Model inference, some approaches

Learning methods and applications

Léo Henry

Univ. Rennes, INRIA & CNRS
Rennes (France)

Summary

The Passive Model-Learning Problem

Definition: Passive learning

Input $(S_+, S_-) \in (\Sigma^*)^{n+m}$

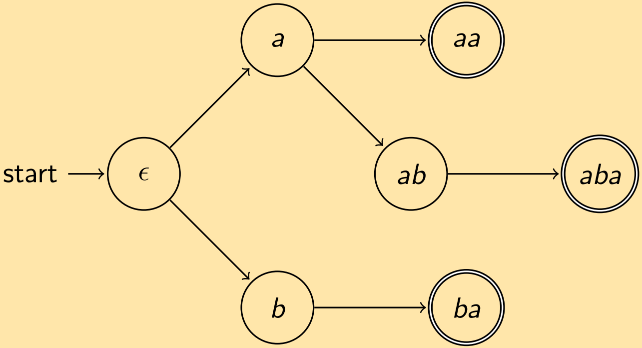
Output A model $\mathcal{A}$ s.t. $S_+ \subseteq L(\mathcal{A})$ and $L(\mathcal{A}) \cap S_- = \emptyset$

The Passive Model-Learning Problem

Definition: Passive learning

Input $(S_+, S_-) \in (\Sigma^*)^{n+m}$
Output A model $\mathcal{A}$ s.t. $S_+ \subseteq L(\mathcal{A})$ and $L(\mathcal{A}) \cap S_- = \emptyset$

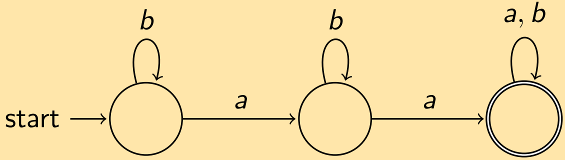
Consider : $\mathcal{A} \in \mathcal{DFA}$
Example : $S_+ = \{aa, ba, aba\}$, $S_- = \{a, ab\}$.



Occam's razor

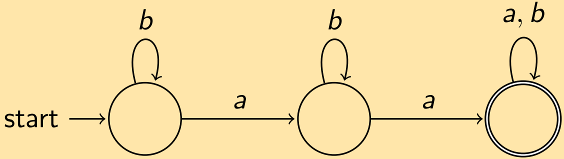
A more general model is required $\rightsquigarrow$ find a simpler model.

A more general model is required $\rightsquigarrow$ find a simpler model.
Example : $S_+ = \{aa, ba, aba\}$, $S_- = \{a, ab\}$.



Occam's razor

A more general model is required $\rightsquigarrow$ find a simpler model.
Example : $S_+ = \{aa, ba, aba\}$, $S_- = \{a, ab\}$.



Property: Gold 1978

"Given a sample $S = (S_+, S_-)$ and $k \in \mathbb{N}$, does a DFA with at most k states and consistent with S exist ?" is NP-complete.

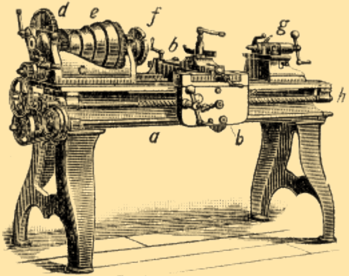
Input: (S_+, S_-)
Construct $PTA(S_+)$
Order the states of $PTA(S_+) = \{q_0, .., q_n\}$ according to the canonical order on words.
for $i \in [1, n]$ **do**
    if q_i *has not been merged with a smaller state* **then**
        Try to merge q_i with $q_0, ..., q_{i-1}$ until a merged DFA does not accept a word in S_-
    end
end
return the final DFA

Algorithm 1: Regular Positive Negative Inference

Property: Oncina and Garcia 1992

Given a sample S, RPNI provides a consistent DNA in polynomial time.
Furthermore if "enough information" on the original language $\mathcal{L}$ is provided in S, then RPNI returns the minimal accepting DFA for $\mathcal{L}$.

