns $\mathcal{C}$ and if, given any two finite sequences $\sigma, \tau \in (\mathbb{N} \cup \{\#\})^*$ such that $M(\sigma) \neq M(\sigma\tau)$, there is a number x with $x \in \text{range}(\sigma\tau) - \overline{W}_{M(\sigma)}$.

Data	M's conjecture	Mind change?
3	$\overline{W}_1 = \emptyset$	$*$
$3, 8$	$\overline{W}_1 = \emptyset$	No
$3, 8, 5$	$\overline{W}_2 = \{3, 8\}$	Yes
$3, 8, 5, 13$	$\overline{W}_2 = \{3, 8\}$	No

Conservative Learning of Co-r.e. Classes

[Angluin] A recursive learner M is said to *conservatively* ($Cnsv_{co-r.e.}$) learn $\mathcal{C}$ if it $Ex_{co-r.e.}$ learns $\mathcal{C}$ and if, given any two finite sequences $\sigma, \tau \in (\mathbb{N} \cup \{\#\})^*$ such that $M(\sigma) \neq M(\sigma\tau)$, there is a number x with $x \in \text{range}(\sigma\tau) - \overline{W}_{M(\sigma)}$.

Data	M's conjecture	Mind change?
3	$\overline{W}_1 = \emptyset$	$*$
$3, 8$	$\overline{W}_1 = \emptyset$	No
$3, 8, 5$	$\overline{W}_2 = \{3, 8\}$	Yes
$3, 8, 5, 13$	$\overline{W}_2 = \{3, 8\}$	No
$3, 8, 5, 13, \#$	$\overline{W}_2 = \{3, 8\}$	No

Conservative Learning of Co-r.e. Classes

[Angluin] A recursive learner M is said to *conservatively* ($Cnsv_{co-r.e.}$) learn $\mathcal{C}$ if it $Ex_{co-r.e.}$ learns $\mathcal{C}$ and if, given any two finite sequences $\sigma, \tau \in (\mathbb{N} \cup \{\#\})^*$ such that $M(\sigma) \neq M(\sigma\tau)$, there is a number x with $x \in \text{range}(\sigma\tau) - \overline{W}_{M(\sigma)}$.

Data	M's conjecture	Mind change?
3	$\overline{W}_1 = \emptyset$	$*$
$3, 8$	$\overline{W}_1 = \emptyset$	No
$3, 8, 5$	$\overline{W}_2 = \{3, 8\}$	Yes
$3, 8, 5, 13$	$\overline{W}_2 = \{3, 8\}$	No
$3, 8, 5, 13, \#$	$\overline{W}_2 = \{3, 8\}$	No
$3, 8, 5, 13, \#, 2$	$\overline{W}_3 = \{3, 8, 5, 13, 2\}$	Yes

Conservative Learning of Co-r.e. Classes

[Angluin] A recursive learner M is said to *conservatively* ($Cnsv_{co-r.e.}$) learn $\mathcal{C}$ if it $Ex_{co-r.e.}$ learns $\mathcal{C}$ and if, given any two finite sequences $\sigma, \tau \in (\mathbb{N} \cup \{\#\})^*$ such that $M(\sigma) \neq M(\sigma\tau)$, there is a number x with $x \in \text{range}(\sigma\tau) - \overline{W}_{M(\sigma)}$.

Data	M's conjecture	Mind change?
3	$\overline{W}_1 = \emptyset$	$*$
$3, 8$	$\overline{W}_1 = \emptyset$	No
$3, 8, 5$	$\overline{W}_2 = \{3, 8\}$	Yes
$3, 8, 5, 13$	$\overline{W}_2 = \{3, 8\}$	No
$3, 8, 5, 13, \#$	$\overline{W}_2 = \{3, 8\}$	No
$3, 8, 5, 13, \#, 2$	$\overline{W}_3 = \{3, 8, 5, 13, 2\}$	Yes

Tell-tale Characterisation of Conservatively Learnable Indexed Families of R.e. Languages

- A family of finite sets $\{H_i\}_{i \in \mathbb{N}}$ is *uniformly recursively generable* if there is a total effective procedure g, which, on every input j, generates all elements of H_j and stops.

