Verification of Machine Learning Models by Abstract Interpretation

Lecture for the
Software Verification course

Marco Zanella
Ph.D in Computer Science
University of Padova

Outline

Verification

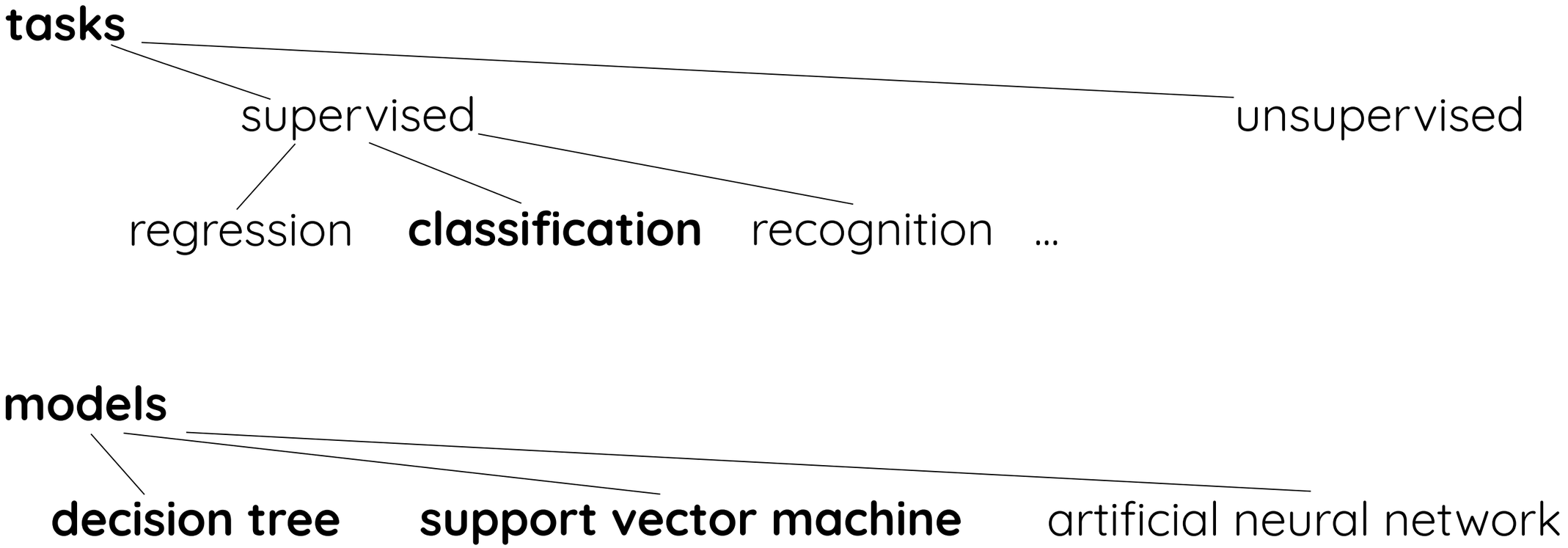
Training

Open Questions

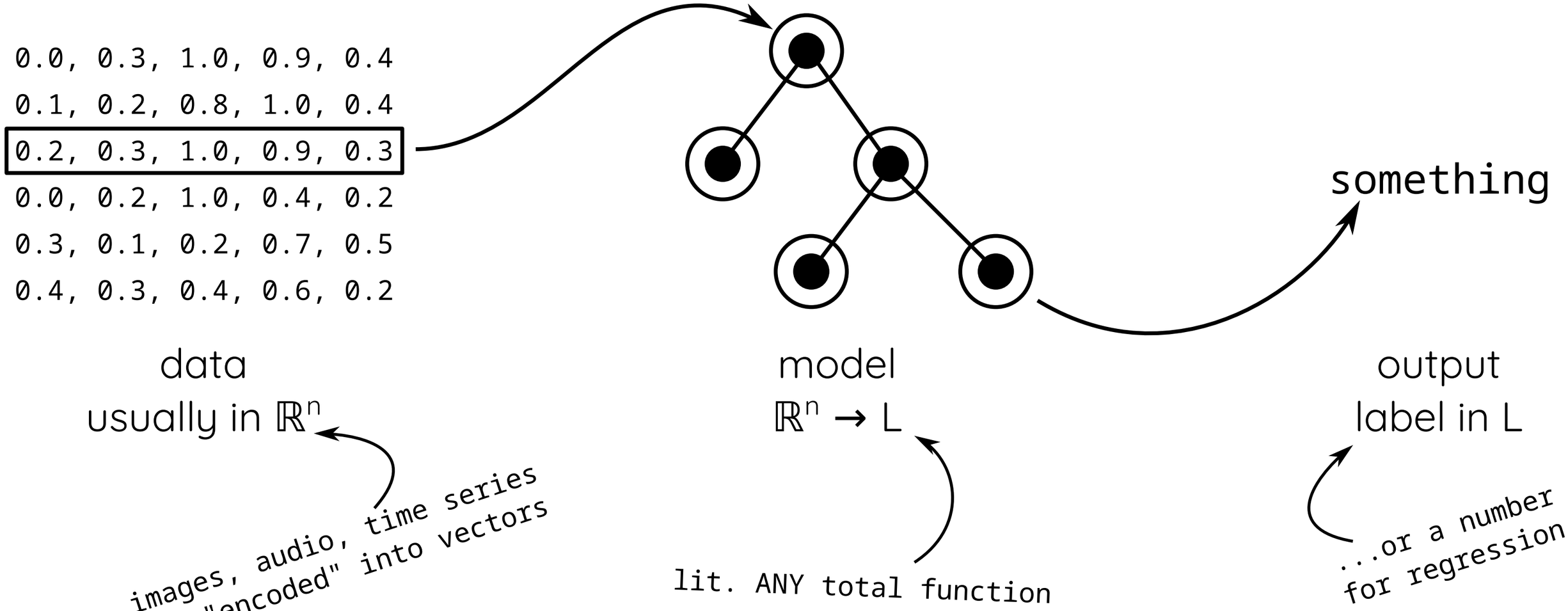
Ideas for thesis

Verification - Machine Learning

Machine Learning



Verification - Classification



Verification - Classification

Example

Iris Classification

data in $\mathbb{R}^4$

L = {setosa, virginica, versicolor}

```
5.1, 3.5, 1.4, 0.2
x1   x2   x3   x4
```

```
if x3 < 2.5
  return setosa
else
  if x4 < 1.5
    return versicolor
  else
    return virginica
```

setosa

x_1: sepal length (cm)

x_2: sepal width (cm)

x_3: petal length (cm)

x_4: petal width (cm)