Cyber-Physical Production Systems



Lathe, p. 1218.

- **Complex** and **critical** systems.

  $\Downarrow$

  Analysis and verification desirable
- S_{-} is not accessible

$\rightsquigarrow$ CPPSs are learnt "from text".

Probabilistic approach

Property: Gold 1967

Regular languages are not identifiable in the limit from positive examples only.

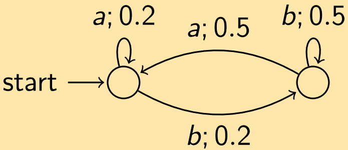
Probabilistic approach

Property: Gold 1967

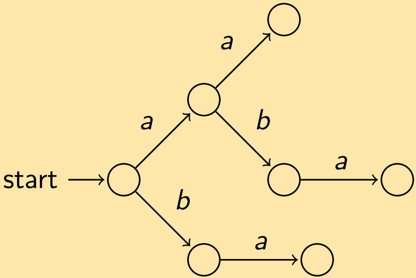
Regular languages are not identifiable in the limit from positive examples only.

Property: Angluin 1988

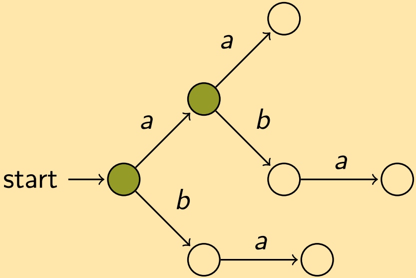
Stochastic regular languages are identifiable from text only with probability 1 (when the probabilities are rational)



ALERGIA



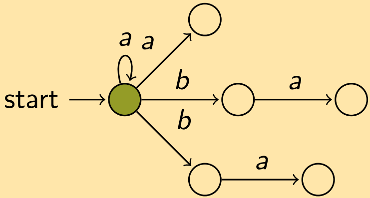
ALERGIA



$$|\frac{C(q,a)}{C(q)} - \frac{C(q',a)}{C(q')}| < K \quad (1)$$

$$\delta(q,a) \text{ and } \delta(q',a) \text{ compatibles} \quad (2)$$

ALERGIA



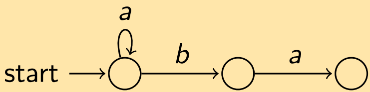
$$\left|\frac{C(q,a)}{C(q)} - \frac{C(q',a)}{C(q')}\right| < K \tag{1}$$

$$\delta(q,a) \text{ and } \delta(q',a) \text{ compatibles} \tag{2}$$

Alergia & MDI

Integrating probabilities

ALERGIA



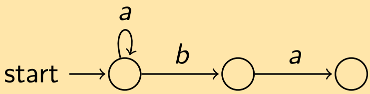
$$\left|\frac{C(q,a)}{C(q)} - \frac{C(q',a)}{C(q')}\right| < K \tag{1}$$

$$\delta(q,a) \text{ and } \delta(q',a) \text{ compatibles} \tag{2}$$

Alergia & MDI

Integrating probabilities

ALERGIA



MDI

$$\frac{\Delta(A, A')}{|A| - |A'|} \tag{3}$$

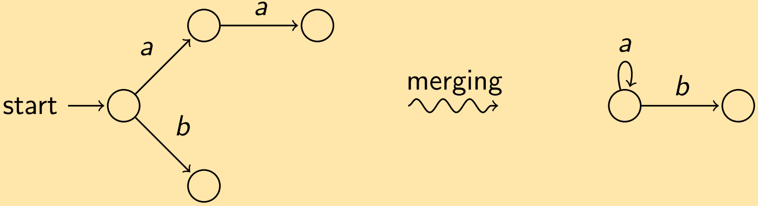
Entropy increase

Generalization

Adding time

The splitting operation

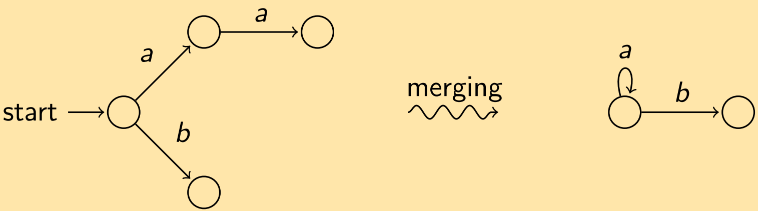
General approach :