Tell-tale Characterisation of Conservatively Learnable Indexed Families of R.e. Languages

- A family of finite sets $\{H_i\}_{i \in \mathbb{N}}$ is *uniformly recursively generable* if there is a total effective procedure g, which, on every input j, generates all elements of H_j and stops.

Tell-tale Characterisation of Conservatively Learnable Indexed Families of R.e. Languages

- A family of finite sets $\{H_i\}_{i\in\mathbb{N}}$ is *uniformly recursively generable* if there is a total effective procedure g, which, on every input j, generates all elements of H_j and stops.

Theorem (Lange, Zeugmann and Kapur)

Let $\mathcal{L}$ be an indexed family of non-empty recursive languages. Then $\mathcal{L}$ is conservatively learnable if and only if there is a hypothesis space $\mathcal{G} = \{G_i\}_{i\in\mathbb{N}}$ and a uniformly recursively generable family $\{H_i\}_{i\in\mathbb{N}}$ of finite non-empty sets such that

- $range(\mathcal{L}) \subseteq range(\mathcal{G})$;
- *for all* $i \in \mathbb{N}$, $H_i \subseteq G_i$;
- *for all* $i, j \in \mathbb{N}$ *with* $H_i \subseteq G_j$ *it holds that* $G_j \not\subset G_i$.

Tell-tale Characterisation of Conservatively Learnable Families of Co-r.e. Languages

- We say that $\{L_0, L_1, L_2, \ldots\}$ *covers* a class $\mathcal{C}$ just if $\mathcal{C} \subseteq \{L_0, L_1, L_2, \ldots\}$.

Tell-tale Characterisation of Conservatively Learnable Families of Co-r.e. Languages

- We say that $\{L_0, L_1, L_2, \dots\}$ *covers* a class $\mathcal{C}$ just if $\mathcal{C} \subseteq \{L_0, L_1, L_2, \dots\}$.
- A tell-tale characterisation of all conservatively learnable classes of co-r.e. sets may be formulated as follows.

Tell-tale Characterisation of Conservatively Learnable Families of Co-r.e. Languages

- We say that $\{L_0, L_1, L_2, \ldots\}$ *covers* a class $\mathcal{C}$ just if $\mathcal{C} \subseteq \{L_0, L_1, L_2, \ldots\}$.
- A tell-tale characterisation of all conservatively learnable classes of co-r.e. sets may be formulated as follows.

Tell-tale Characterisation of Conservatively Learnable Families of Co-r.e. Languages

- We say that $\{L_0, L_1, L_2, \ldots\}$ *covers* a class $\mathcal{C}$ just if $\mathcal{C} \subseteq \{L_0, L_1, L_2, \ldots\}$.
- A tell-tale characterisation of all conservatively learnable classes of co-r.e. sets may be formulated as follows.

Theorem

Let $\mathcal{C}$ be a class of co-r.e. sets. Then $\mathcal{C}$ is $Cnsv_{co-r.e.}$ learnable iff there is a uniformly co-r.e. family $L_0, L_1, L_2, \ldots$ covering $\mathcal{C}$, and a canonical indexing of finite recursive sets $H_0, H_1, H_2, \ldots$ such that for all n, m, $H_n \subseteq L_n$, and whenever $H_n \subseteq L_m \subseteq L_n$, it holds that $L_m = L_n$.

Tell-tale Characterisation of Conservatively Learnable Families of Co-r.e. Languages

- We say that $\{L_0, L_1, L_2, \ldots\}$ *covers* a class $\mathcal{C}$ just if $\mathcal{C} \subseteq \{L_0, L_1, L_2, \ldots\}$.
- A tell-tale characterisation of all conservatively learnable classes of co-r.e. sets may be formulated as follows.

