

Learning Regular Languages over Large Alphabets

Oded Maler Irini Eleftheria Mens

CNRS-VERIMAG
University of Grenoble
2, avenue de Vignate
38610 Gieres
France

TACAS 2014

April 10, 2014

Outline

Learning Regular Languages

Learning regular languages

L* Algorithm

Queries

Observation tables

Symbolic Automata

Symbolic automata

Symbolic languages

Symbolic Learning

Evidences

Symbolic Algorithm

Example

Outline

Learning Regular Languages

Learning regular languages

L* Algorithm

Queries

Observation tables

Symbolic Automata

Symbolic automata

Symbolic languages

Symbolic Learning

Evidences

Symbolic Algorithm

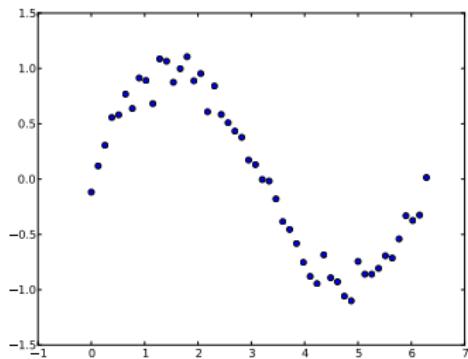
Example

Machine learning in general

- given $M = \{(x, y) \mid x \in X, y \in Y\}$
- find $f : X \rightarrow Y$ such that $f(x) = y$
- predict or identify $f(x)$ for all $x \in X$

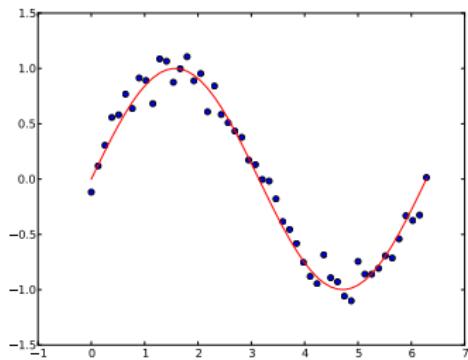
Machine learning in general

- given $M = \{(x, y) \mid x \in X, y \in Y\}$
- find $f : X \rightarrow Y$ such that $f(x) = y$
- predict or identify $f(x)$ for all $x \in X$



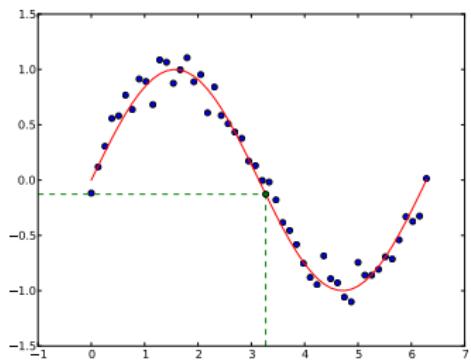
Machine learning in general

- given $M = \{(x, y) \mid x \in X, y \in Y\}$
- find $f : X \rightarrow Y$ such that $f(x) = y$
- predict or identify $f(x)$ for all $x \in X$



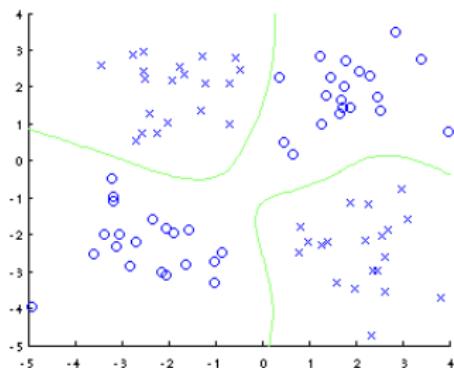
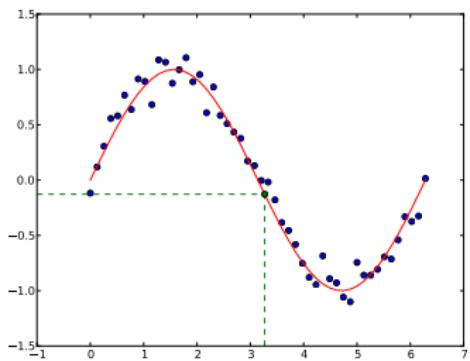
Machine learning in general

- given $M = \{(x, y) \mid x \in X, y \in Y\}$
- find $f : X \rightarrow Y$ such that $f(x) = y$
- predict or identify $f(x)$ for all $x \in X$



Machine learning in general

- given $M = \{(x, y) \mid x \in X, y \in Y\}$
- find $f : X \rightarrow Y$ such that $f(x) = y$
- predict or identify $f(x)$ for all $x \in X$



Learning
○●○○○○

Symbolic Automata
○○

Symbolic Learning
○○○

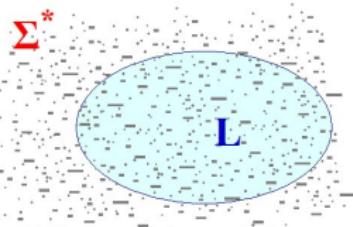
Conclusion
○

Learning regular languages

Learning regular languages

- Σ alphabet
- $L \subseteq \Sigma^*$ an unknown regular language (*target language*)

$$u \in \Sigma^* \longrightarrow \{+, -\}$$

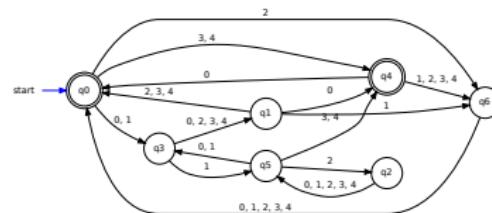
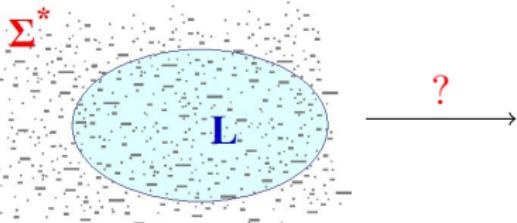


Learning regular languages

- Σ alphabet
- $L \subseteq \Sigma^*$ an unknown regular language (*target language*)

$$u \in \Sigma^* \longrightarrow \{+, -\}$$

Find automaton A , such that $L = L(A)$



Learning regular languages

- Edward F Moore, 1956
Gedanken-experiments on sequential machines
- E. Mark Gold, 1972
System Identification via State Characterization
- Dana Angluin, 1987
Learning regular sets from queries and counterexamples

Learning regular languages

- Edward F Moore, 1956
Gedanken-experiments on sequential machines
- E. Mark Gold, 1972
System Identification via State Characterization
- Dana Angluin, 1987
Learning regular sets from queries and counterexamples

- learning ω - languages [MP95]
- learning parameterized languages [BJR06]
- learning register automata [HSM11, HSJC12]

L^* Algorithm

- Active learning using two types of queries

L^* Algorithm

- Active learning using two types of queries

Membership Queries



L^* Algorithm

- Active learning using two types of queries

Membership Queries



Equivalence Queries



L^* Algorithm

- Active learning using two types of queries

Membership Queries



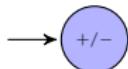
Equivalence Queries



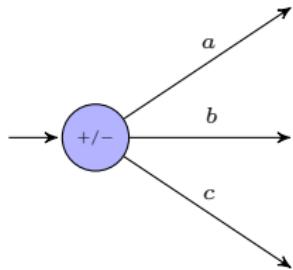
- Exactly learns the minimal DFA in polynomial time
 - size of automaton (number of states)
 - size of alphabet
 - length of counterexample

L^* Algorithm

- Initialize

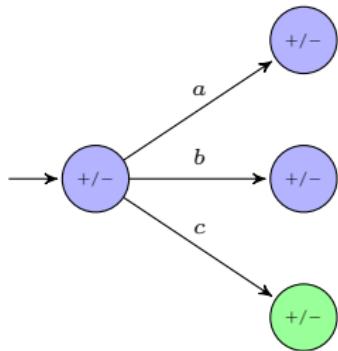


L^* Algorithm



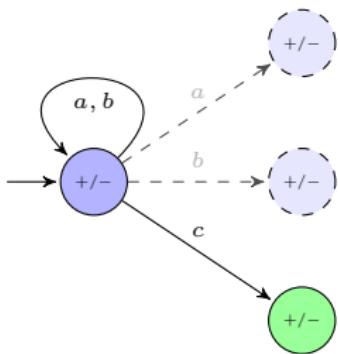
- Initialize
- Ask MQs

L^* Algorithm



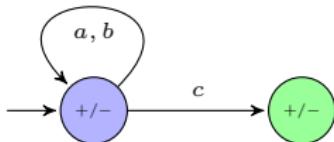
- Initialize
- Ask MQs

L^* Algorithm



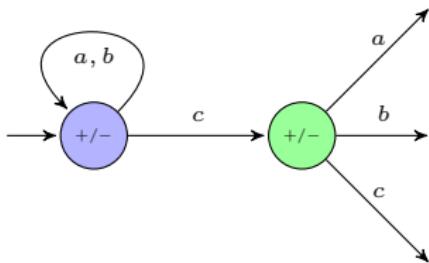
- Initialize
- Ask MQs

L^* Algorithm



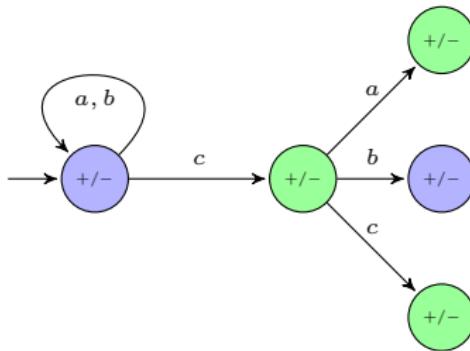
- Initialize
- Ask MQs

L^* Algorithm



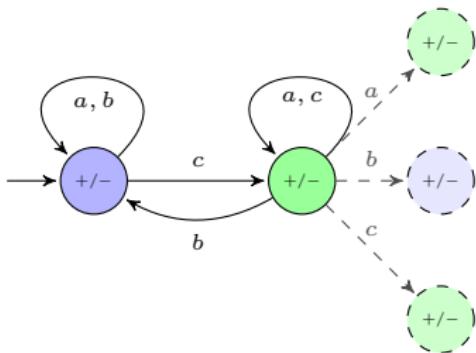
- Initialize
- Ask MQs

L^* Algorithm



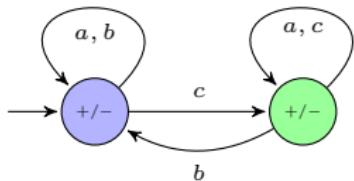
- Initialize
- Ask MQs

L^* Algorithm



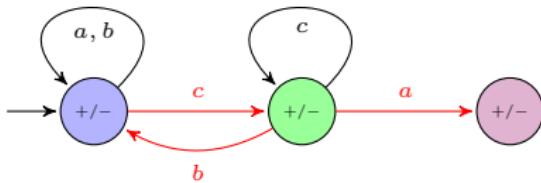
- Initialize
- Ask MQs

L^* Algorithm



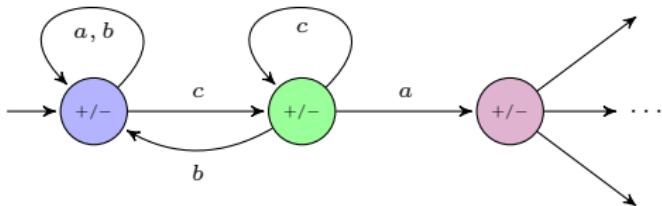
- Initialize
- Ask MQs
- When automaton is well defined
 - Ask EQ

L^* Algorithm



- Initialize
- Ask MQs
- When automaton is well defined
 - Ask EQ
- Use the counterexample to fix the automaton

L^* Algorithm



- Initialize
- Ask MQs
- When automaton is well defined
 - Ask EQ
- Use the counterexample to fix the automaton
- ...