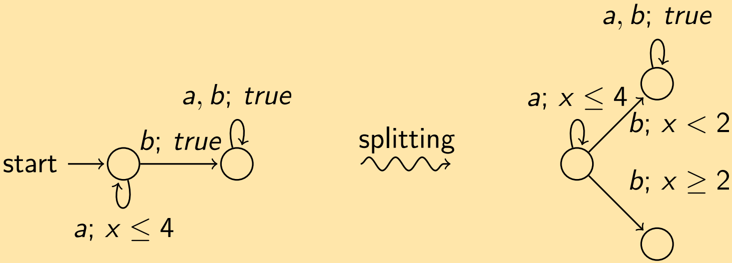
Adding time

The splitting operation

General approach :



With timing constraints :

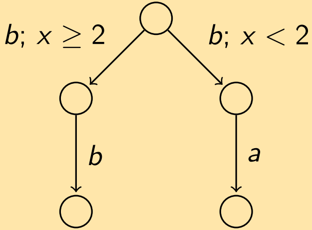


Algorithms for timed CPPSs

Restricted to **1 clock**, reset at each transition

Splitting criterion

Subtree difference



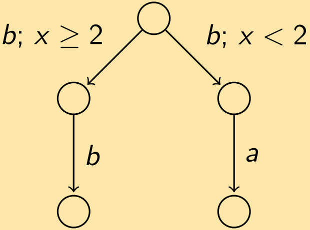
Algorithms for timed CPPSs

Restricted to **1 clock**, reset at each transition

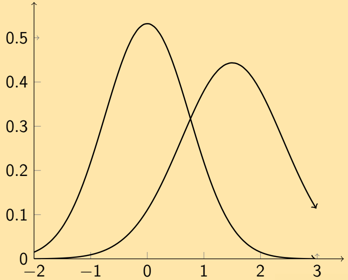
Splitting criterion

Subtree difference

Statistical inference



BUTLA



More CPPSs learning

- Offline $\rightsquigarrow$ Online
- According to a recent survey :
  - A general *learnable* model is required $\rightarrow$ data-driven modeling ;
  - such a model should be *hybrid* and *explicitly timed* ;
  - a component wise approach is recommended to provide symptoms of the problems.

Software analysis

Greybox analysis

Structural hypoteses

Software analysis

Greybox analysis

Structural hypoteses

Specification

Behavior model

Software analysis

Greybox analysis

Structural hypoteses

Specification

Behavior model

efficency increase

testing

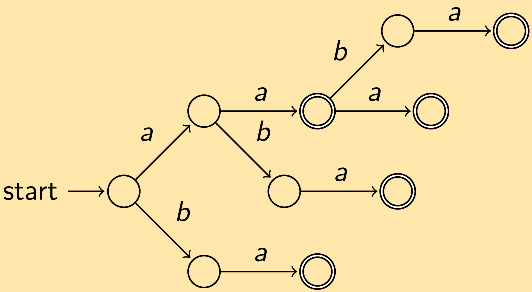
debugging

k-Tail algorithm...

... and its variations

Non stochastic models

k-Tail : $p \sim p' := \forall v \in \Sigma^{\leq k}, p.v \in \mathcal{P} \leftrightarrow p'.v \in \mathcal{P}$

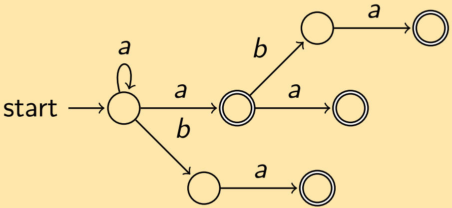


k-Tail algorithm...