Theorem

Let $\mathcal{C}$ be a class of co-r.e. sets. Then $\mathcal{C}$ is $Cnsv_{co-r.e.}$ learnable iff there is a uniformly co-r.e. family $L_0, L_1, L_2, \ldots$ covering $\mathcal{C}$, and a canonical indexing of finite recursive sets $H_0, H_1, H_2, \ldots$ such that for all n, m, $H_n \subseteq L_n$, and whenever $H_n \subseteq L_m \subseteq L_n$, it holds that $L_m = L_n$.

- The family of finite recursive sets $\{H_0, H_1, H_2, \ldots\}$ is, in the sense of the previous theorem, uniformly recursively generable.

Conservative Learning of Co-r.e. Classes

Proof (⇒)

- Suppose that M is a $Cnsv_{co-r.e.}$ learner of $\mathcal{C}$.

Conservative Learning of Co-r.e. Classes

Proof (⇒)

- Suppose that M is a $Cnsv_{co-r.e.}$ learner of $\mathcal{C}$.
- Let $\tau_0, \tau_1, \tau_2, \ldots$ be an enumeration of all sequences which satisfy the following conditions:

Conservative Learning of Co-r.e. Classes

Proof (⇒)

- Suppose that M is a $Cnsv_{co-r.e.}$ learner of $\mathcal{C}$.
- Let $\tau_0, \tau_1, \tau_2, \ldots$ be an enumeration of all sequences which satisfy the following conditions:
  - A number x occurs in τ_n if and only if it occurs in τ_n exactly in position x;

Conservative Learning of Co-r.e. Classes

Proof (⇒)

- Suppose that M is a $Cnsv_{co-r.e.}$ learner of $\mathcal{C}$.
- Let $\tau_0, \tau_1, \tau_2, \ldots$ be an enumeration of all sequences which satisfy the following conditions:
  - A number x occurs in τ_n if and only if it occurs in τ_n exactly in position x;
  - For all $x < |\tau_n|$ such that $x \notin \text{range}(\tau_n)$, $x \in W_{M(\tau_n)}$;

Conservative Learning of Co-r.e. Classes

Proof ($\Rightarrow$)

- Suppose that M is a $Cnsv_{co-r.e.}$ learner of $\mathcal{C}$.
- Let $\tau_0, \tau_1, \tau_2, \ldots$ be an enumeration of all sequences which satisfy the following conditions:
  - A number x occurs in τ_n if and only if it occurs in τ_n exactly in position x;
  - For all $x < |\tau_n|$ such that $x \notin \text{range}(\tau_n)$, $x \in W_{M(\tau_n)}$;
  - For all $\sigma \prec \tau_n$, $M(\sigma) \neq M(\tau_n)$.

Conservative Learning of Co-r.e. Classes

Proof (⇒)

- Now define a uniformly co-r.e. family of sets $\{L_n\}_{n\in\mathbb{N}}$ by

$$L_n = \begin{cases} \overline{W}_{M(\tau_n)} & \text{if } W_{M(\tau_n)} \cap \text{range}(\tau_n) = \emptyset; \\ \text{range}(\tau_n) - \{\#\} & \text{if } W_{M(\tau_n)} \cap \text{range}(\tau_n) \neq \emptyset, \end{cases}$$

and put $H_n = \text{range}(\tau_n) - \{\#\}$ for all n.

Conservative Learning of Co-r.e. Classes

Proof (⇒)

- Now define a uniformly co-r.e. family of sets $\{L_n\}_{n\in\mathbb{N}}$ by

$$L_n = \begin{cases} \overline{W}_{M(\tau_n)} & \text{if } W_{M(\tau_n)} \cap \text{range}(\tau_n) = \emptyset; \\ \text{range}(\tau_n) - \{\#\} & \text{if } W_{M(\tau_n)} \cap \text{range}(\tau_n) \neq \emptyset, \end{cases}$$

  and put $H_n = \text{range}(\tau_n) - \{\#\}$ for all n.
- $\{H_n\}_{n\in\mathbb{N}}$ is a uniformly recursive family of finite sets such that $H_n \subseteq L_n$ holds for all n.

Conservative Learning of Co-r.e. Classes

Proof (⇒)

- Suppose that for some $i, j, H_i = \text{range}(\tau_i) - \{\#\} \subseteq L_j \subseteq L_i$.