decision tree model

output

Verification - Multiple Views

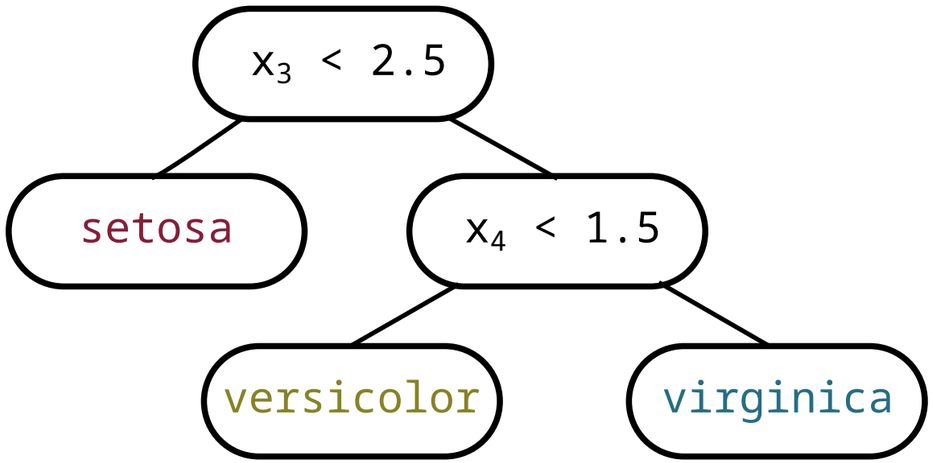
Example

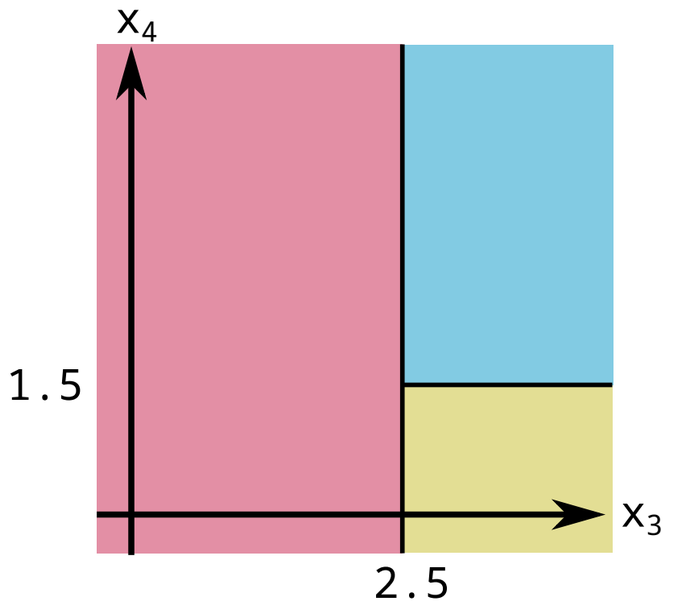
Iris Classification

data in $\mathbb{R}^4$

L = {setosa, virginica, versicolor}

```
if x3 < 2.5
  return setosa
else
  if x4 < 1.5
    return versicolor
  else
    return virginica
```





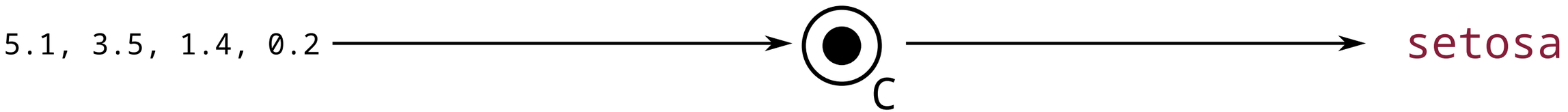
as a **program**

as a **model**

as a **partition**

Verification - Properties

Example
Iris Classification
data in $\mathbb{R}^4$
L = {setosa, virginica, versicolor}



model C **predicts** setosa for (**5.1, 3.5, 1.4, 0.2**)

C(5.1, 3.5, 1.4, 0.2) = setosa

is prediction **correct**? **Fair**? **Stable**? **Robust**?

properties of interest

Verification - Properties

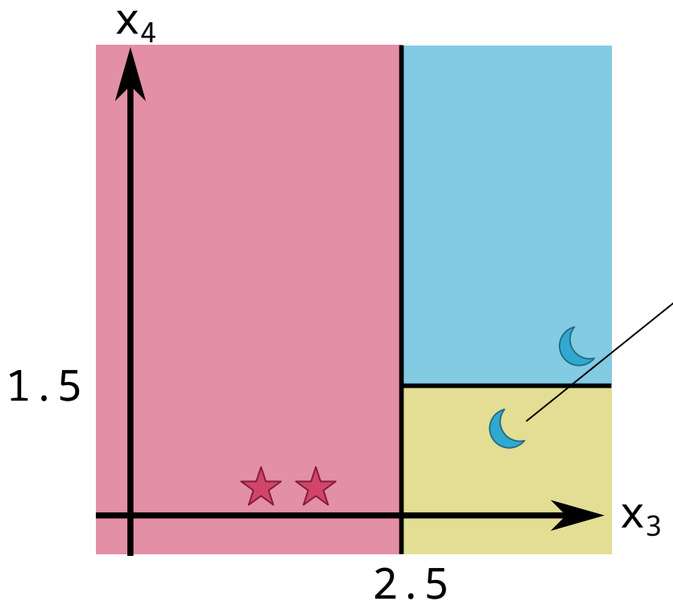
Example
Iris Classification
data in $\mathbb{R}^4$
L = {setosa, virginica, versicolor}

data	prediction	ground truth	correct?
`5.1, 3.5, 1.4, 0.2`	setosa	setosa	yes
`7.0, 3.2, 4.7, 1.4`	virginica	versicolor	no
`7.2, 3.2, 6.0, 1.8`	virginica	virginica	yes
`5.0, 3.0, 1.6, 0.2`	setosa	setosa	yes
		total	3
		relative	0.75

`a.k.a. accuracy`

Verification - Properties

Example
Iris Classification
data in $\mathbb{R}^4$
L = {setosa, virginica, versicolor}



classifier is **wrong** about one sample

according to a **source of thruth** D: $\mathbb{R}^4$ x L

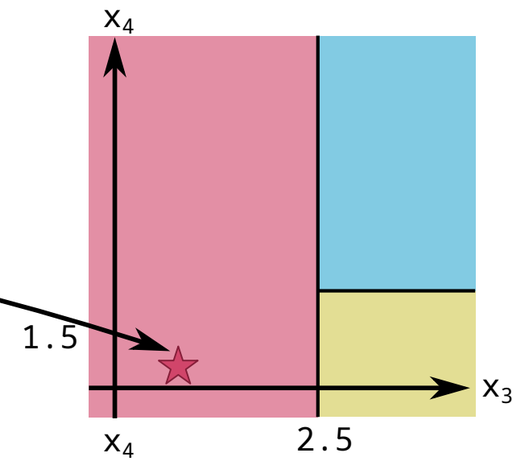
$$\text{accuracy}_D(C) = \frac{|\{(x, y) \text{ in } D \mid C(x) = y\}|}{|D|}$$