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| <hr/> | | |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $\Sigma = \{a, b\}$

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $\Sigma = \{a, b\}$
- $S \subseteq \Sigma^*$ prefixes

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |

| | aa | ab |
|-------|------|------|
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $\Sigma = \{a, b\}$
- $S \subseteq \Sigma^*$ prefixes
- $R = S \cdot \Sigma \setminus S$ boundary

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |

| | aa | ab |
|-------|------|------|
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $\Sigma = \{a, b\}$
- $S \subseteq \Sigma^*$ prefixes
- $R = S \cdot \Sigma \setminus S$ boundary
- $E \subseteq \Sigma^*$ suffixes

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |

| | aa | ab |
|-------|------|------|
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $\Sigma = \{a, b\}$
- $S \subseteq \Sigma^*$ prefixes
- $R = S \cdot \Sigma \setminus S$ boundary
- $E \subseteq \Sigma^*$ suffixes
- $f : S \cup R \times E \rightarrow \{+, -\}$ classification function

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $\Sigma = \{a, b\}$
- $S \subseteq \Sigma^*$ prefixes
- $R = S \cdot \Sigma \setminus S$ boundary
- $E \subseteq \Sigma^*$ suffixes
- $f : S \cup R \times E \rightarrow \{+, -\}$ classification function

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $\Sigma = \{a, b\}$
- $S \subseteq \Sigma^*$ prefixes
- $R = S \cdot \Sigma \setminus S$ boundary
- $E \subseteq \Sigma^*$ suffixes
- $f : S \cup R \times E \rightarrow \{+, -\}$ classification function

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $\Sigma = \{a, b\}$
- $S \subseteq \Sigma^*$ prefixes
- $R = S \cdot \Sigma \setminus S$ boundary
- $E \subseteq \Sigma^*$ suffixes
- $f : S \cup R \times E \rightarrow \{+, -\}$ classification function
- $f_s : E \rightarrow \{+, -\}$ for all $s \in S \cup R$
$$f_s(e) = f(s \cdot e)$$

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $\Sigma = \{a, b\}$
- $S \subseteq \Sigma^*$ prefixes
- $R = S \cdot \Sigma \setminus S$ boundary
- $E \subseteq \Sigma^*$ suffixes
- $f : S \cup R \times E \rightarrow \{+, -\}$ classification function
- $f_s : E \rightarrow \{+, -\}$ for all $s \in S \cup R$
$$f_s(e) = f(s \cdot e)$$

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $\Sigma = \{a, b\}$
- $S \subseteq \Sigma^*$ prefixes
- $R = S \cdot \Sigma \setminus S$ boundary
- $E \subseteq \Sigma^*$ suffixes
- $f : S \cup R \times E \rightarrow \{+, -\}$ classification function
- $f_s : E \rightarrow \{+, -\}$ for all $s \in S \cup R$
$$f_s(e) = f(s \cdot e)$$

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

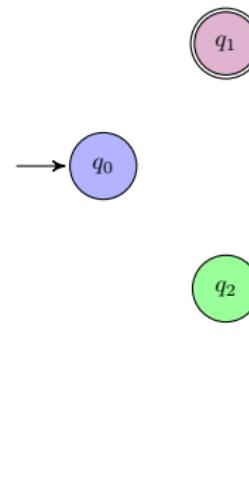
- $\Sigma = \{a, b\}$
- $S \subseteq \Sigma^*$ prefixes
- $R = S \cdot \Sigma \setminus S$ boundary
- $E \subseteq \Sigma^*$ suffixes
- $f : S \cup R \times E \rightarrow \{+, -\}$ classification function
- $f_s : E \rightarrow \{+, -\}$ for all $s \in S \cup R$
$$f_s(e) = f(s \cdot e)$$

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| | | |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

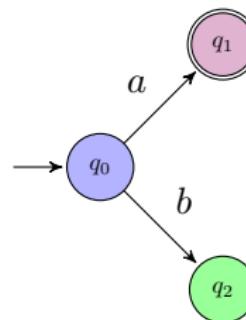
Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| | | |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |



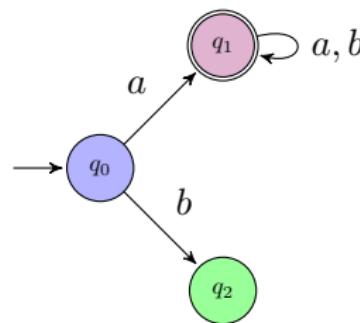
Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |



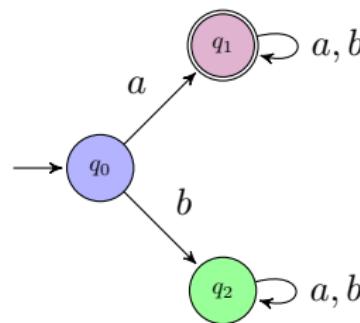
Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |



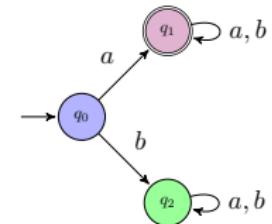
Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |



Observation Table - $T = (\Sigma, S, R, E, f)$

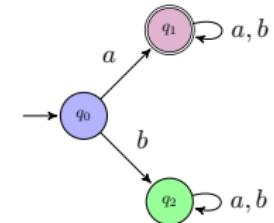
| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |



Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| | | |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

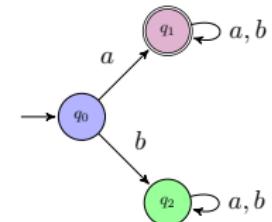
- $S \cup R$ is prefix-closed



Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| | | |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $S \cup R$ is prefix-closed
- E is suffix-closed



Observation Table - $T = (\Sigma, S, R, E, f)$

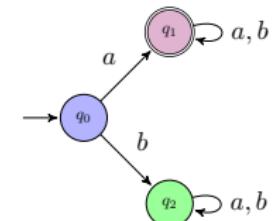
| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| | | |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | - | - |

- $S \cup R$ is prefix-closed

- E is suffix-closed

- T closed

$$\forall r \in R, \exists s \in S, f_r = f_s$$



Observation Table - $T = (\Sigma, S, R, E, f)$

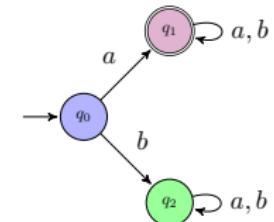
| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| bab | + | - |

- $S \cup R$ is prefix-closed

- E is suffix-closed

- T closed

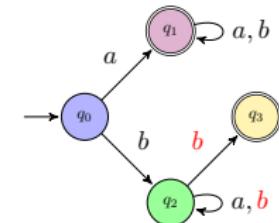
$$\forall r \in R, \exists s \in S, f_r = f_s$$



Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| bab | + | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| : | : | : |

- $S \cup R$ is prefix-closed
- E is suffix-closed

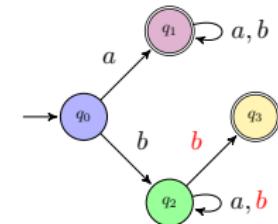


- T closed
 $\forall r \in R, \exists s \in S, f_r = f_s$

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| bab | + | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| : | : | : |

- $S \cup R$ is prefix-closed
- E is suffix-closed

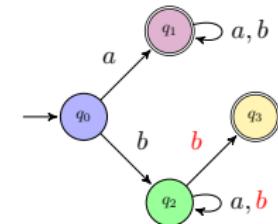


- T closed
 $\forall r \in R, \exists s \in S, f_r = f_s$
- T consistent
 $\forall s, s' \in S, \forall a \in \Sigma, f_s = f_{s'} \Rightarrow f_{s \cdot a} = f_{s' \cdot a}$

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| bab | + | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| : | : | : |

- $S \cup R$ is prefix-closed
- E is suffix-closed

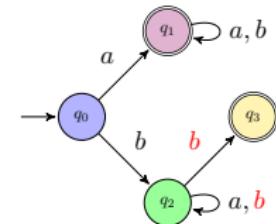


- T closed
 $\forall r \in R, \exists s \in S, f_r = f_s$
- T consistent
 $\forall s, s' \in S, \forall a \in \Sigma, f_s = f_{s'} \Rightarrow f_{s \cdot a} = f_{s' \cdot a}$

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | + |
| b | - | - |
| ba | - | - |
| bab | + | - |
| aa | + | + |
| ab | + | + |
| bb | - | - |
| baa | - | - |
| : | : | : |

- $S \cup R$ is prefix-closed
- E is suffix-closed

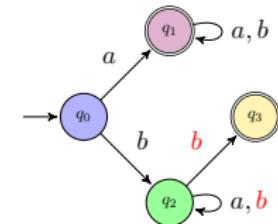


- T closed
 $\forall r \in R, \exists s \in S, f_r = f_s$
- T consistent
 $\forall s, s' \in S, \forall a \in \Sigma, f_s = f_{s'} \Rightarrow f_{s \cdot a} = f_{s' \cdot a}$

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a | b |
|------------|------------|------|------|
| ϵ | — | + | |
| a | + | + | |
| b | — | — | |
| ba | — | — | |
| bab | + | — | |
| | | | |
| aa | + | + | |
| ab | + | + | |
| bb | — | — | |
| baa | — | — | |
| : | : | : | |

- $S \cup R$ is prefix-closed
- E is suffix-closed

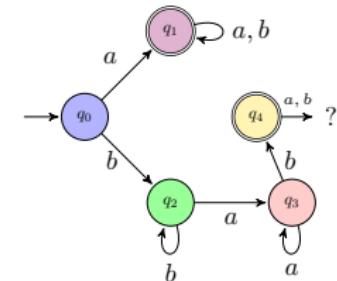


- T closed
 $\forall r \in R, \exists s \in S, f_r = f_s$
- T consistent
 $\forall s, s' \in S, \forall a \in \Sigma, f_s = f_{s'} \Rightarrow f_{s \cdot a} = f_{s' \cdot a}$