Conservative Learning of Co-r.e. Classes

Proof (⇒)

- Suppose that for some $i, j, H_i = \text{range}(\tau_i) - \{\#\} \subseteq L_j \subseteq L_i$.

  1. Suppose that $\tau_i \prec \tau_j$. By the third condition in the construction of $\{\tau_n\}_{n\in\mathbb{N}}$, $M(\tau_i) \neq M(\tau_j)$, and so by the conservativeness of M, there is an $x \in \text{range}(\tau_j) \cap W_{M(\tau_i)}$, and as $\text{range}(\tau_j) - \{\#\} \subseteq L_j$, this implies that $L_j \not\subset L_i$, a contradiction.

Conservative Learning of Co-r.e. Classes

Proof ($\Rightarrow$)

- Suppose that for some $i, j, H_i = \text{range}(\tau_i) - \{\#\} \subseteq L_j \subseteq L_i$.

1. Suppose that $\tau_i \prec \tau_j$. By the third condition in the construction of $\{\tau_n\}_{n \in \mathbb{N}}$, $M(\tau_i) \neq M(\tau_j)$, and so by the conservativeness of M, there is an $x \in \text{range}(\tau_j) \cap W_{M(\tau_i)}$, and as $\text{range}(\tau_j) - \{\#\} \subseteq L_j$, this implies that $L_j \not\subset L_i$, a contradiction.
2. Suppose that $\tau_j \prec \tau_i$. Again, $M(\tau_i) \neq M(\tau_j)$, and so by the conservativeness of M, there is an $x \in \text{range}(\tau_i) \cap W_{M(\tau_j)}$. Hence $\text{range}(\tau_i) - \{\#\} \not\subset L_j$, a contradiction.

Conservative Learning of Co-r.e. Classes

Proof ($\Rightarrow$)

3. Suppose that $\tau_i(x) \neq \tau_j(x)$ for some $x < \min\{|\tau_i|, |\tau_j|\}$. Let e be the least such x. If $\tau_i(e) = \#$ and $\tau_j(e) = e$, then by the second condition in the construction of $\{\tau_n\}_{n \in \mathbb{N}}$, $e \in W_{M(\tau_i)}$. Further, if $L_i = \text{range}(\tau_i) - \{\#\}$, then one has $e \in L_j - L_i$, contrary to the condition that $L_j \subseteq L_i$. If $L_i = \overline{W}_{M(\tau_i)}$, then again $e \in L_j - L_i$, giving rise to the same contradiction. On the other hand, if $\tau_i(e) = e$ and $\tau_j(e) = \#$, then $e \in \text{range}(\tau_i) - L_j$, contradicting the condition that $\text{range}(\tau_i) - \{\#\} \subseteq L_j$.

Conservative Learning of Co-r.e. Classes

Proof ($\Rightarrow$)

3. Suppose that $\tau_i(x) \neq \tau_j(x)$ for some $x < \min\{|\tau_i|, |\tau_j|\}$. Let e be the least such x. If $\tau_i(e) = \#$ and $\tau_j(e) = e$, then by the second condition in the construction of $\{\tau_n\}_{n \in \mathbb{N}}$, $e \in W_{M(\tau_i)}$. Further, if $L_i = \text{range}(\tau_i) - \{\#\}$, then one has $e \in L_j - L_i$, contrary to the condition that $L_j \subseteq L_i$. If $L_i = \overline{W}_{M(\tau_i)}$, then again $e \in L_j - L_i$, giving rise to the same contradiction. On the other hand, if $\tau_i(e) = e$ and $\tau_j(e) = \#$, then $e \in \text{range}(\tau_i) - L_j$, contradicting the condition that $\text{range}(\tau_i) - \{\#\} \subseteq L_j$.

- Therefore $\tau_i = \tau_j$, and so $L_i = L_j$. This completes one direction of the proof.

Conservative Learning of Co-r.e. Classes

Proof ($\Leftarrow$)

- Let $\{L_e\}_{e \in \mathbb{N}}$ be the uniformly co-r.e. class covering $\mathcal{C}$, and $\{H_e\}_{e \in \mathbb{N}}$ the corresponding uniformly recursive class of tell-tale sets.