Verification - Properties

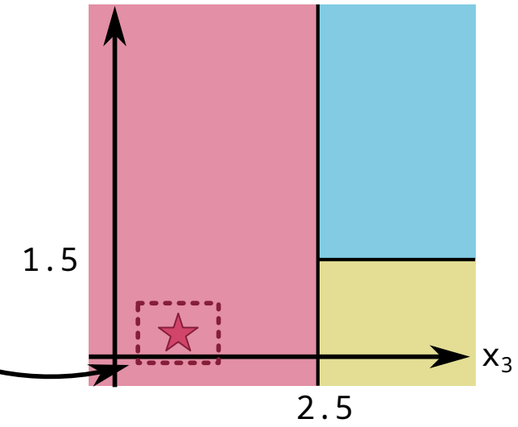
Example
Iris Classification
data in $\mathbb{R}^4$
L = {setosa, virginica, versicolor}

```
5.1, 3.5, 1.4, 0.2

7.0, 3.2, 4.7, 1.4

7.2, 3.2, 6.0, 1.8

5.0, 3.0, 1.6, 0.2
```

flower sizes were measured **manually**:
systematic errors



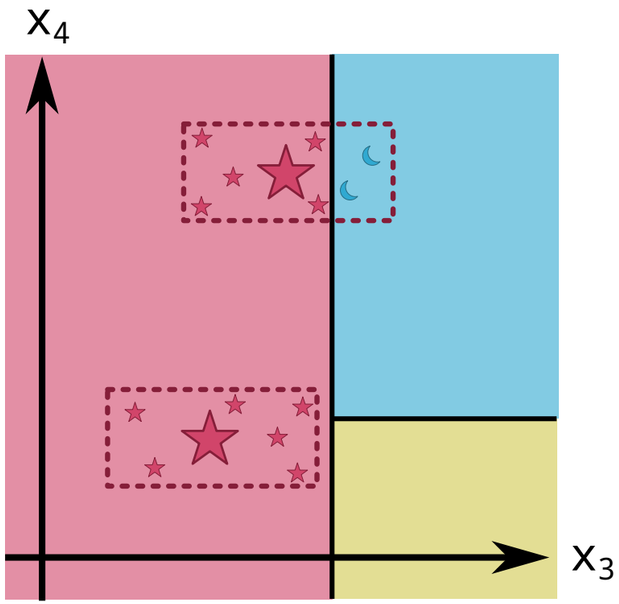

```
5.1±0.05, 3.5±0.06, 1.4±0.05, 0.2±0.03
```



(just an example, not part of the Iris dataset)

Verification - Properties

Example
Iris Classification
data in $\mathbb{R}^4$
L = {setosa, virginica, versicolor}



consistency may be **desirable**:

$$\text{stability}_{D,\delta,\varepsilon}(C) = \frac{|\{(x, \_) \text{ in } D \mid \forall x' \colon \delta(x, x') < \varepsilon \Rightarrow C(x') = C(x)\}|}{|D|}$$

Verification - Properties

Example
Iris Classification
data in $\mathbb{R}^4$
L = {setosa, virginica, versicolor}

properties can be **combined**:

correct	output as expected	$C(x) = y$
stable	resists small changes	$\forall x': \delta(x, x') < \varepsilon \Rightarrow C(x') = C(x)$
robust	correct **and** stable	$\forall x': \delta(x, x') < \varepsilon \Rightarrow C(x') = C(x) = y$
	...	

Verification - Abstract Interpretation

verification

standard techniques	abstract interpretation
classifier C: $\mathbb{R}^n \to L$	abstract classifier C^A: $A \to \wp(L)$
input in $\mathbb{R}^n$	input in A, abstracts $\wp(\mathbb{R}^n)$
output: label in L	output: set of labels in $\wp(L)$

example

C(3.5) = 

C(4.2) = 

$C^I([2.8, 3.6])$ = {  }

$C^I([3.8, 4.5])$ = {  }

Verification - Abstract Interpretation

soundness

$$\forall X \text{ in } \wp(\mathbb{R}^n),\ a \text{ in } A:\ X \subseteq \gamma(a) \Rightarrow \underbrace{\bigcup_{x \text{ in } X} \{C(x)\}}_{\text{collecting semantics}} \subseteq C^A(a)$$

example

$X = \{1.0, 1.5\}$ $a = [1.0, 1.6]$

$C(1.0) =$ 

$C(1.5) =$ 

$C^A_1(a) = \{$,  $\}$ **sound**

$C^A_2(a) = \{$  ,  ,  $\}$ **sound**

$C^A_3(a) = \{$  $\}$ **unsound**

Verification - Abstract Interpretation

soundness

$$\forall X \text{ in } \wp(\mathbb{R}^n),\ a \text{ in } A:\ X \subseteq \gamma(a) \Rightarrow \bigcup_{x \text{ in } X} \{C(x)\} \subseteq C^A(a)$$

consequences

$C^A(a) = \{k\} \Rightarrow \forall x \text{ in } X: C(x) = k$ **exactly one label** in a ⇒ **exactly the same** in X:

C **stable** over X

$|C^A(a)| > 1$ nothing can be said

`loss of precision?`

Verification - Abstract Interpretation

completeness

$$\forall X \text{ in } \wp(\mathbb{R}^n),\ a \text{ in } A\text{: } X \subseteq \gamma(a) \Rightarrow \bigcup_{x \text{ in } X} \{C(x)\} = C^A(a)$$

consequences

$\forall k \text{ in } C^A(a)\text{: } \exists x \text{ in } X\text{: } C(x) = k$

more labels in a ⇒ **different labels** in X:

C **not stable** over X

Verification - Abstract Classifier

stability verification

1. given C, D, **δ**, **ε**

2. **choose** abstraction **A**

3. **build** C^A

4. for each **x** in D:

4.1. **find a** abstracting set X of x′ s.t. **δ**(x, x′) < **ε** `(hint: use a)`