Observation Table - $T = (\Sigma, S, R, E, f)$

| | ϵ | a | b |
|------------|------------|------|------|
| ϵ | - | + | : |
| a | + | + | : |
| b | - | - | - |
| ba | - | - | + |
| bab | + | - | : |
| | | | |
| aa | + | + | : |
| ab | + | + | : |
| bb | - | - | : |
| baa | - | - | : |
| : | : | : | : |

- $S \cup R$ is prefix-closed
- E is suffix-closed



- T closed
 $\forall r \in R, \exists s \in S, f_r = f_s$
- T consistent
 $\forall s, s' \in S, \forall a \in \Sigma, f_s = f_{s'} \Rightarrow f_{s \cdot a} = f_{s' \cdot a}$

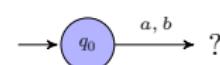
L^* Example ($\Sigma = \{a, b\}$)

L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | |
| b | |

hypothesis automaton

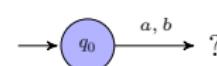


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | - |
| a | + |
| b | - |

hypothesis automaton

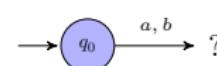


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | - |
| a | + |
| b | - |

hypothesis automaton

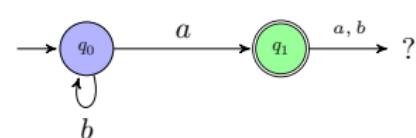


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | + |
| b | — |

hypothesis automaton

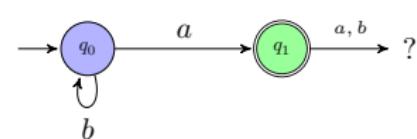


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | + |
| b | — |
| aa | |
| ab | |

hypothesis automaton

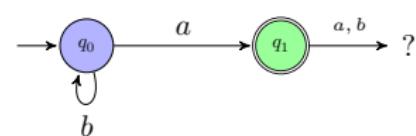


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | + |
| b | — |
| aa | — |
| ab | + |

hypothesis automaton

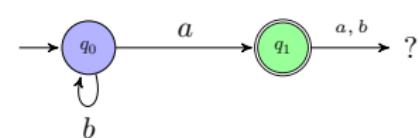


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | + |
| b | — |
| aa | — |
| ab | + |

hypothesis automaton

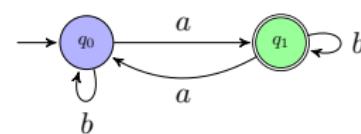


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | - |
| a | + |
| b | - |
| aa | - |
| ab | + |

hypothesis automaton

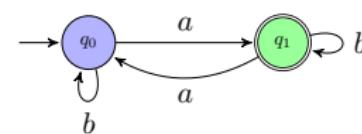


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | + |
| b | — |
| aa | — |
| ab | + |

hypothesis automaton



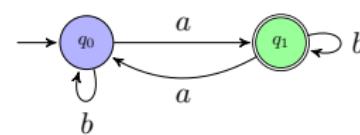
Ask Equivalence Query:
counterexample: **-ba**

L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | + |
| ba | — |
| b | — |
| aa | — |
| ab | + |

hypothesis automaton

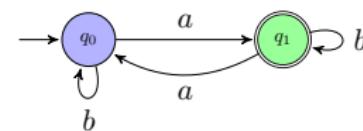
counterexample: $-ba$

L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | + |
| b | — |
| ba | — |
| | |
| aa | — |
| ab | + |

hypothesis automaton

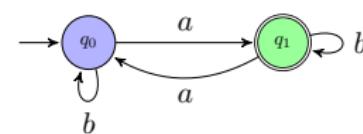
counterexample: $-ba$

L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | + |
| b | — |
| ba | — |
| | |
| aa | — |
| ab | + |
| bb | |
| baa | |
| bab | |

hypothesis automaton

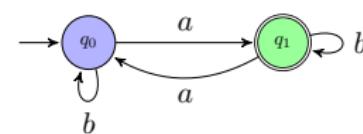


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | + |
| b | — |
| ba | — |
| | |
| aa | — |
| ab | + |
| bb | + |
| baa | — |
| bab | + |

hypothesis automaton



L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | + |
| b | — |
| ba | — |

| | aa |
|-------|------|
| aa | — |
| ab | + |
| bb | + |
| baa | — |
| bab | + |

hypothesis automaton

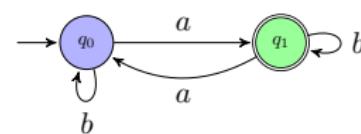


Table is not consistent!

L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a | + |
| b | — |
| ba | — |

| | aa |
|-------|------|
| aa | — |
| ab | + |
| bb | + |
| baa | — |
| bab | + |

hypothesis automaton

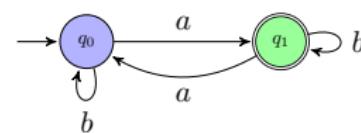


Table is not consistent!

L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ | a |
|------------|------------|-----|
| ϵ | — | |
| a | + | |
| b | — | |
| ba | — | |
| <hr/> | | |
| aa | — | |
| ab | + | |
| bb | + | |
| baa | — | |
| bab | + | |

hypothesis automaton

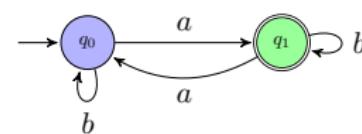


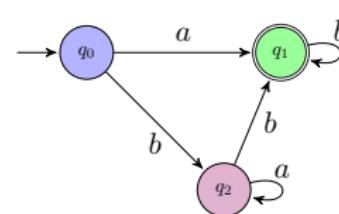
Table is not consistent!

L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ | a |
|------------|------------|------|
| ϵ | — | + |
| a | + | — |
| b | — | — |
| ba | — | — |
| | | |
| aa | — | |
| ab | + | |
| bb | + | |
| baa | — | |
| bab | + | |

hypothesis automaton

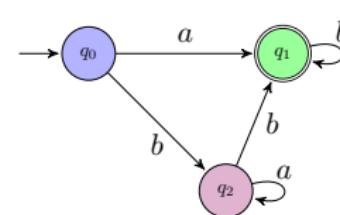


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ | a |
|------------|------------|------|
| ϵ | - | + |
| a | + | - |
| b | - | - |
| ba | - | - |
| | | |
| aa | - | + |
| ab | + | - |
| bb | + | - |
| baa | - | - |
| bab | + | - |

hypothesis automaton

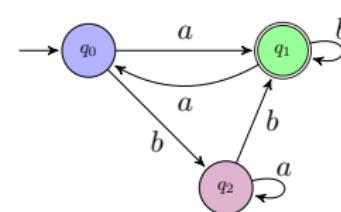


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ | a |
|------------|------------|-----|
| ϵ | — | + |
| a | + | — |
| b | — | — |
| ba | — | — |
| | | |
| aa | — | + |
| ab | + | — |
| bb | + | — |
| baa | — | — |
| bab | + | — |

hypothesis automaton

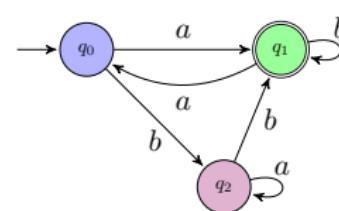


L^* Example ($\Sigma = \{a, b\}$)

observation table

| | ϵ | a |
|------------|------------|-----|
| ϵ | - | + |
| a | + | - |
| b | - | - |
| ba | - | - |
| <hr/> | | |
| aa | - | + |
| ab | + | - |
| bb | + | - |
| baa | - | - |
| bab | + | - |

hypothesis automaton



Ask Equivalence Query:

True

Outline

Learning Regular Languages

Learning regular languages

L* Algorithm

Queries

Observation tables

Symbolic Automata

Symbolic automata

Symbolic languages

Symbolic Learning

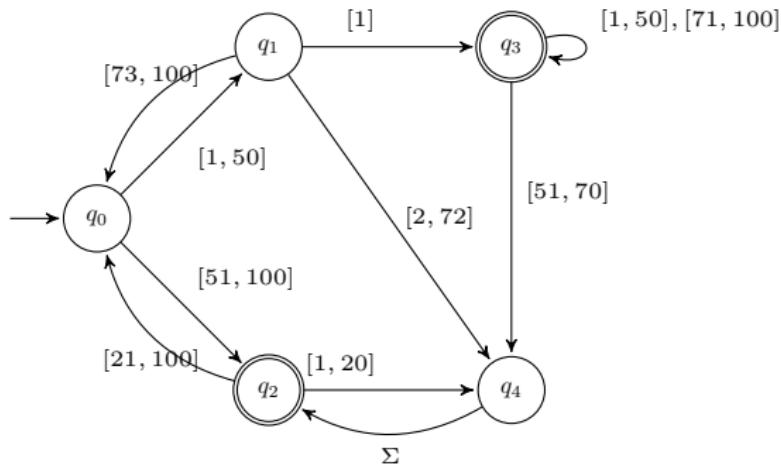
Evidences

Symbolic Algorithm

Example

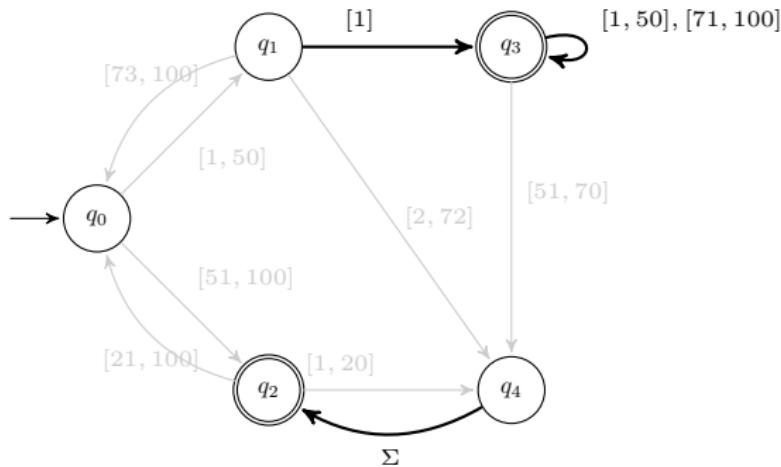
Deterministic Symbolic Automaton

$$\Sigma = [1, 100] \subseteq \mathbb{N}$$



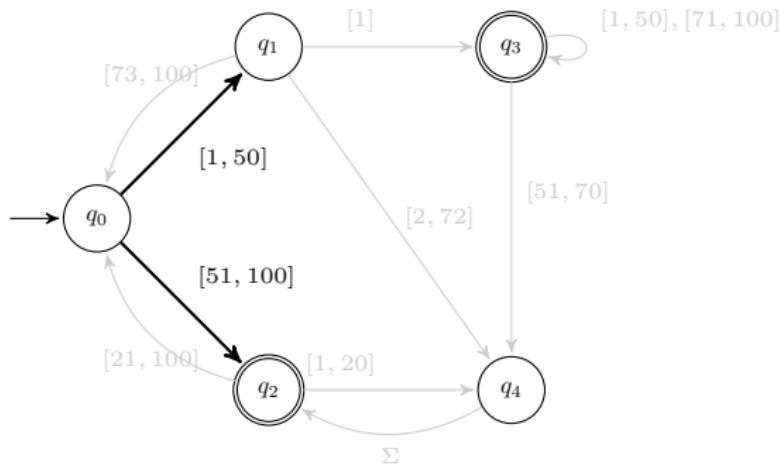
Deterministic Symbolic Automaton

$$\Sigma = [1, 100] \subseteq \mathbb{N}$$



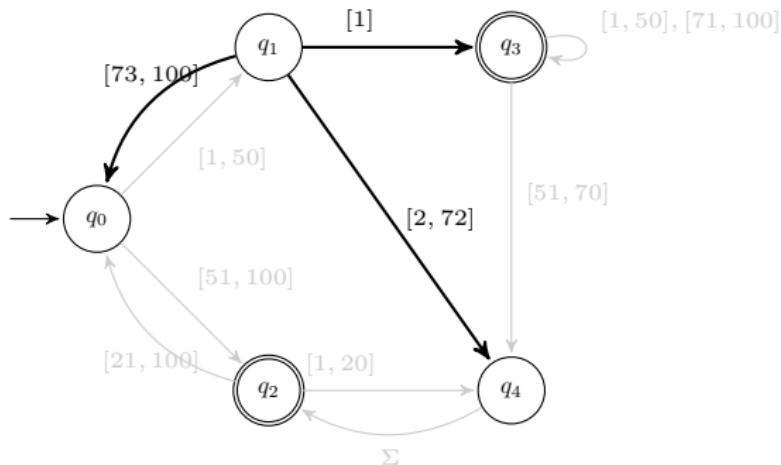
Deterministic Symbolic Automaton

$$\Sigma = [1, 100] \subseteq \mathbb{N}$$



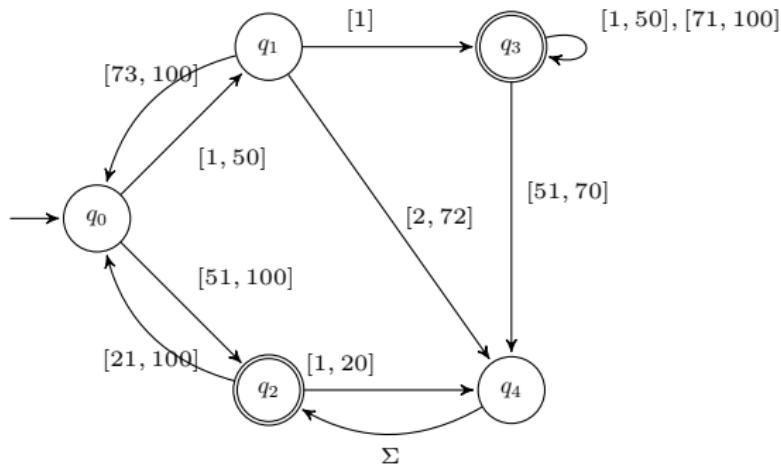
Deterministic Symbolic Automaton

$$\Sigma = [1, 100] \subseteq \mathbb{N}$$



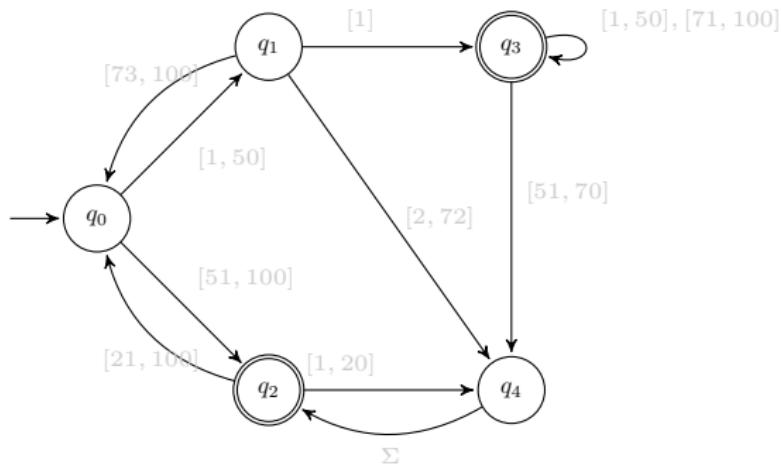
Deterministic Symbolic Automaton

$$\Sigma = [1, 100] \subseteq \mathbb{N}$$



Deterministic Symbolic Automaton

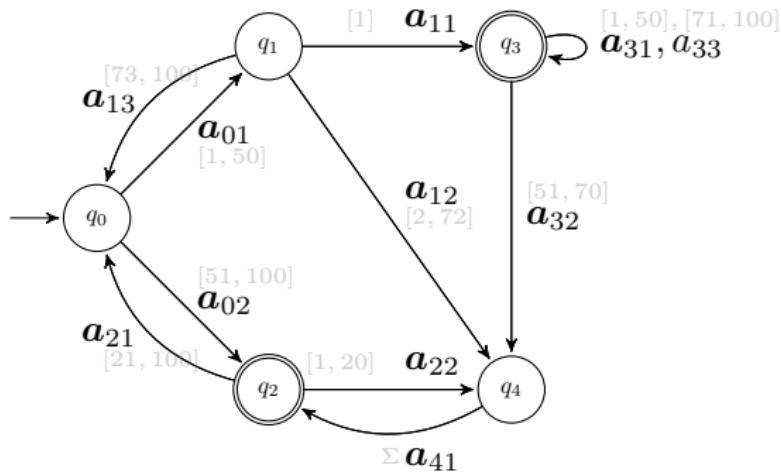
$$\Sigma = [1, 100] \subseteq \mathbb{N}$$



Deterministic Symbolic Automaton

$$\Sigma = [1, 100] \subseteq \mathbb{N}$$

$$\begin{aligned}\mathbf{a}_{01} &= [1, 50] \\ \mathbf{a}_{02} &= [51, 100] \\ \mathbf{a}_{11} &= \dots \\ &\dots\end{aligned}$$



Deterministic Symbolic Automaton

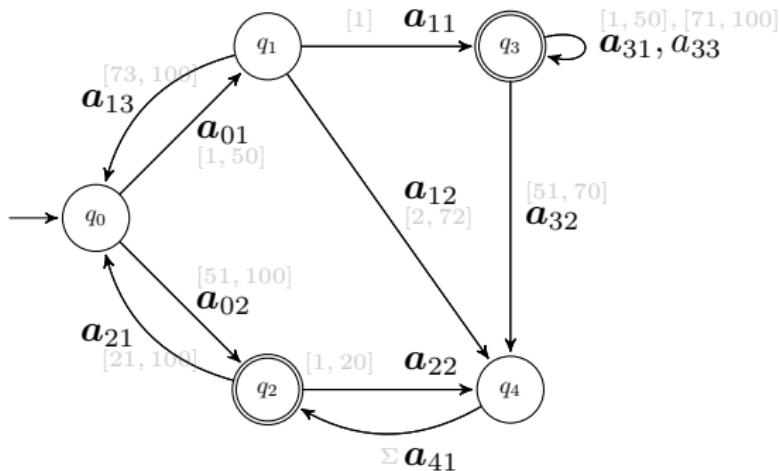
$$\Sigma = [1, 100] \subseteq \mathbb{N}$$

$$\begin{aligned}\mathbf{a}_{01} &= [1, 50] \\ \mathbf{a}_{02} &= [51, 100]\end{aligned}$$

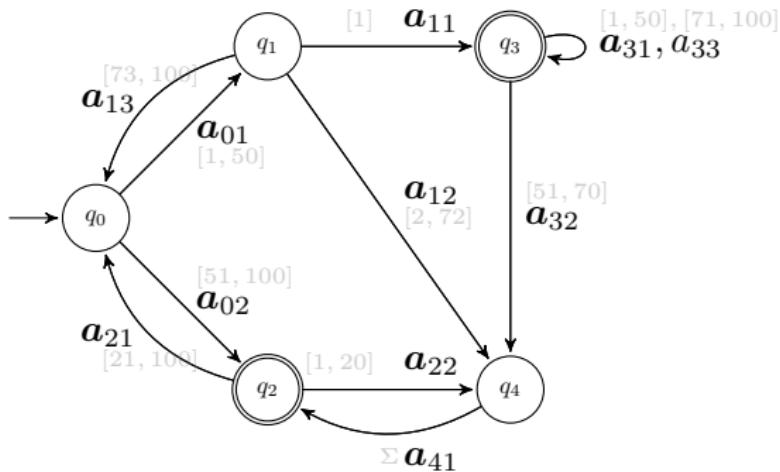
$$\mathbf{a}_{11} = \dots$$

...

$$\Sigma = \{\mathbf{a}_{01}, \mathbf{a}_{02}, \dots\}$$



Deterministic Symbolic Automaton



$$\Sigma = [1, 100] \subseteq \mathbb{N}$$

$$a_{01} = [1, 50]$$

$$a_{02} = [51, 100]$$

$$a_{11} = \dots$$

...

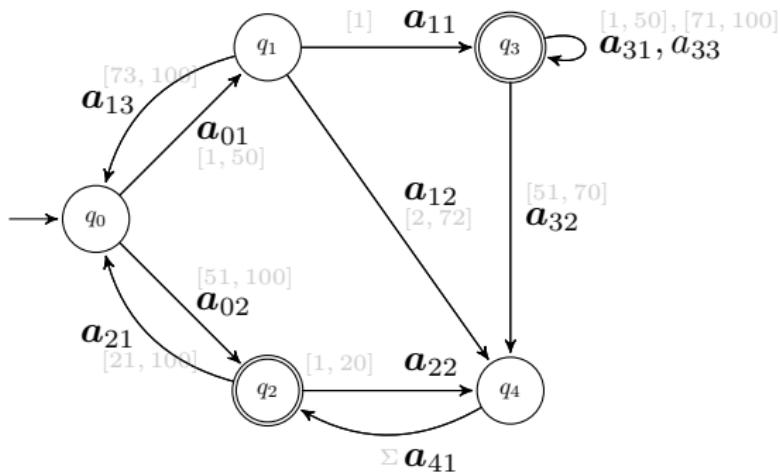
$$\Sigma = \{a_{01}, a_{02}, \dots\}$$

$$\Sigma_{q_0} = \{a_{01}, a_{02}\}$$

$$\Sigma_{q_1} = \{a_{11}, a_{12}, a_{13}\}$$

...