... and its variations

Non stochastic models

k-Tail : $p \sim p' := \forall v \in \Sigma^{\leq k}, p.v \in \mathcal{P} \leftrightarrow p'.v \in \mathcal{P}$

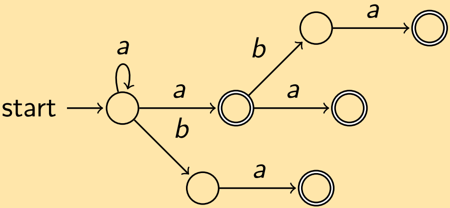


k-Tail algorithm...

... and its variations

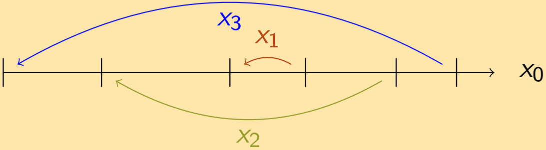
Non stochastic models

k-Tail : $p \sim p' := \forall v \in \Sigma^{\leq k}, p.v \in \mathcal{P} \leftrightarrow p'.v \in \mathcal{P}$



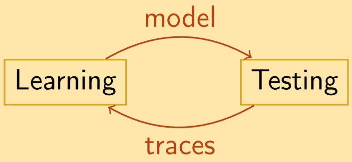
gk-Tail :handles parameters by using an ad-hoc function

Tk-Tail : adds time modeled by unbounded number of clocks.



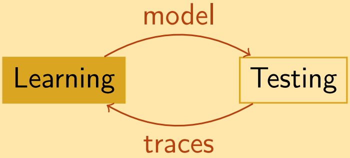
Integrating testing and learning

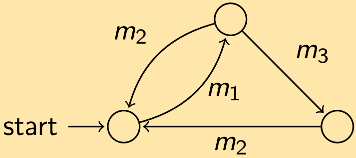
… using TAUTOKO



Integrating testing and learning

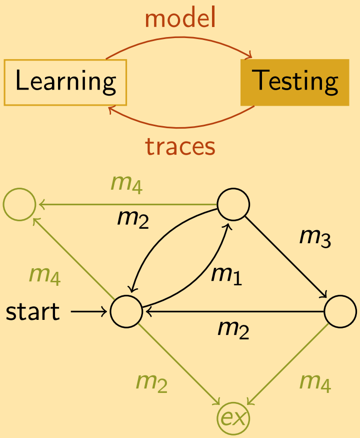
... using TAUTOKO





Integrating testing and learning

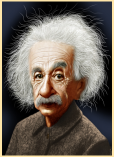
... using TAUTOKO



Active learning

the minimally adequate teacher

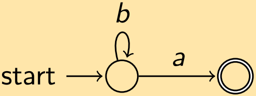
learner



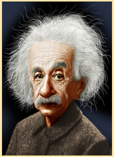
teacher

Active learning

the minimally adequate teacher



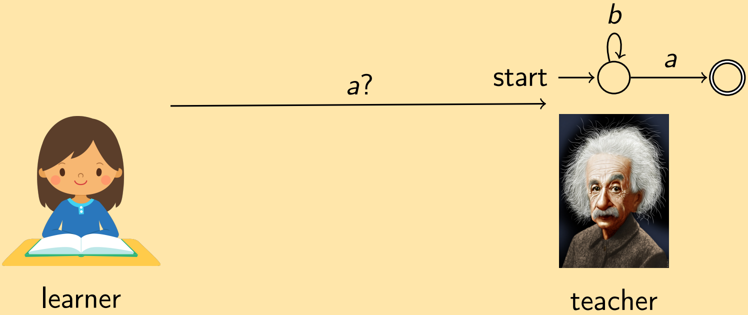
learner



teacher

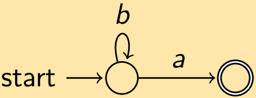
Active learning

the minimally adequate teacher



Active learning

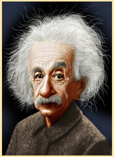
the minimally adequate teacher



a?