4.1. if $|C^A(a)| = 1$ then C(X) is **stable**

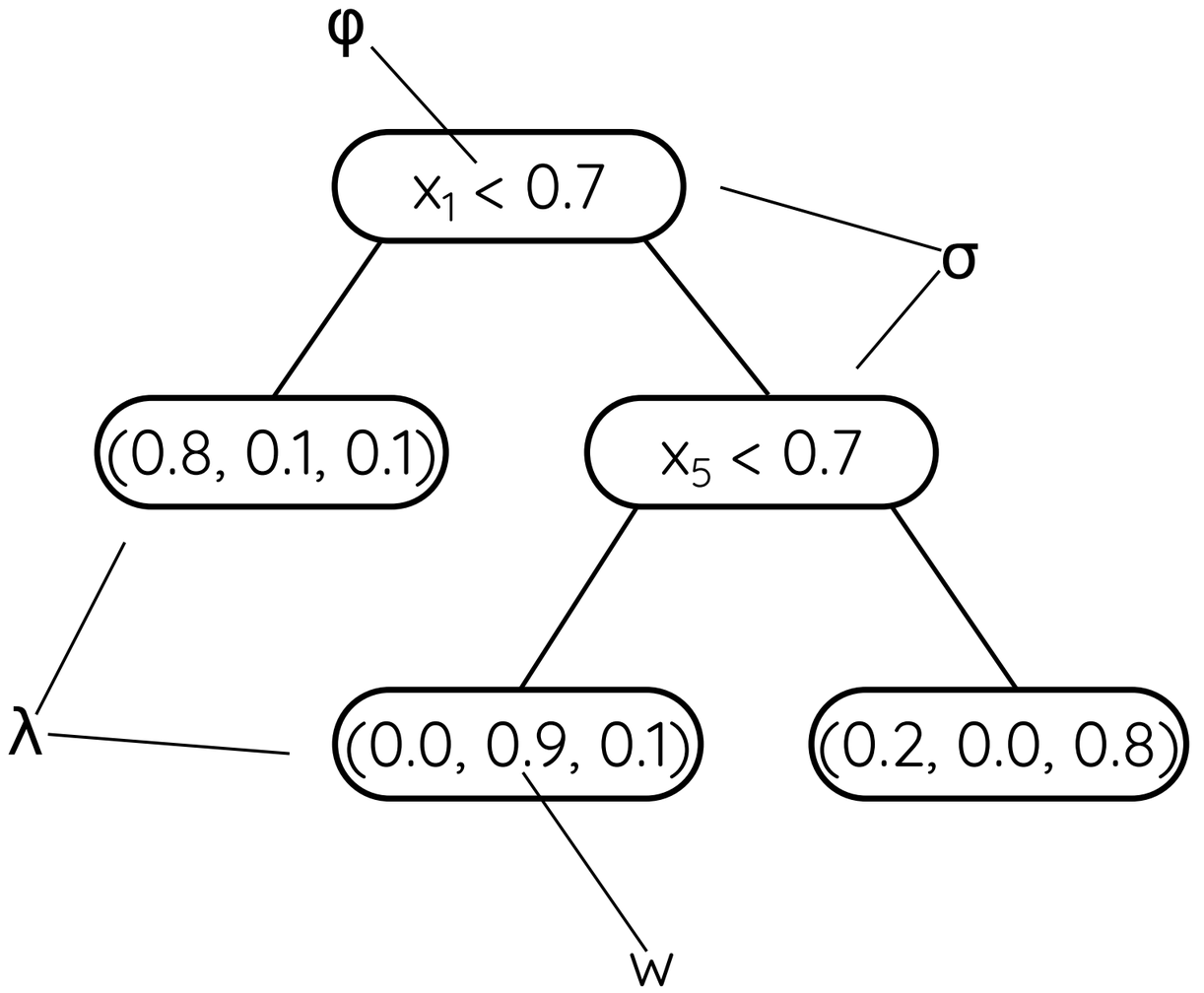
Verification - Decision Tree

decision tree

$$\nu ::= \lambda(\mathbb{R}^{|L|})$$
$$\mid \sigma(\mathbb{R}^n \to \{\text{true, false}\}, \nu, \nu)$$

classification

$$C_{\lambda(w)}(\_) = \arg\max_{k \text{ in } L}(w_k)$$
$$C_{\sigma(\varphi, L, R)}(x) = \text{if } \varphi(x) \text{ then } C_L(x) \text{ else } C_R(x)$$

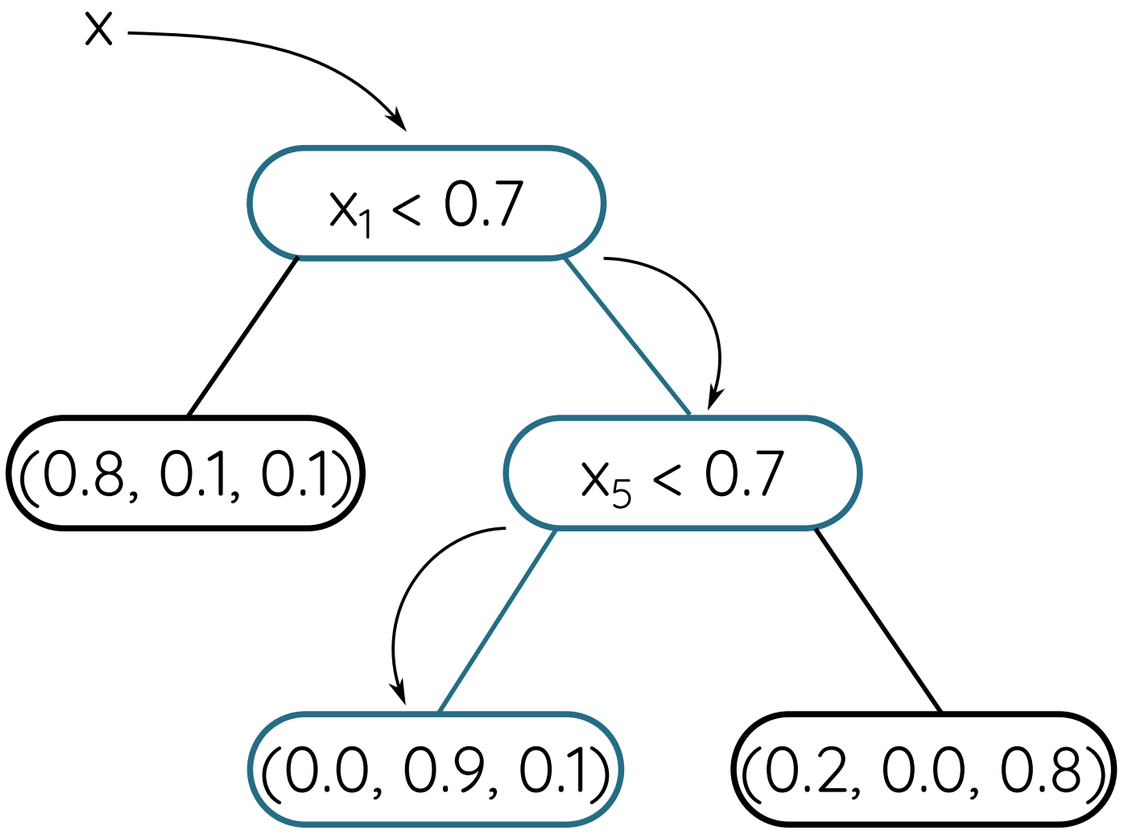


Verification - Decision Tree

classification as reachability

find **λ reachable** by x:
- **λ** is a leaf
- **φ**(x) for every **φ** in the root-leaf path

$C(x) = C_{\lambda}(\_)$



example

x = (0.9, 0.3, 5.1, 2.3, 0.2)

λ = (0.0, 0.9, 0.1)