Conservative Learning of Co-r.e. Classes

Proof ($\Leftarrow$)

- Let $\{L_e\}_{e \in \mathbb{N}}$ be the uniformly co-r.e. class covering $\mathcal{C}$, and $\{H_e\}_{e \in \mathbb{N}}$ the corresponding uniformly recursive class of tell-tale sets.
- Denote the approximation to $\overline{L}_e$ at step n by $J_{e,n}$.

Conservative Learning of Co-r.e. Classes

Proof ($\Leftarrow$)

- Let $\{L_e\}_{e\in\mathbb{N}}$ be the uniformly co-r.e. class covering $\mathcal{C}$, and $\{H_e\}_{e\in\mathbb{N}}$ the corresponding uniformly recursive class of tell-tale sets.
- Denote the approximation to $\overline{L}_e$ at step n by $J_{e,n}$.
- Upon reading the input σ, M searches for the least index $e \leq n$ so that

$$H_{e,n} \subseteq \text{range}(\sigma) - \{\#\} \subseteq \overline{J}_{e,n}.$$

If no such index is found, let M output an index for $\mathbb{N}$; otherwise, set M to conjecture an index c with

$$\overline{W}_c = \begin{cases} \emptyset & \text{if } H_e \neq H_{e,n}; \\ L_e & \text{if } H_e = H_{e,n}. \end{cases}$$

Conservative Learning of Co-r.e. Classes

Proof (⇐)

- One can verify that M is indeed a $Cnsv_{co-r.e.}$ learner of $\mathcal{C}$ $\square$.

Conservative Learning of Co-r.e. Classes

Proof (⇐)

- One can verify that M is indeed a $Cnsv_{co-r.e.}$ learner of $\mathcal{C}$ $\square$.
- Furthermore, if one explicitly adds the empty set to $\{L_0, L_1, L_2, \ldots\}$, then we have that M is even "prudent" - that is, M $Cnsv_{co-r.e.}$ learns every set it conjectures.

Goals of the Project

Objective 2

- Separation of co-r.e. learning notions.

Overview of Results

General Classes of Co-Recursively Enumerable Sets

$\mathbf{Const}$ & $\mathbf{Cnsv_{co-r.e.}}$ $=$ $\mathbf{Const}$ & $\mathbf{PrudCnsv_{co-r.e.}}$ & $\mathbf{SetDriven}$

$\cap$

$\mathbf{Cnsv_{co-r.e.}} = \mathbf{PrudCnsv_{co-r.e.}} = \mathbf{WeakCnsv_{co-r.e.}}$

$=$ $\mathbf{Const}$ & $\mathbf{WeakCnsv_{co-r.e.}}$ & $\mathbf{SetDriven}$

$\cap$

$\mathbf{PrudEx_{co-r.e.}} \subset \mathbf{PrudBC_{co-r.e.}}$

$\cap$ $\quad\quad\quad\quad\quad\quad\quad\quad$ $\cap$

$\mathbf{Ex_{co-r.e.}} = \mathbf{Vac_{co-r.e.}} \subset \mathbf{BC_{co-r.e.}}$;

Uniformly r.e.: $\mathbf{BC_{co-r.e.}} \subset \mathbf{Ex_{rec}}$

Overview of Results

General Classes of Recursive Sets

$Fin_{co-r.e.} - BC_{r.e.} \neq \emptyset$

$Fin_{r.e.} - BC_{co-r.e.} \neq \emptyset$

Vacillatory Learning

[Case] A recursive learner M is said to *vacillatorily* ($Vac_{co-r.e.}$) learn $\mathcal{C}$ if it $BC_{co-r.e.}$ learns $\mathcal{C}$ and outputs on every text T_L for each L in $\mathcal{C}$ only finitely many different indices $i_0, i_1, \ldots, i_n$.