yes

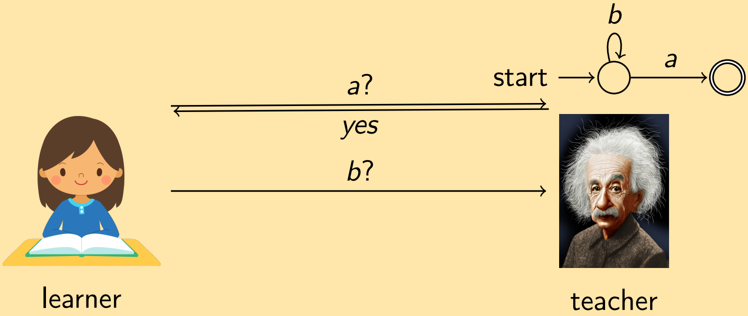
learner



teacher

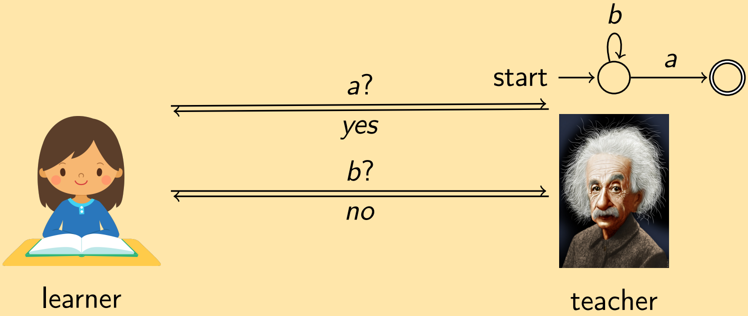
Active learning

the minimally adequate teacher



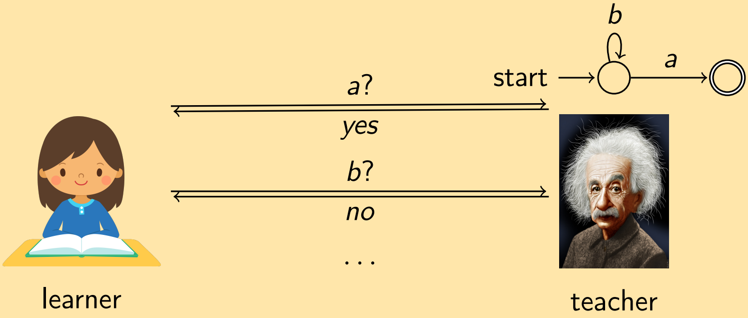
Active learning

the minimally adequate teacher



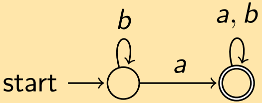
Active learning

the minimally adequate teacher

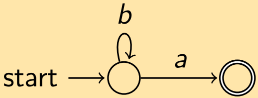


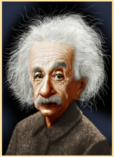
Active learning

the minimally adequate teacher



learner

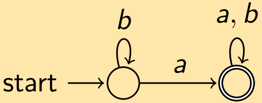


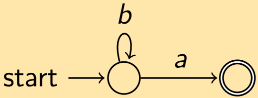


teacher

Active learning

the minimally adequate teacher

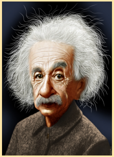




learner

my model ?

no : $ab \notin \mathcal{L}$



teacher

Active learning

the minimally adequate teacher

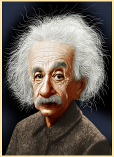
learner

Inclusion ω ?

result

Equivalence L ?