C(x) =

Verification - Decision Tree

abstract classification as reachability

find set Λ of **every** leaf **reachable** by points in $\gamma(a)$

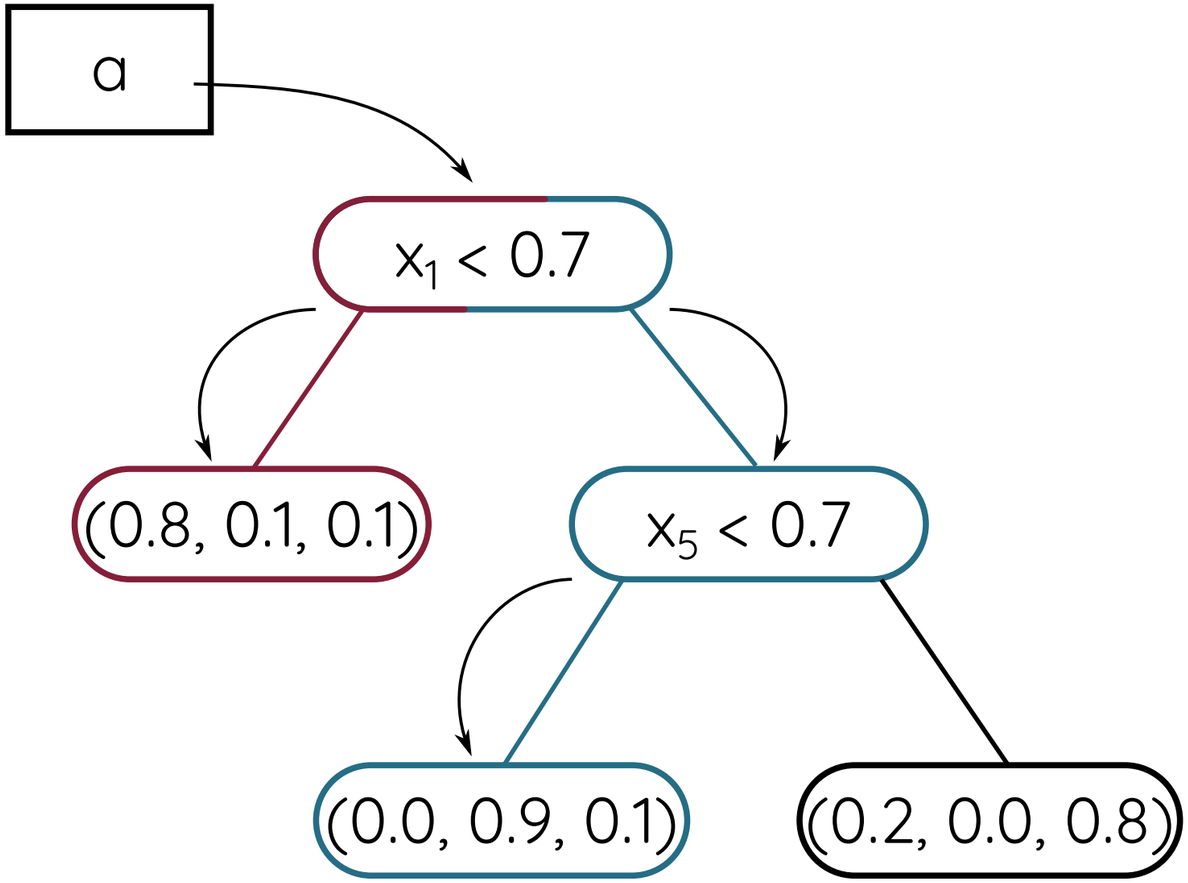
$C^A(a) = U_{\lambda \text{ in } \Lambda} C_\lambda(\_)$

example

$a = ([0.4, 0.8], ..., [0.2, 0.6])$

$\Lambda = \{(0.8, 0.1, 0.1), (0.0, 0.9, 0.1)\}$

$C^A(a) = C_{(0.8, 0.1, 0.1)}(\_) \cup C_{(0.0, 0.9, 0.1)}(\_) = \{\bigstar, \leftmoon\}$



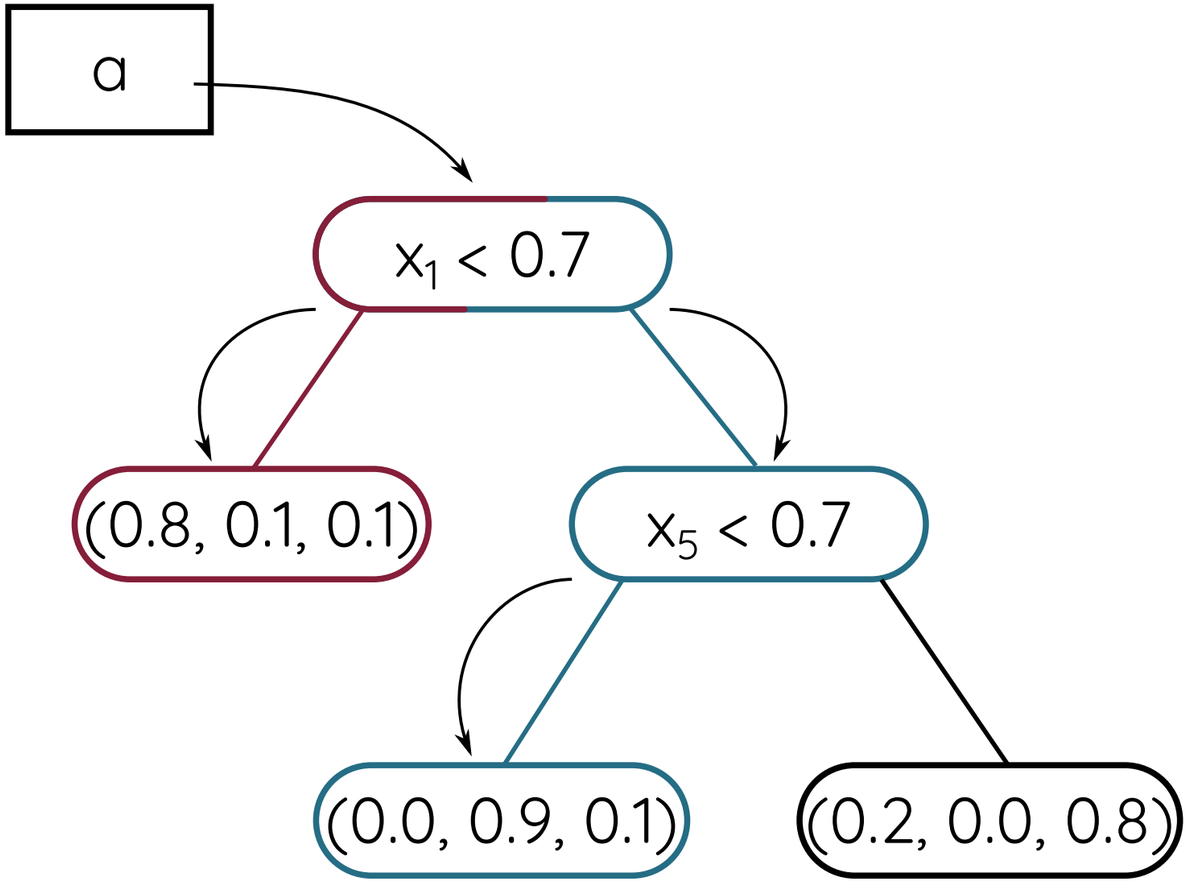
Verification - Decision Tree

soundness

no labels are lost

hint

if k is part of $U_{x \text{ in } \gamma(a)} C(x)$,
then some λ s.t. $C_\lambda(\_) = k$
is reachable by some x.
Hence λ in Λ, and k in $C^A(a)$