Vacillatory Learning

[Case] A recursive learner M is said to *vacillatorily* ($Vac_{co-r.e.}$) learn $\mathcal{C}$ if it $BC_{co-r.e.}$ learns $\mathcal{C}$ and outputs on every text T_L for each L in $\mathcal{C}$ only finitely many different indices $i_0, i_1, \ldots, i_n$.

- The finite set of indices that M outputs may vary with either L or the corresponding text T_L presented.

Vacillatory Learning

[Case] A recursive learner M is said to *vacillatorily* ($Vac_{co-r.e.}$) learn $\mathcal{C}$ if it $BC_{co-r.e.}$ learns $\mathcal{C}$ and outputs on every text T_L for each L in $\mathcal{C}$ only finitely many different indices $i_0, i_1, \ldots, i_n$.

- The finite set of indices that M outputs may vary with either L or the corresponding text T_L presented.
- A slight peculiarity of co-r.e. learning is that vacillatory and explanatory learning coincide.

Vacillatory Learning

[Case] A recursive learner M is said to *vacillatorily* ($Vac_{co-r.e.}$) learn $\mathcal{C}$ if it $BC_{co-r.e.}$ learns $\mathcal{C}$ and outputs on every text T_L for each L in $\mathcal{C}$ only finitely many different indices $i_0, i_1, \ldots, i_n$.

- The finite set of indices that M outputs may vary with either L or the corresponding text T_L presented.
- A slight peculiarity of co-r.e. learning is that vacillatory and explanatory learning coincide.

Vacillatory Learning

[Case] A recursive learner M is said to *vacillatorily* ($Vac_{co-r.e.}$) learn $\mathcal{C}$ if it $BC_{co-r.e.}$ learns $\mathcal{C}$ and outputs on every text T_L for each L in $\mathcal{C}$ only finitely many different indices $i_0, i_1, \ldots, i_n$.

- The finite set of indices that M outputs may vary with either L or the corresponding text T_L presented.
- A slight peculiarity of co-r.e. learning is that vacillatory and explanatory learning coincide.

Theorem

If a class is $Vac_{co-r.e.}$ learnable, then it is $Ex_{co-r.e.}$ learnable.

Vacillatory Learning

[Case] A recursive learner M is said to *vacillatorily* ($Vac_{co-r.e.}$) learn $\mathcal{C}$ if it $BC_{co-r.e.}$ learns $\mathcal{C}$ and outputs on every text T_L for each L in $\mathcal{C}$ only finitely many different indices $i_0, i_1, \ldots, i_n$.

- The finite set of indices that M outputs may vary with either L or the corresponding text T_L presented.
- A slight peculiarity of co-r.e. learning is that vacillatory and explanatory learning coincide.

Theorem

If a class is $Vac_{co-r.e.}$ learnable, then it is $Ex_{co-r.e.}$ learnable.

- This result does not hold for the r.e. case.

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W}_i = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W}_i = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

- Again, we suppose that $\overline{W}_0$ is the set of Fibonacci numbers.

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W}_i = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

- Again, we suppose that $\overline{W}_0$ is the set of Fibonacci numbers.

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W}_i = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

- Again, we suppose that $\overline{W}_0$ is the set of Fibonacci numbers.

Data	M's conjecture	Stabilised?

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W}_i = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

- Again, we suppose that $\overline{W}_0$ is the set of Fibonacci numbers.

Data	M's conjecture	Stabilised?
3	?	No

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W_i} = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

- Again, we suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture	Stabilised?
3	?	No
$3, 8$	?	No

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W_i} = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

- Again, we suppose that $\overline{W_0}$ is the set of Fibonacci numbers.

Data	M's conjecture	Stabilised?
3	?	No
3, 8	?	No
3, 8, 5	?	No

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W}_i = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

- Again, we suppose that $\overline{W}_0$ is the set of Fibonacci numbers.

Data	M's conjecture	Stabilised?
3	?	No
$3, 8$	?	No
$3, 8, 5$	?	No
$3, 8, 5, 13$	?	No

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W}_i = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

- Again, we suppose that $\overline{W}_0$ is the set of Fibonacci numbers.