Y/Counterexample



teacher

Observation tables

	ϵ	a	ba
ϵ	1	0	0
a	0	1	0
b	0	1	0
aa	1	1	1
ab	0	0	1
ba	1	1	1
bb	1	0	0

		ϵ	a	ba
R	ϵ	1	0	0
	a	0	1	0
	b	0	1	0
$R.\Sigma$	aa	1	1	1
	ab	0	0	1
	ba	1	1	1
	bb	1	0	0

Observation tables

		S: ϵ	a	ba
R	ϵ	1	0	0
	a	0	1	0
	b	0	1	0
$R.\Sigma$	aa	1	1	1
	ab	0	0	1
	ba	1	1	1
	bb	1	0	0

Observation tables

		S		
		ϵ	a	ba
R	ϵ	1	0	0
	a	0	1	0
	b	0	1	0
$R.\Sigma$	aa	1	1	1
	ab	0	0	1
	ba	1	1	1
	bb	1	0	0

Observation tables

S

		ϵ	a	ba
R	ϵ	1	0	0
	a	0	1	0
	b	0	1	0
$R.\Sigma$	aa	1	1	1
	ab	0	0	1
	ba	1	1	1
	bb	1	0	0

Observation tables

		ϵ	a	ba
R	ϵ	1	0	0
R	a	0	1	0
R	b	0	1	0
$R.\Sigma$	aa	1	1	1
$R.\Sigma$	ab	0	0	1
$R.\Sigma$	ba	1	1	1
$R.\Sigma$	bb	1	0	0

closed $\forall u \in R, \forall a \in \Sigma, \exists v \in R u.a \sim_O v$

Observation tables

		S		
		ϵ	a	ba
R	ϵ	1	0	0
	a	0	1	0
	b	0	1	0
$R.\Sigma$	aa	1	1	1
	ab	0	0	1
	ba	1	1	1
	bb	1	0	0

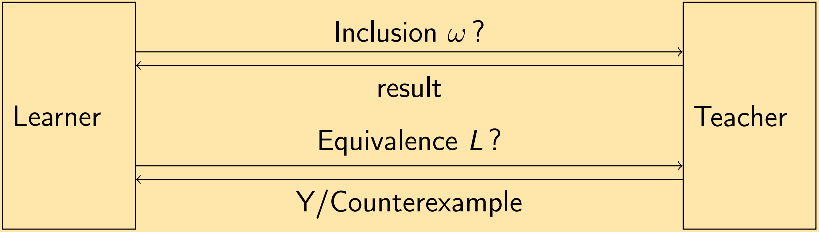
closed $\forall u \in R, \forall a \in \Sigma, \exists v \in R u.a \sim_O v$

consistent $\forall u, v \in R, (u \sim_O v \Rightarrow \forall a \in \Sigma u.a \sim_O v.a)$

Construct an empty observation table O ($R = S = \emptyset$)
repeat
  Make O closed and consistent (with membership queries)
  Perform an equivalence query
  if *The teacher provided a counterexample* u **then**
    Add u to O ($R \leftarrow R \cup u$)
    Complete O (with membership queries)
  end
until *the teacher replies "yes" to an equivalence query*;
Result: the model constructed from O