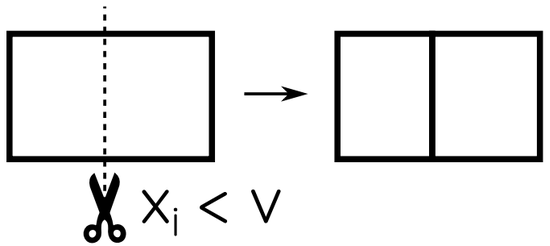


`[tldr] label in the concrete, label in the abstract`

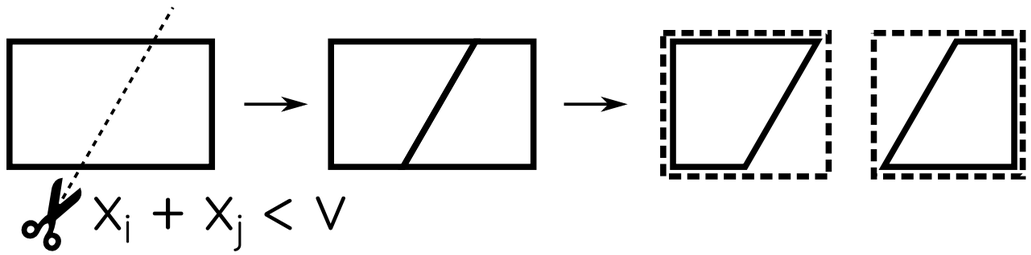
Verification - Decision Tree

completeness (for box domain)
no stray labels

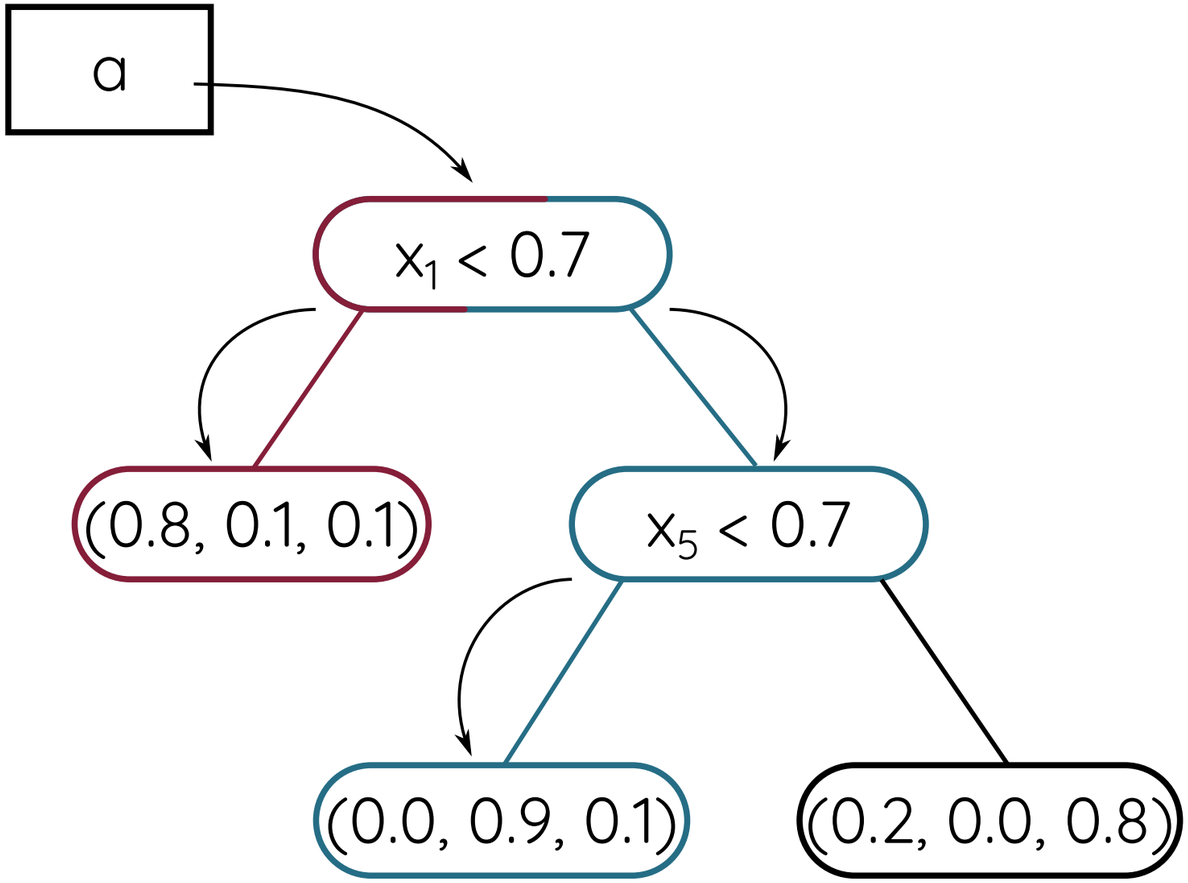
hint
a box split by φ: $x_i < v$ yields two boxes
no "adjustements" needed



no information loss



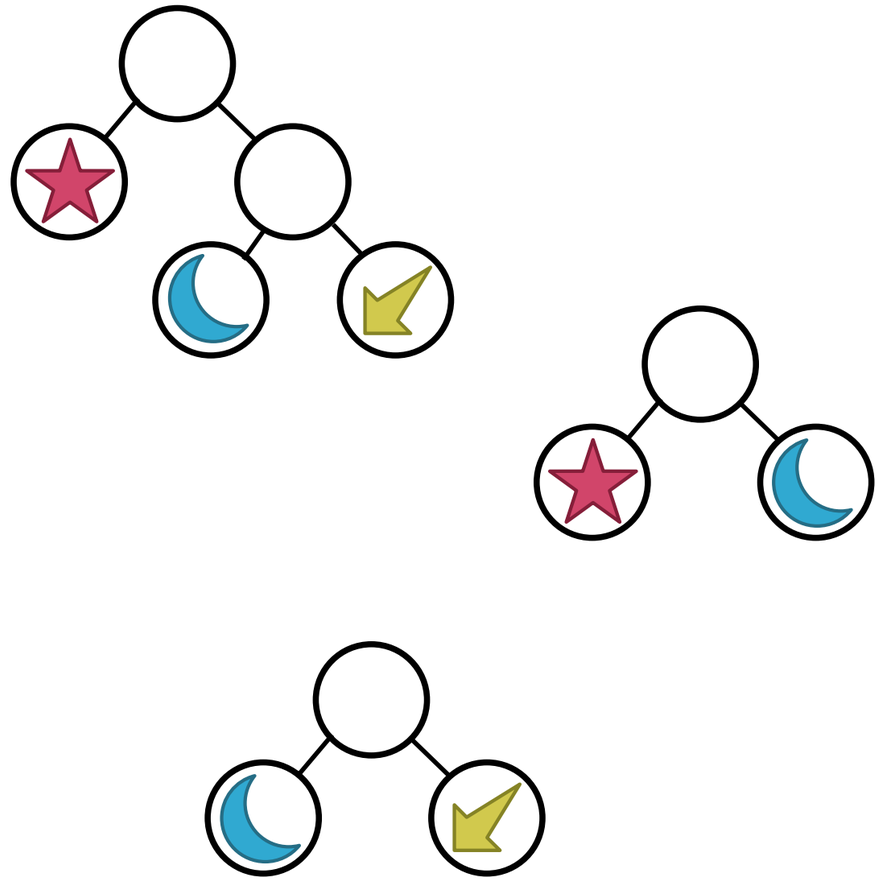
overapproximation



Verification - Forest

forest

set of decision trees



classification

$F(x) = F.map(t \rightarrow C_t(x)).reduce(w, \boldsymbol{\lambda} \rightarrow w + \boldsymbol{\lambda}_w)$