Data	M's conjecture	Stabilised?
3	?	No
$3, 8$	?	No
$3, 8, 5$	?	No
$3, 8, 5, 13$	?	No
$3, 8, 5, 13, \#$	?	No

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W}_i = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

- Again, we suppose that $\overline{W}_0$ is the set of Fibonacci numbers.

Data	M's conjecture	Stabilised?
3	?	No
$3, 8$	?	No
$3, 8, 5$	?	No
$3, 8, 5, 13$	?	No
$3, 8, 5, 13, \#$	?	No
$3, 8, 5, 13, \#, 2$	?	No

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W}_i = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

- Again, we suppose that $\overline{W}_0$ is the set of Fibonacci numbers.

Data	M's conjecture	Stabilised?
3	?	No
$3, 8$	?	No
$3, 8, 5$	?	No
$3, 8, 5, 13$	?	No
$3, 8, 5, 13, \#$	?	No
$3, 8, 5, 13, \#, 2$	?	No
$3, 8, 5, 13, \#, 2, 1$	0	Yes

Finite Learning

[Bārzdiņš and Freivalds] A recursive learner M is said to *finitely* ($Fin_{co-r.e.}$) learn $\mathcal{C}$ if, for each L in $\mathcal{C}$ and any text T_L for L, there is an index i and a number n such that $\overline{W}_i = L$, $M(T_L[k]) = i$ for all $k \geq n$ and $M(T_L[k]) = ?$ for all $k < n$.

- Again, we suppose that $\overline{W}_0$ is the set of Fibonacci numbers.

Data	M's conjecture	Stabilised?
3	?	No
$3, 8$	?	No
$3, 8, 5$	?	No
$3, 8, 5, 13$	?	No
$3, 8, 5, 13, \#$	?	No
$3, 8, 5, 13, \#, 2$	?	No
$3, 8, 5, 13, \#, 2, 1$	0	Yes

Finite Learning Versus BC Learning

- The following is a pair of complementary results for finite versus behaviourally correct learning.

Finite Learning Versus BC Learning

- The following is a pair of complementary results for finite versus behaviourally correct learning.

Finite Learning Versus BC Learning

- The following is a pair of complementary results for finite versus behaviourally correct learning.

Theorem

There is a uniformly co-r.e. class of recursive sets which is $Fin_{co-r.e.}$ learnable but not $BC_{r.e.}$ learnable.

Finite Learning Versus BC Learning

- The following is a pair of complementary results for finite versus behaviourally correct learning.

Theorem

There is a uniformly co-r.e. class of recursive sets which is $Fin_{co-r.e.}$ learnable but not $BC_{r.e.}$ learnable.

Theorem

There is a uniformly r.e. class of recursive sets which is $Fin_{r.e.}$ learnable but not $BC_{co-r.e.}$ learnable.

Finite Learning Versus BC Learning

Proof for $Fin_{r.e.} - BC_{co-r.e.} \neq \emptyset$

- Let $L_n = \{n\} \oplus \{m : W_{n,m} \subset W_n \vee (m > 0 \wedge W_{n,m} = W_{n,m-1})\}$ and $\mathcal{C} = \{L_0, L_1, \ldots\}$.

Finite Learning Versus BC Learning

Proof for $Fin_{r.e.} - BC_{co-r.e.} \neq \emptyset$

- Let $L_n = \{n\} \oplus \{m : W_{n,m} \subset W_n \vee (m > 0 \wedge W_{n,m} = W_{n,m-1})\}$ and $\mathcal{C} = \{L_0, L_1, \ldots\}$.
- This family is uniformly r.e.

Finite Learning Versus BC Learning

Proof for $Fin_{r.e.} - BC_{co-r.e.} \neq \emptyset$

- Let $L_n = \{n\} \oplus \{m : W_{n,m} \subset W_n \vee (m > 0 \wedge W_{n,m} = W_{n,m-1})\}$ and $\mathcal{C} = \{L_0, L_1, \ldots\}$.
- This family is uniformly r.e.
- This family is $Fin_{r.e.}$ learnable: on seeing the first n such that $2n$ is contained in the input, the learner M conjectures an r.e. index for L_n.