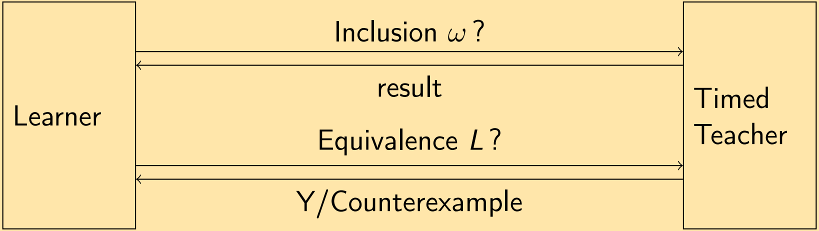
Learning and testing

on Mealy machines with timers



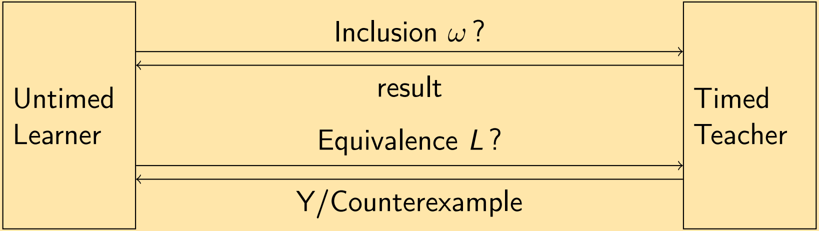
Learning and testing

on Mealy machines with timers



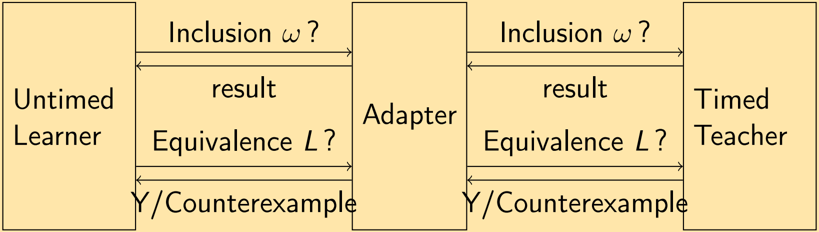
Learning and testing

on Mealy machines with timers



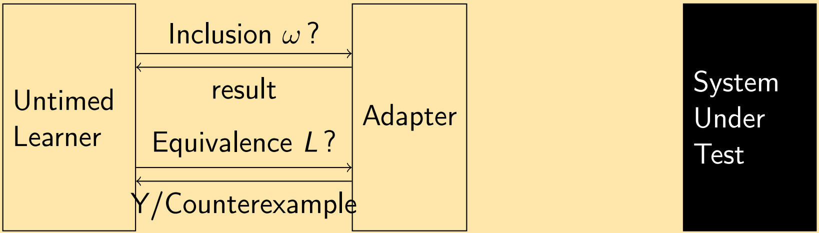
Learning and testing

on Mealy machines with timers



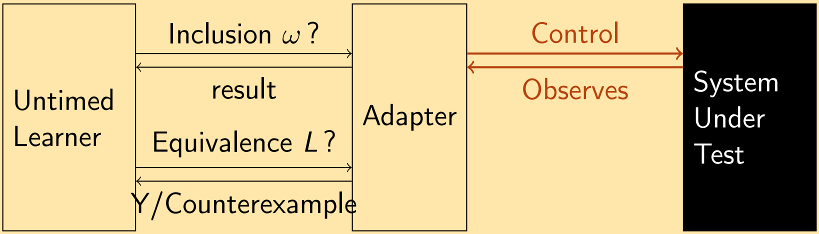
Learning and testing

on Mealy machines with timers



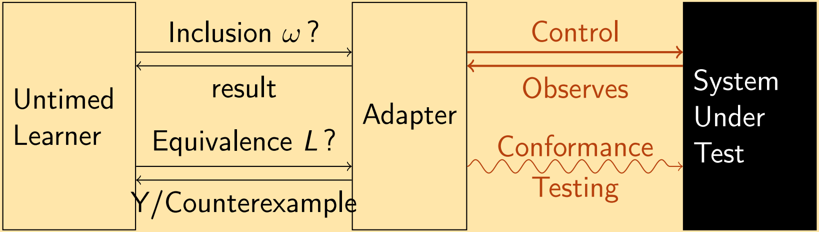
Learning and testing

on Mealy machines with timers



Learning and testing

on Mealy machines with timers

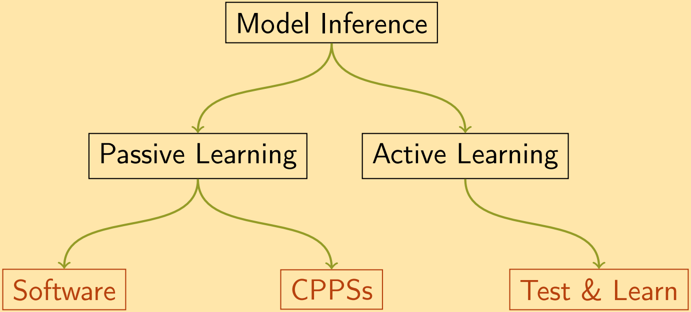


- Deterministic ERA can be learned by active learning ;

Event recording automata

- Deterministic ERA can be learned by active learning ;
- A very high complexity can be mitigated by targeting subclasses.

Take Away


Model Inference
Passive Learning
Active Learning
Software
CPPSs
Test & Learn