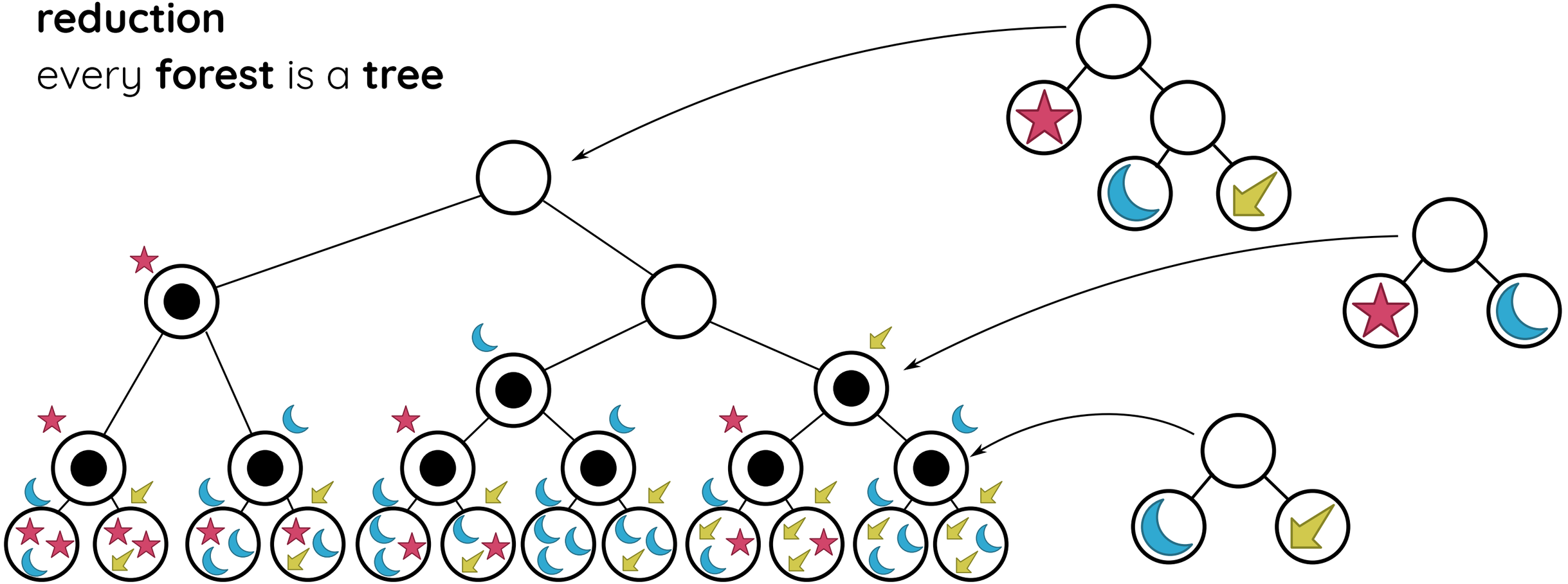
$F(x) = F.map(t \rightarrow C_t(x)).reduce(argmax_{k \text{ in } L})$

$F(x) = \ldots$

Verification - Forest

reduction

every **forest** is a **tree**

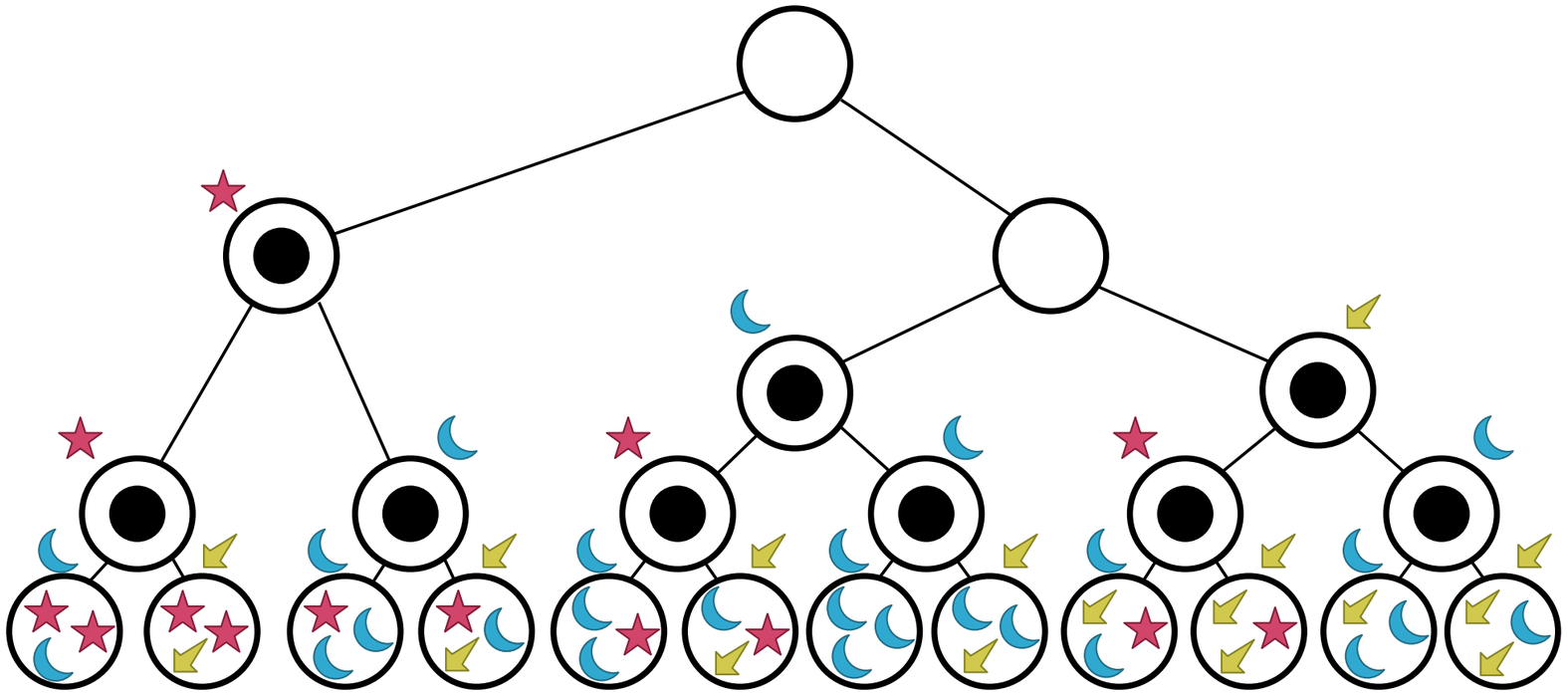


complete verification for **trees** ⇒ **complete** verification for **forests**

Verification - Forest

issue

combinatorial explosion of leaves



mitigation

best-first **heuristic** traversal
prioritize new labels

mixed sound-complete analysis

★ wins

stop on first **counter example**
{★, ☾, ...} ⇒ unstable

`...and more graph tricks`

Verification - Forest

silva

Silvarum **I**nterpretatione **L**ator **V**alens **A**nalysis

coded the hard way.
in C.