Finite Learning Versus BC Learning

Proof for $Fin_{r.e.} - BC_{co-r.e.} \neq \emptyset$

- Let $L_n = \{n\} \oplus \{m : W_{n,m} \subset W_n \vee (m > 0 \wedge W_{n,m} = W_{n,m-1})\}$ and $\mathcal{C} = \{L_0, L_1, \ldots\}$.
- This family is uniformly r.e.
- This family is $Fin_{r.e.}$ learnable: on seeing the first n such that $2n$ is contained in the input, the learner M conjectures an r.e. index for L_n.
- Assume that some recursive learner N $BC_{co-r.e.}$ learns $\mathcal{C}$.

Finite Learning Versus BC Learning

Proof for $Fin_{r.e.} - BC_{co-r.e.} \neq \emptyset$

- Let $L_n = \{n\} \oplus \{m : W_{n,m} \subset W_n \vee (m > 0 \wedge W_{n,m} = W_{n,m-1})\}$ and $\mathcal{C} = \{L_0, L_1, \ldots\}$.
- This family is uniformly r.e.
- This family is $Fin_{r.e.}$ learnable: on seeing the first n such that $2n$ is contained in the input, the learner M conjectures an r.e. index for L_n.
- Assume that some recursive learner N $BC_{co-r.e.}$ learns $\mathcal{C}$.
- Then W_n is infinite iff on every text T_L for L_n, there is a sufficiently long prefix $T_L[l]$ of T_L such that $W_{M(T_L[k])}$ does not contain any number from $\{n\} \oplus \mathbb{N}$ for all $k \geq l$.

Finite Learning Versus BC Learning

Proof for $Fin_{r.e.} - BC_{co-r.e.} \neq \emptyset$

- Hence the index set $\{n : |W_n| = \infty\}$ is reducible to a Σ^0_2 formula, which is impossible.

Finite Learning Versus BC Learning

Proof for $Fin_{r.e.} - BC_{co-r.e.} \neq \emptyset$

- Hence the index set $\{n : |W_n| = \infty\}$ is reducible to a Σ^0_2 formula, which is impossible.
- So this family is not $BC_{co-r.e.}$ learnable. □

Summary of Conclusions

- Learning using co-r.e. indices does seem to be a bit more restrictive under some circumstances.

Summary of Conclusions

- Learning using co-r.e. indices does seem to be a bit more restrictive under some circumstances.
- For example, vacillatory and explanatory learnability using co-r.e. indices coincide.

Summary of Conclusions

- Learning using co-r.e. indices does seem to be a bit more restrictive under some circumstances.
- For example, vacillatory and explanatory learnability using co-r.e. indices coincide.
- Similarly, for uniformly recursive families of sets, $BC_{co-r.e.}$ learnability is equivalent to $Cnsv_{co-r.e.}$ learnability.

Summary of Conclusions

- Learning using co-r.e. indices does seem to be a bit more restrictive under some circumstances.
- For example, vacillatory and explanatory learnability using co-r.e. indices coincide.
- Similarly, for uniformly recursive families of sets, $BC_{co-r.e.}$ learnability is equivalent to $Cnsv_{co-r.e.}$ learnability.
- This may be based on the observation that in general, if one can only enumerate counter-examples to the observed data, it may be a little more difficult to ascertain whether one has arrived at a correct conjecture.

Summary of Conclusions

- Learning using co-r.e. indices does seem to be a bit more restrictive under some circumstances.
- For example, vacillatory and explanatory learnability using co-r.e. indices coincide.
- Similarly, for uniformly recursive families of sets, $BC_{co-r.e.}$ learnability is equivalent to $Cnsv_{co-r.e.}$ learnability.
- This may be based on the observation that in general, if one can only enumerate counter-examples to the observed data, it may be a little more difficult to ascertain whether one has arrived at a correct conjecture.
- The following question is still unresolved: is there a uniformly co-r.e. class of recursive languages which is $Ex_{r.e.}$ learnable but not $BC_{co-r.e.}$ learnable?

Thank you.

Thank you.