Deterministic Symbolic Automaton



$$\Sigma = [1, 100] \subseteq \mathbb{N}$$

$$a_{01} = [1, 50]$$

$$a_{02} = [51, 100]$$

$$a_{11} = \dots$$

...

$$\Sigma = \{a_{01}, a_{02}, \dots\}$$

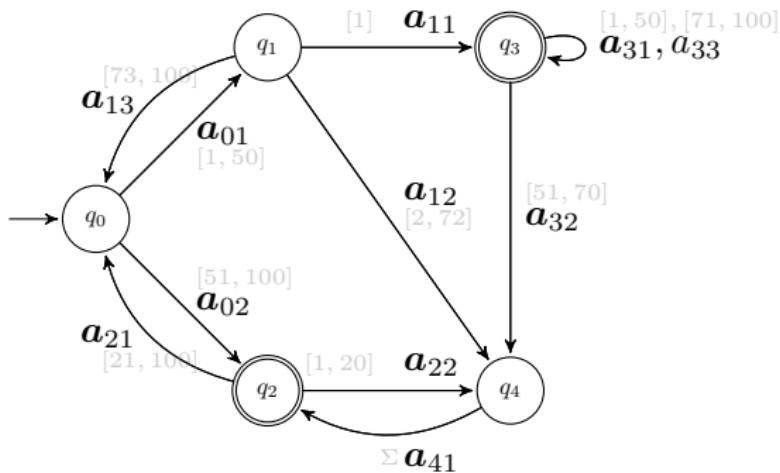
$$\Sigma_{q_0} = \{a_{01}, a_{02}\}$$

$$\Sigma_{q_1} = \{a_{11}, a_{12}, a_{13}\}$$

...

$$\biguplus_{a \in \Sigma_q} a = \Sigma$$

Deterministic Symbolic Automaton



$$\Sigma = [1, 100] \subseteq \mathbb{N}$$

$$a_{01} = [1, 50]$$

$$a_{02} = [51, 100]$$

$$a_{11} = \dots$$

...

$$\Sigma = \{a_{01}, a_{02}, \dots\}$$

$$\Sigma_{q_0} = \{a_{01}, a_{02}\}$$

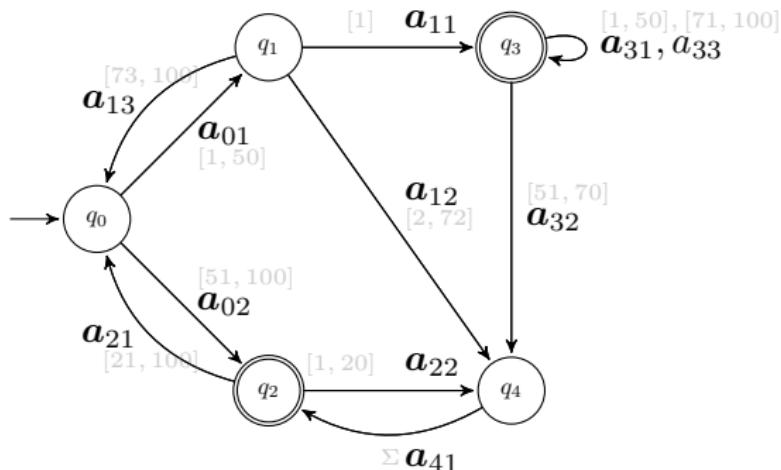
$$\Sigma_{q_1} = \{a_{11}, a_{12}, a_{13}\}$$

...

$$\biguplus_{a \in \Sigma_q} a = \Sigma$$

$$\psi_q : \Sigma \rightarrow \Sigma_q, \forall q \in Q$$

Deterministic Symbolic Automaton



$$\Sigma = [1, 100] \subseteq \mathbb{N}$$

$$a_{01} = [1, 50]$$

$$a_{02} = [51, 100]$$

$$a_{11} = \dots$$

...

$$\Sigma = \{a_{01}, a_{02}, \dots\}$$

$$\Sigma_{q_0} = \{a_{01}, a_{02}\}$$

$$\Sigma_{q_1} = \{a_{11}, a_{12}, a_{13}\}$$

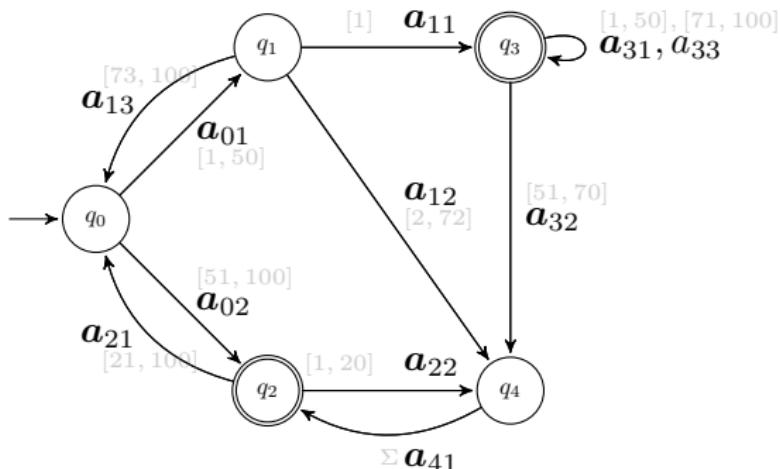
...

$$\biguplus_{a \in \Sigma_q} a = \Sigma$$

$$\psi_q : \Sigma \rightarrow \Sigma_q, \forall q \in Q$$

$$\psi_{q_1}(10) = a_{12}$$

Deterministic Symbolic Automaton



$$\Sigma = [1, 100] \subseteq \mathbb{N}$$

$$a_{01} = [1, 50]$$

$$a_{02} = [51, 100]$$

$$a_{11} = \dots$$

...

$$\Sigma = \{a_{01}, a_{02}, \dots\}$$

$$\Sigma_{q_0} = \{a_{01}, a_{02}\}$$

$$\Sigma_{q_1} = \{a_{11}, a_{12}, a_{13}\}$$

...

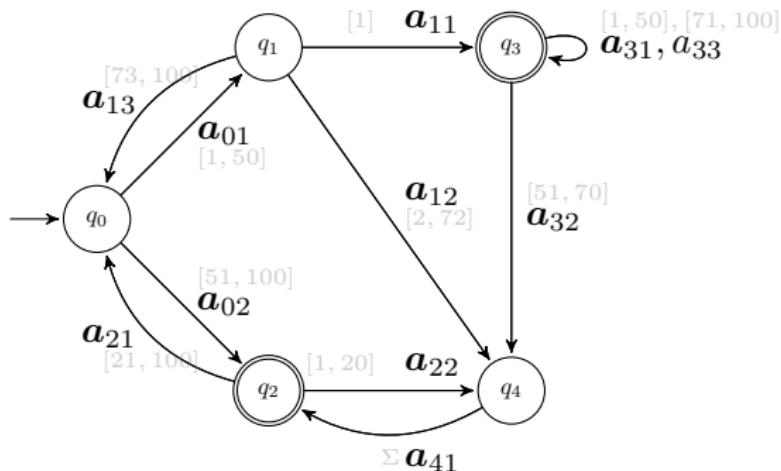
$$\biguplus_{a \in \Sigma_q} a = \Sigma$$

$$\psi_q : \Sigma \rightarrow \Sigma_q, \forall q \in Q$$

$$\psi_{q_1}(10) = a_{12}$$

$$\delta : Q \times \Sigma \rightarrow Q$$

Deterministic Symbolic Automaton



$$\Sigma = [1, 100] \subseteq \mathbb{N}$$

$$a_{01} = [1, 50]$$

$$a_{02} = [51, 100]$$

$$a_{11} = \dots$$

...

$$\Sigma = \{a_{01}, a_{02}, \dots\}$$

$$\Sigma_{q_0} = \{a_{01}, a_{02}\}$$

$$\Sigma_{q_1} = \{a_{11}, a_{12}, a_{13}\}$$

...

$$\biguplus_{a \in \Sigma_q} a = \Sigma$$

$$\psi_q : \Sigma \rightarrow \Sigma_q, \forall q \in Q$$

$$\psi_{q_1}(10) = a_{12}$$

$$\delta : Q \times \Sigma \rightarrow Q$$

$$\delta(q, a) = q' \text{ iff } \delta(q, a) = q', a \in a$$

Deterministic Symbolic Automaton

$$\mathcal{A} = (\Sigma, \Sigma, \psi, Q, \delta, q_0, F)$$

Deterministic Symbolic Automaton

$\mathcal{A} = (\Sigma, \Sigma, \psi, Q, \delta, q_0, F)$, where

- Σ is the input alphabet,
- Q is a finite set of states,
- q_0 is the initial state,
- F is the set of accepting states,

Deterministic Symbolic Automaton

$\mathcal{A} = (\Sigma, \Sigma, \psi, Q, \delta, q_0, F)$, where

- Σ is the input alphabet,
- Q is a finite set of states,
- q_0 is the initial state,
- F is the set of accepting states,
- Σ is a finite alphabet, decomposable into $\Sigma = \biguplus_{q \in Q} \Sigma_q$,

Deterministic Symbolic Automaton

$\mathcal{A} = (\Sigma, \Sigma, \psi, Q, \delta, q_0, F)$, where

- Σ is the input alphabet,
- Q is a finite set of states,
- q_0 is the initial state,
- F is the set of accepting states,
- Σ is a finite alphabet, decomposable into $\Sigma = \biguplus_{q \in Q} \Sigma_q$,
- $\psi = \{\psi_q : q \in Q\}$ is a family of total surjective functions $\psi_q : \Sigma \rightarrow \Sigma_q$,

Deterministic Symbolic Automaton

$\mathcal{A} = (\Sigma, \Sigma, \psi, Q, \delta, q_0, F)$, where

- Σ is the input alphabet,
- Q is a finite set of states,
- q_0 is the initial state,
- F is the set of accepting states,
- Σ is a finite alphabet, decomposable into $\Sigma = \biguplus_{q \in Q} \Sigma_q$,
- $\psi = \{\psi_q : q \in Q\}$ is a family of total surjective functions $\psi_q : \Sigma \rightarrow \Sigma_q$,
- $\delta : Q \times \Sigma \rightarrow Q$ is a partial transition function decomposable into a family of total functions $\delta_q : \{q\} \times \Sigma_q \rightarrow Q$,

Deterministic Symbolic Automaton

$\mathcal{A} = (\Sigma, \Sigma, \psi, Q, \delta, q_0, F)$, where

- Σ is the input alphabet,
- Q is a finite set of states,
- q_0 is the initial state,
- F is the set of accepting states,
- Σ is a finite alphabet, decomposable into $\Sigma = \biguplus_{q \in Q} \Sigma_q$,
- $\psi = \{\psi_q : q \in Q\}$ is a family of total surjective functions $\psi_q : \Sigma \rightarrow \Sigma_q$,
- $\delta : Q \times \Sigma \rightarrow Q$ is a partial transition function decomposable into a family of total functions $\delta_q : \{q\} \times \Sigma_q \rightarrow Q$,

$w = a_1 a_2 \dots a_n$ accepted

$$\delta^*(q_0, a_1 a_2 \dots a_n) = \delta^*(\delta(q_0, \psi_q(a_1)), a_2 \dots a_n) \in F$$

Deterministic Symbolic Automaton

$\mathcal{A} = (\Sigma, \Sigma, \psi, Q, \delta, q_0, F)$, where

- Σ is the input alphabet,
- Q is a finite set of states,
- q_0 is the initial state,
- F is the set of accepting states,
- Σ is a finite alphabet, decomposable into $\Sigma = \biguplus_{q \in Q} \Sigma_q$,
- $\psi = \{\psi_q : q \in Q\}$ is a family of total surjective functions $\psi_q : \Sigma \rightarrow \Sigma_q$,
- $\delta : Q \times \Sigma \rightarrow Q$ is a partial transition function decomposable into a family of total functions $\delta_q : \{q\} \times \Sigma_q \rightarrow Q$,

$$L(A) = \{w \in \Sigma^* | \delta^*(q_0, w) \in F\}$$

$w = a_1 a_2 \dots a_n$ accepted

$$\delta^*(q_0, a_1 a_2 \dots a_n) = \delta^*(\delta(q_0, \psi_q(a_1)), a_2 \dots a_n) \in F$$

Outline

Learning Regular Languages

Learning regular languages

L* Algorithm

Queries

Observation tables

Symbolic Automata

Symbolic automata

Symbolic languages

Symbolic Learning

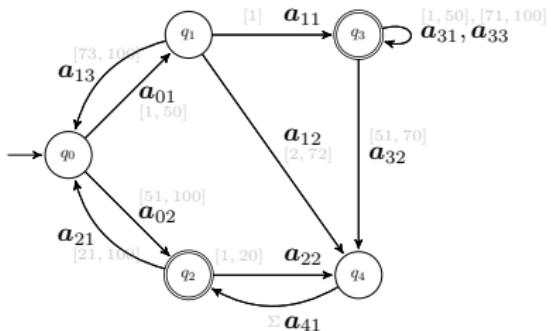
Evidences

Symbolic Algorithm

Example

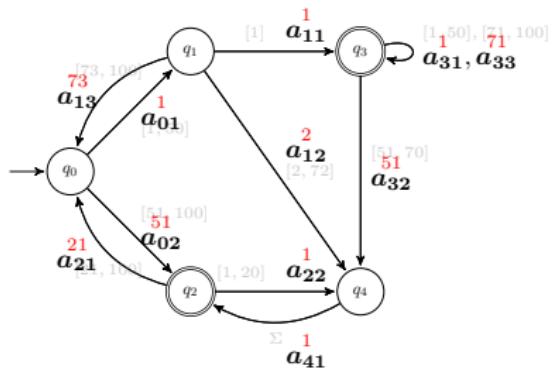
Symbolic Algorithm

Evidences



Symbolic Algorithm

Evidences

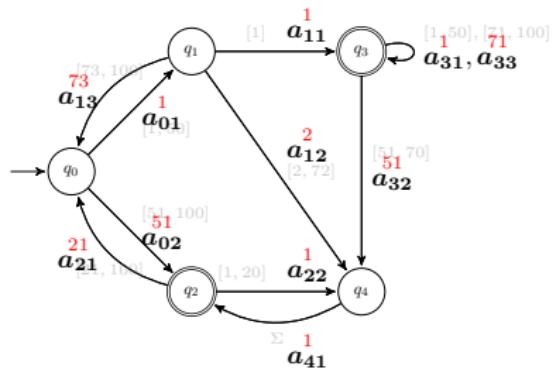


$$\mu : \Sigma \rightarrow \Sigma$$

Symbolic Algorithm

Evidences

Counterexample



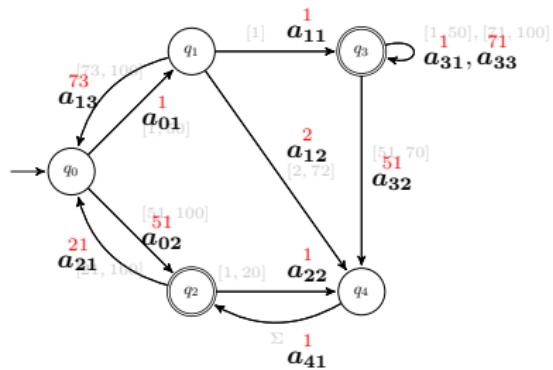
$$\mu : \Sigma \rightarrow \Sigma$$

minimal

- length
- lexicographically

Symbolic Algorithm

Evidences



$$\mu : \Sigma \rightarrow \Sigma$$
$$\mu(\mathbf{a}) = \min\{a \in \Sigma \mid \psi(a) = \mathbf{a}\}$$

$$\mu(\mathbf{a}_{01}) = 1$$

$$\mu(\mathbf{a}_{02}) = 51$$

...

$$\mu(\mathbf{a}) \subseteq [\mathbf{a}]$$

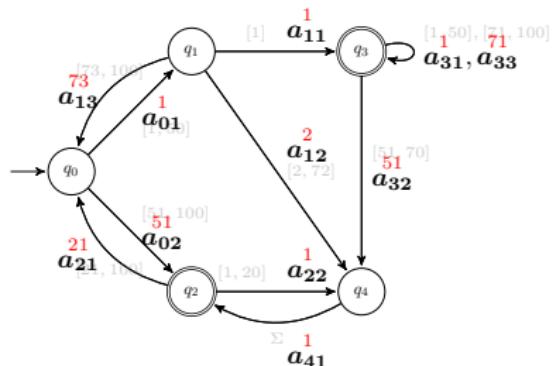
Counterexample

minimal

- length
- lexicographically

Symbolic Algorithm

Evidences



$$\mu : \Sigma \rightarrow \Sigma$$

$$\mu(\mathbf{a}) = \min\{a \in \Sigma \mid \psi(a) = \mathbf{a}\}$$

$$\mu(\mathbf{a}_{01}) = 1$$

$$\mu(\mathbf{a}_{02}) = 51$$

...

$$\mu(\mathbf{a}) \subseteq [\mathbf{a}]$$

Counterexample

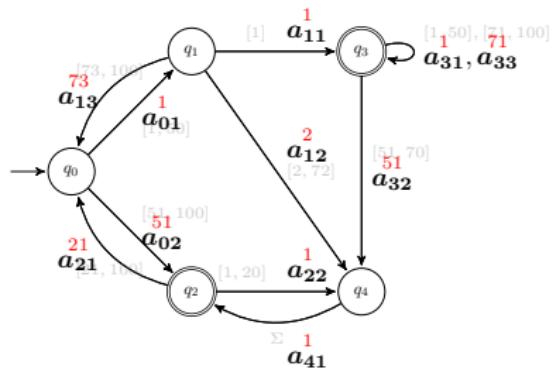
minimal

- length
- lexicographically

- Discover new states
- Refine partitions

Symbolic Algorithm

Evidences



$$\mu : \Sigma \rightarrow \Sigma$$

$$\mu(\mathbf{a}) = \min\{a \in \Sigma \mid \psi(a) = \mathbf{a}\}$$

$$\mu(a_{01}) = 1$$

$$\mu(a_{02}) = 51$$

...

$$\mu(\mathbf{a}) \subseteq [\mathbf{a}]$$

Counterexample

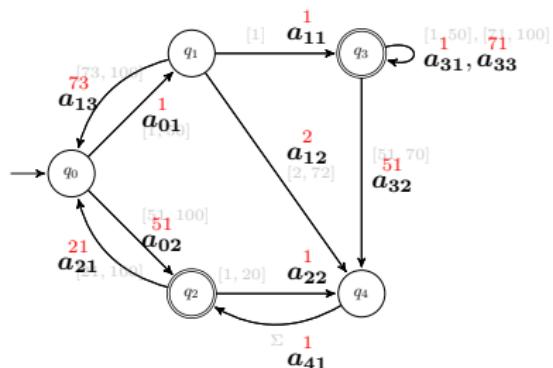
minimal

- length
- lexicographically

- Discover new states
- Refine partitions
 - Split a symbolic letter
 - Move letters from one symbolic letter to another

Symbolic Algorithm

Evidences



$$\mu : \Sigma \rightarrow \Sigma$$

$$\mu(\mathbf{a}) = \min\{a \in \Sigma \mid \psi(a) = \mathbf{a}\}$$

$$\mu(\mathbf{a}_{01}) = 1$$

$$\mu(\mathbf{a}_{02}) = 51$$

...

$$\mu(\mathbf{a}) \subseteq [\mathbf{a}]$$

Counterexample

minimal

- length
- lexicographically

- Discover new states
- Refine partitions
 - Split a symbolic letter
 - Move letters from one symbolic letter to another

All evidences of the same symbolic letter behave the same
evidence compatibility

Symbolic algorithm

Repeat

- Create symbolic letters for all states and set evidences
- Ask MQs to complete the observation table
- Find observation table that is closed, consistent and evidence compatible
- Ask EQ
- Use the counterexample to update the table
 - Discover new state (vertical expansion)
 - Refine symbolic letters (horizontal expansion)

Example ($\Sigma = [1, 100]$)

observation table

$$\psi = \{\psi_s\}_{s \in S}$$

hypothesis automaton

| | |
|------------|------------|
| | ϵ |
| ϵ | |
| | |

Example ($\Sigma = [1, 100]$)

observation table

$$\psi = \{\psi_s\}_{s \in S}$$

hypothesis automaton

| | |
|------------|------------|
| | ϵ |
| ϵ | — |
| | |

$$\psi_\epsilon$$

Example ($\Sigma = [1, 100]$)

observation table

| | |
|------------|------------|
| | ϵ |
| ϵ | — |
| a_0 | 1 |

$$\psi = \{\psi_s\}_{s \in S}$$

hypothesis automaton

$$\begin{aligned}\psi_\epsilon \\ [\alpha_0] = [1, 100]\end{aligned}$$

Example ($\Sigma = [1, 100]$)

observation table

$$\psi = \{\psi_s\}_{s \in S}$$

hypothesis automaton

| | |
|------------|------------|
| | ϵ |
| ϵ | — |
| a_0 | + |

$$\begin{aligned}\psi_\epsilon \\ [\alpha_0] = [1, 100]\end{aligned}$$

Example ($\Sigma = [1, 100]$)

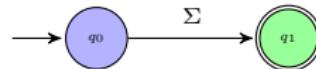
observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a_0 | + |

$$\psi = \{\psi_s\}_{s \in S}$$

$$\begin{aligned}\psi_\epsilon \\ [\alpha_0] = [1, 100]\end{aligned}$$

hypothesis automaton



Example ($\Sigma = [1, 100]$)

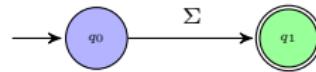
observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a_0 | + |

$$\psi = \{\psi_s\}_{s \in S}$$

$$\begin{aligned}\psi_\epsilon \\ [\alpha_0] = [1, 100] \\ \psi_{a_0}\end{aligned}$$

hypothesis automaton



Example ($\Sigma = [1, 100]$)

observation table

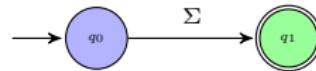
| | ϵ |
|------------|------------|
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | |

$$\psi = \{\psi_s\}_{s \in S}$$

$$\begin{aligned}\psi_\epsilon \\ [\alpha_0] = [1, 100]\end{aligned}$$

$$\begin{aligned}\psi_{a_0} \\ [\alpha_1] = [1, 100]\end{aligned}$$

hypothesis automaton



Example ($\Sigma = [1, 100]$)

observation table

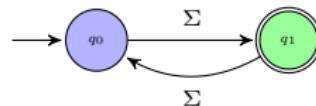
| | ϵ |
|------------|------------|
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |

$$\psi = \{\psi_s\}_{s \in S}$$

$$\begin{aligned}\psi_{\epsilon} \\ [\alpha_0] = [1, 100]\end{aligned}$$

$$\begin{aligned}\psi_{a_0} \\ [\alpha_1] = [1, 100]\end{aligned}$$

hypothesis automaton



Ask Equivalence Query:

Example ($\Sigma = [1, 100]$)

observation table

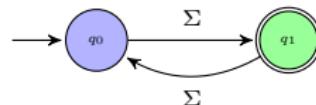
| | ϵ |
|------------|------------|
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |

$$\psi = \{\psi_s\}_{s \in S}$$

$$\begin{aligned}\psi_{\epsilon} \\ [\alpha_0] = [1, 100]\end{aligned}$$

$$\begin{aligned}\psi_{a_0} \\ [\alpha_1] = [1, 100]\end{aligned}$$

hypothesis automaton



Ask Equivalence Query:
counterexample — 24

Example ($\Sigma = [1, 100]$)

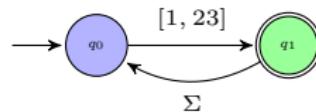
observation table

| | |
|------------|----------------|
| | ϵ |
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |
| a_2 | $\frac{1}{24}$ |

$\psi = \{\psi_s\}_{s \in S}$

$$\begin{aligned}\psi_\epsilon & \\ [\alpha_0] &= [1, 23] \\ [\alpha_2] &= [24, 100] \\ \psi_{a_0} & \\ [\alpha_1] &= [1, 100]\end{aligned}$$

hypothesis automaton



Ask Equivalence Query:
counterexample -24

Example ($\Sigma = [1, 100]$)

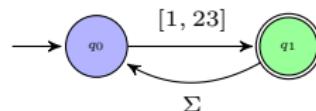
observation table

| | |
|------------|------------|
| | ϵ |
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |
| a_2 | — |

$\psi = \{\psi_s\}_{s \in S}$

$$\begin{aligned}\psi_\epsilon \\ [\alpha_0] &= [1, 23] \\ [\alpha_2] &= [24, 100] \\ \psi_{a_0} \\ [\alpha_1] &= [1, 100]\end{aligned}$$

hypothesis automaton



Ask Equivalence Query:
counterexample -24

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |
| a_2 | — |

$$\psi = \{\psi_s\}_{s \in S}$$

$$\psi_\epsilon$$

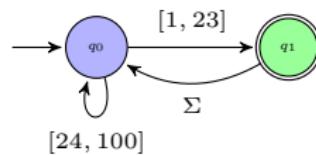
$$[a_0] = [1, 23]$$

$$[a_2] = [24, 100]$$

$$\psi_{a_0}$$

$$[a_1] = [1, 100]$$

hypothesis automaton



Ask Equivalence Query:

Example ($\Sigma = [1, 100]$)

observation table

| | |
|------------|------------|
| | ϵ |
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |
| a_2 | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

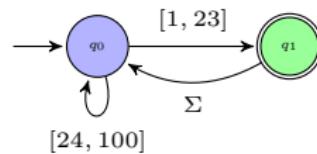
$$[a_0] = [1, 23]$$

$$[a_2] = [24, 100]$$

 ψ_{a_0}

$$[a_1] = [1, 100]$$

hypothesis automaton



Ask Equivalence Query:
 counterexample $+1 \cdot 66$

Example ($\Sigma = [1, 100]$)

observation table

| | |
|------------|------------|
| | ϵ |
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |
| a_2 | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

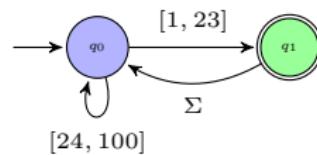
$$[a_0] = [1, 23]$$

$$[a_2] = [24, 100]$$

 ψ_{a_0}

$$[a_1] = [1, 100]$$

hypothesis automaton



Ask Equivalence Query:
 counterexample $+1 \cdot 66$

Example ($\Sigma = [1, 100]$)

observation table

| | |
|------------|------------|
| | ϵ |
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |
| a_2 | — |
| $a_0 a_3$ | — |

$\psi = \{\psi_s\}_{s \in S}$

 ψ_ϵ

$[a_0] = [1, 23]$

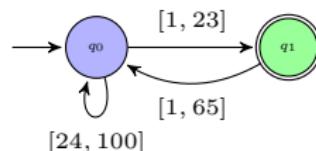
$[a_2] = [24, 100]$

 ψ_{a_0}

$[a_1] = [1, 65]$

$[a_3] = [66, 100]$

hypothesis automaton

Ask Equivalence Query:
counterexample $+1 \cdot 66$

Example ($\Sigma = [1, 100]$)

observation table

| | |
|------------|------------|
| | ϵ |
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |
| a_2 | — |
| $a_0 a_3$ | + |

$\psi = \{\psi_s\}_{s \in S}$

 ψ_ϵ

$[a_0] = [1, 23]$

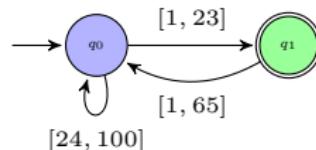
$[a_2] = [24, 100]$

 ψ_{a_0}

$[a_1] = [1, 65]$

$[a_3] = [66, 100]$

hypothesis automaton

Ask Equivalence Query:
counterexample $+1 \cdot 66$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |
| a_2 | — |
| $a_0 a_3$ | + |

$\psi = \{\psi_s\}_{s \in S}$

ψ_ϵ

$[a_0] = [1, 23]$

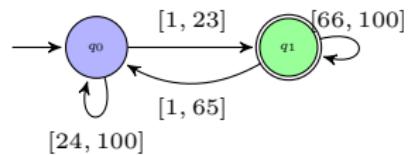
$[a_2] = [24, 100]$

ψ_{a_0}

$[a_1] = [1, 65]$

$[a_3] = [66, 100]$

hypothesis automaton



Ask Equivalence Query:

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |
| a_2 | — |
| $a_0 a_3$ | + |

$\psi = \{\psi_s\}_{s \in S}$

 ψ_ϵ

$[a_0] = [1, 23]$

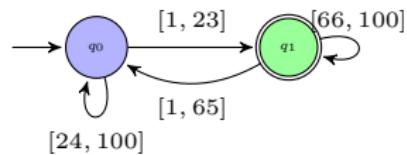
$[a_2] = [24, 100]$

 ψ_{a_0}

$[a_1] = [1, 65]$

$[a_3] = [66, 100]$

hypothesis automaton



Ask Equivalence Query:
 counterexample $-24 \cdot 1$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a_0 | + |
| $a_0 a_1$ | — |
| a_2 | — |
| $a_0 a_3$ | + |

$\psi = \{\psi_s\}_{s \in S}$

 ψ_ϵ

$[a_0] = [1, 23]$

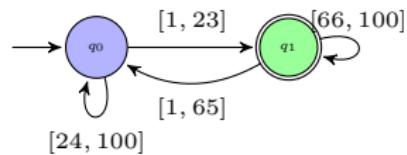
$[a_2] = [24, 100]$

 ψ_{a_0}

$[a_1] = [1, 65]$

$[a_3] = [66, 100]$

hypothesis automaton



Ask Equivalence Query:
 counterexample $-24 \cdot 1$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ |
|------------|------------|
| ϵ | — |
| a_0 | + |
| a_2 | — |
| $a_0 a_1$ | — |
| $a_0 a_3$ | + |

$\psi = \{\psi_s\}_{s \in S}$

ψ_ϵ

$[a_0] = [1, 23]$

$[a_2] = [24, 100]$

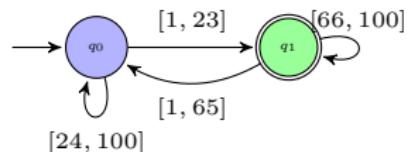
ψ_{a_0}

$[a_1] = [1, 65]$

$[a_3] = [66, 100]$

ψ_{a_2}

hypothesis automaton



Ask Equivalence Query:
 counterexample $-24 \cdot 1$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | |
| a_0 | + | |
| a_2 | — | |
| $a_0 a_1$ | — | |
| $a_0 a_3$ | + | |

$\psi = \{\psi_s\}_{s \in S}$

ψ_ϵ

$[a_0] = [1, 23]$

$[a_2] = [24, 100]$

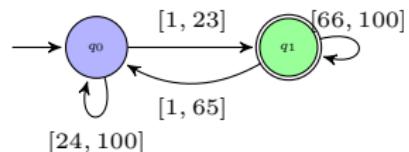
ψ_{a_0}

$[a_1] = [1, 65]$

$[a_3] = [66, 100]$

ψ_{a_2}

hypothesis automaton



Ask Equivalence Query:
 counterexample $-24 \cdot 1$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |

$\psi = \{\psi_s\}_{s \in S}$

 ψ_ϵ

$[a_0] = [1, 23]$

$[a_2] = [24, 100]$

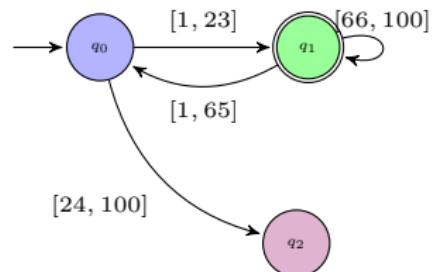
 ψ_{a_0}

$[a_1] = [1, 65]$

$[a_3] = [66, 100]$

 ψ_{a_2}

hypothesis automaton



Ask Equivalence Query:
 counterexample $-24 \cdot 1$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[a_0] = [1, 23]$$

$$[a_2] = [24, 100]$$

 ψ_{a_0}

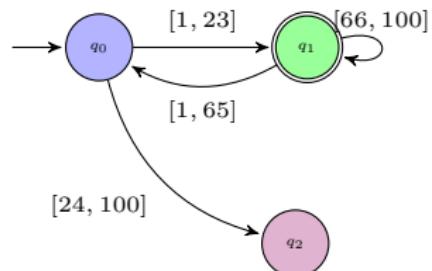
$$[a_1] = [1, 65]$$

$$[a_3] = [66, 100]$$

 ψ_{a_2}

$$[a_4] = [1, 100]$$

hypothesis automaton



Ask Equivalence Query:
 counterexample $-24 \cdot 1$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[\mathbf{a}_0] = [1, 23]$$

$$[\mathbf{a}_2] = [24, 100]$$

 ψ_{a_0}

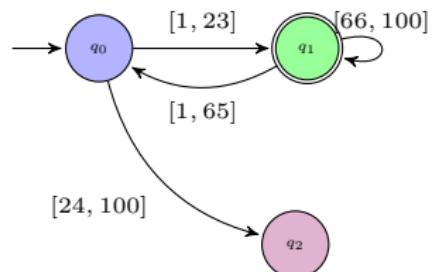
$$[\mathbf{a}_1] = [1, 65]$$

$$[\mathbf{a}_3] = [66, 100]$$

 ψ_{a_2}

$$[\mathbf{a}_4] = [1, 100]$$

hypothesis automaton



Ask Equivalence Query:
 counterexample $-24 \cdot 1$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[\mathbf{a}_0] = [1, 23]$$

$$[\mathbf{a}_2] = [24, 100]$$

 ψ_{a_0}

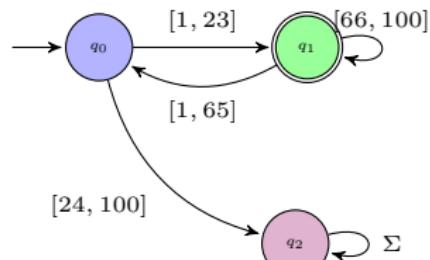
$$[\mathbf{a}_1] = [1, 65]$$

$$[\mathbf{a}_3] = [66, 100]$$

 ψ_{a_2}

$$[\mathbf{a}_4] = [1, 100]$$

hypothesis automaton



Ask Equivalence Query:

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[\mathbf{a}_0] = [1, 23]$$

$$[\mathbf{a}_2] = [24, 100]$$

 ψ_{a_0}

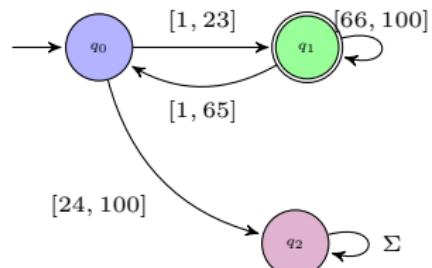
$$[\mathbf{a}_1] = [1, 65]$$

$$[\mathbf{a}_3] = [66, 100]$$

 ψ_{a_2}

$$[\mathbf{a}_4] = [1, 100]$$

hypothesis automaton



Ask Equivalence Query:
counterexample $+ 24 \cdot 51$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[\mathbf{a}_0] = [1, 23]$$

$$[\mathbf{a}_2] = [24, 100]$$

 ψ_{a_0}

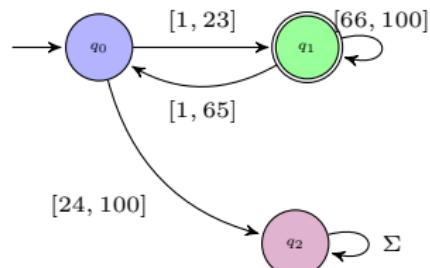
$$[\mathbf{a}_1] = [1, 65]$$

$$[\mathbf{a}_3] = [66, 100]$$

 ψ_{a_2}

$$[\mathbf{a}_4] = [1, 100]$$

hypothesis automaton



Ask Equivalence Query:
counterexample $+ 24 \cdot 51$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |
| $a_2 a_5$ | 24 51 | |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[\mathbf{a}_0] = [1, 23]$$

$$[\mathbf{a}_2] = [24, 100]$$

 ψ_{a_0}

$$[\mathbf{a}_1] = [1, 65]$$

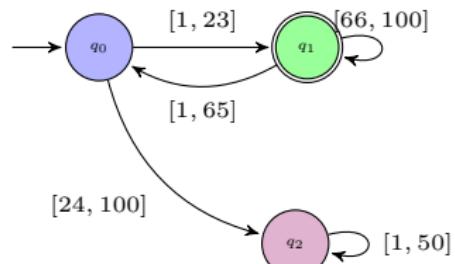
$$[\mathbf{a}_3] = [66, 100]$$

 ψ_{a_2}

$$[\mathbf{a}_4] = [1, 50]$$

$$[\mathbf{a}_5] = [51, 100]$$

hypothesis automaton



Ask Equivalence Query:
counterexample $+ 24 \cdot 51$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |
| $a_2 a_5$ | + | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[\mathbf{a}_0] = [1, 23]$$

$$[\mathbf{a}_2] = [24, 100]$$

 ψ_{a_0}

$$[\mathbf{a}_1] = [1, 65]$$

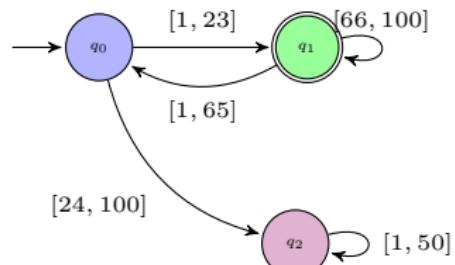
$$[\mathbf{a}_3] = [66, 100]$$

 ψ_{a_2}

$$[\mathbf{a}_4] = [1, 50]$$

$$[\mathbf{a}_5] = [51, 100]$$

hypothesis automaton



Ask Equivalence Query:
 counterexample $+ 24 \cdot 51$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |
| $a_2 a_5$ | + | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[\alpha_0] = [1, 23]$$

$$[\alpha_2] = [24, 100]$$

 ψ_{a_0}

$$[\alpha_1] = [1, 65]$$

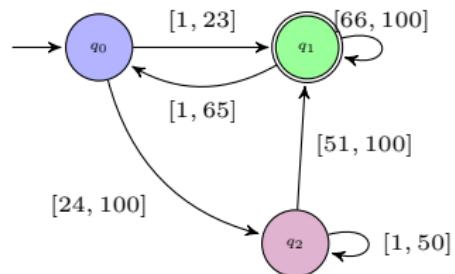
$$[\alpha_3] = [66, 100]$$

 ψ_{a_2}

$$[\alpha_4] = [1, 50]$$

$$[\alpha_5] = [51, 100]$$

hypothesis automaton



Ask Equivalence Query:
 counterexample $+ 24 \cdot 51$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |
| $a_2 a_5$ | + | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[\alpha_0] = [1, 23]$$

$$[\alpha_2] = [24, 100]$$

 ψ_{a_0}

$$[\alpha_1] = [1, 65]$$

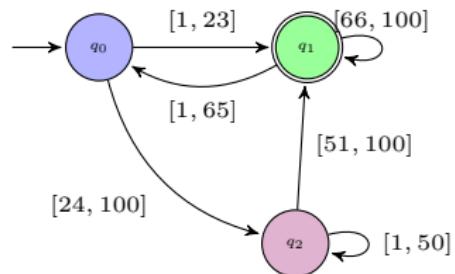
$$[\alpha_3] = [66, 100]$$

 ψ_{a_2}

$$[\alpha_4] = [1, 50]$$

$$[\alpha_5] = [51, 100]$$

hypothesis automaton



Ask Equivalence Query:
 counterexample $+ 24 \cdot 51$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |
| $a_2 a_5$ | + | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[\mathbf{a}_0] = [1, 23]$$

$$[\mathbf{a}_2] = [24, 100]$$

 ψ_{a_0}

$$[\mathbf{a}_1] = [1, 65]$$

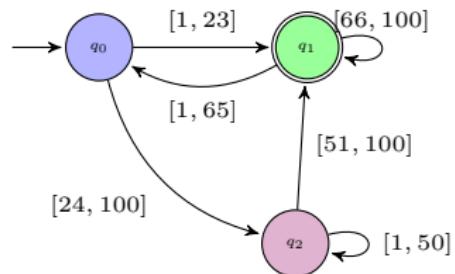
$$[\mathbf{a}_3] = [66, 100]$$

 ψ_{a_2}

$$[\mathbf{a}_4] = [1, 50]$$

$$[\mathbf{a}_5] = [51, 100]$$

hypothesis automaton



Ask Equivalence Query:
True

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |
| $a_2 a_5$ | + | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[a_0] = [1, 23]$$

$$[a_2] = [24, 100]$$

 ψ_{a_0}

$$[a_1] = [1, 65]$$

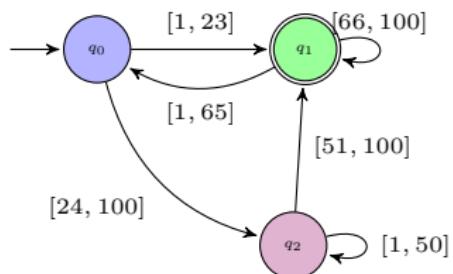
$$[a_3] = [66, 100]$$

 ψ_{a_2}

$$[a_4] = [1, 50]$$

$$[a_5] = [51, 100]$$

hypothesis automaton



Ask Equivalence Query:
True

$$M = \{\epsilon, 1, 24, 11, 166, 241, 2451, 111, 1661, 2411, 24511\}$$

$$|M| = 11, |MQ| = 7, |EQ| = 5, |S| = 3, |R| = 4$$

Example ($\Sigma = [1, 100]$)

observation table

| | ϵ | 1 |
|------------|------------|---|
| ϵ | — | + |
| a_0 | + | — |
| a_2 | — | — |
| $a_0 a_1$ | — | + |
| $a_0 a_3$ | + | — |
| $a_2 a_4$ | — | — |
| $a_2 a_5$ | + | — |

$$\psi = \{\psi_s\}_{s \in S}$$

 ψ_ϵ

$$[\mathbf{a}_0] = [1, 23]$$

$$[\mathbf{a}_2] = [24, 100]$$

 ψ_{a_0}

$$[\mathbf{a}_1] = [1, 65]$$

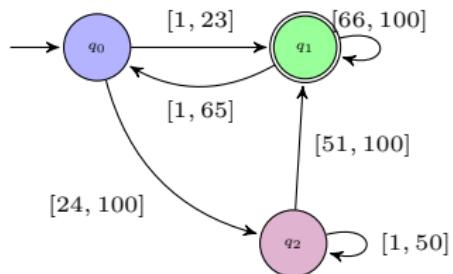
$$[\mathbf{a}_3] = [66, 100]$$

 ψ_{a_2}

$$[\mathbf{a}_4] = [1, 50]$$

$$[\mathbf{a}_5] = [51, 100]$$

hypothesis automaton



Ask Equivalence Query:
True

$$M = \{\epsilon, 1, 24, 11, 166, 241, 2451, 111, 1661, 2411, 24511\}$$

$$|M| = 790, |MQ| = 789, |EQ| = 2, |S| = 4, |R| = 396$$

$$|M| = 11, |MQ| = 7, |EQ| = 5, |S| = 3, |R| = 4$$

Conclusion - Future work

Conclusion:

- learning languages over large or infinite alphabets
 - finite number of convex intervals
 - without using variables
 - polynomial time in the number of states and maximum number of intervals

Future Work:

- PAC-learning (Probably Approximately Correct Learning)
 - the counterexample is not smallest
 - learn without equivalence queries
- what if we do not have intervals?
- other types of alphabets
- ...

Conclusion - Future work

Conclusion:

- learning languages over large or infinite alphabets
 - finite number of convex intervals
 - without using variables
 - polynomial time in the number of states and maximum number of intervals

Future Work:

- PAC-learning (Probably Approximately Correct Learning)
 - the counterexample is not smallest
 - learn without equivalence queries
- what if we do not have intervals?
- other types of alphabets
- ...

Thank you !

Any questions?