https://github.com/abstract-machine-learning/silva

Verification - Forest

silva

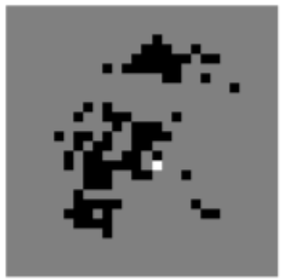
Silvarum **I**nterpretatione **L**ator **V**alens **A**nalysis

original image

prediction: 7

altered image

prediction: 2



difference (emphasys)

https://github.com/abstract-machine-learning/silva

Verification - Forest

silva

support for different models

RF		Gini, max			Gini, average		
B	d	acc.%	stab.%	time(s)	acc.%	stab.%	time(s)
5	5	66.7	14.3	0.6	76.4	18.1	0.5
5	25	90.5	21.4	0.9	90.6	21.4	0.9
5	50	90.5	19.5	1.0	90.5	19.5	1.0
25	5	82.8	12.1	3.0	85.7	16.8	3.6
25	25	96.0	31.7	22.5	96.1	31.8	24.4
25	50	96.0	26.5	30.2	96.0	26.5	30.0
50	5	84.0	12.7	40.8	85.8	18.3	662.9
50	25	96.6	35.1 $\pm$1.7	965.4	96.6	35.2 $\pm$1.7	970.8
50	50	96.5	35.3 $\pm$2.1	1126.2	96.5	35.3 $\pm$2.1	1136.8

RF		entropy, max			entropy, average		
B	d	acc.%	stab.%	time(s)	acc.%	stab.%	time(s)
5	5	67.5	9.6	0.5	76.1	20.2	0.5
5	25	91.3	28.4	0.8	91.3	28.4	0.8
5	50	91.3	22.8	0.9	91.3	22.8	0.9
25	5	81.3	16.5	3.7	85.6	19.7	6.9
25	25	96.2	39.4	28.7	96.2	39.4	28.8
25	50	96.2	36.4	36.7	96.2	36.4	34.9
50	5	83.4	20.8	67.4	85.4	24.2	811.9
50	25	96.5	43.1 $\pm$1.5	863.9	96.5	43.1 $\pm$1.5	874.1
50	50	96.6	41.3 $\pm$1.4	824.5	96.6	41.3 $\pm$1.4	826.1

https://github.com/abstract-machine-learning/silva

Verification - Forest

silva

support for different indicators

RF							
B	d	acc. %	stab. %	rob. %	frag. %	vuln. %	break. %
5	5	76.1	20.2	18.0	58.1	2.2	21.7
5	25	91.3	28.4	27.9	63.5	0.5	8.1
5	50	91.3	22.8	22.4	68.9	0.4	8.3
25	5	85.6	19.7	19.2	66.4	0.5	13.9
25	25	96.2	39.4	39.3	56.8	0.1	3.7
25	50	96.2	36.4	36.2	60.0	0.1	3.6
50	5	85.4	24.2	23.6	61.7	0.6	14.0
50	25	96.5	43.1 $\pm$1.5	43.0 $\pm$1.5	53.5 $\pm$1.5	0.1	3.4
50	50	96.6	41.3 $\pm$1.4	41.2 $\pm$1.4	55.4 $\pm$1.4	0.1	3.3

https://github.com/abstract-machine-learning/silva

Verification - Support Vector Machine

Binary Support Vector Machine

$$C(x) = \text{sign}(\Sigma_i\, k(x, sv_i) + b)$$

label: ±1

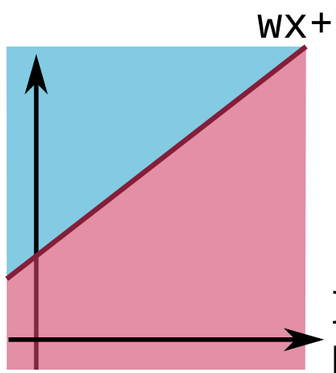
kernel

"similarity" between its arguments

input

support vector

chosen during training



linear kernel

$k(x_1, x_2) = x_1 \cdot x_2$

Verification - Support Vector Machine

Binary Support Vector Machine

$$C(x) = \text{sign}(\Sigma_i\, k(x, sv_i) + b)$$

$$C^A(a) = \text{sign}^A(\Sigma^A_i\, k^A(a, sv_i) +^A b)$$

depends on kernel

linear $x \cdot sv_i$

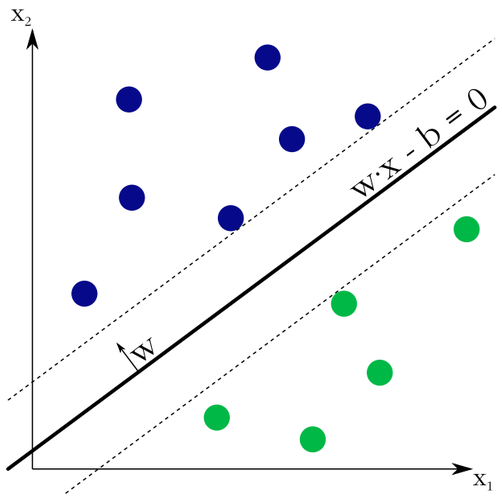
polynomial $(x \cdot sv_i + c)^d$

RBF $e^{-\|x - sv_i\|^2/z}$

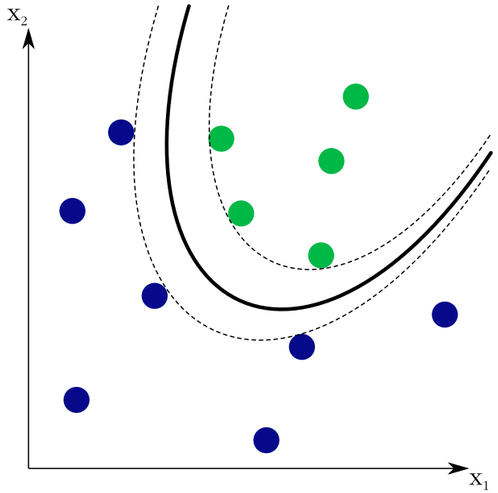
easy

Verification - Support Vector Machine

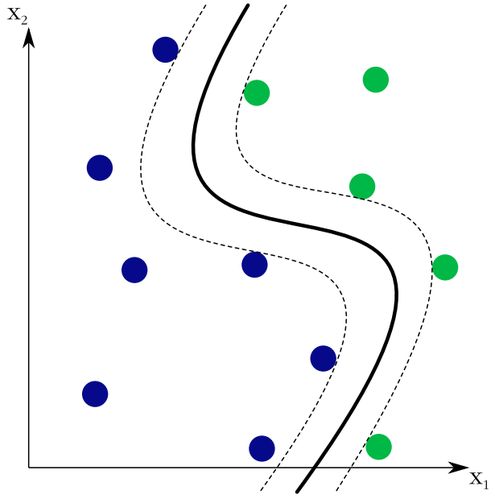
examples of kernels



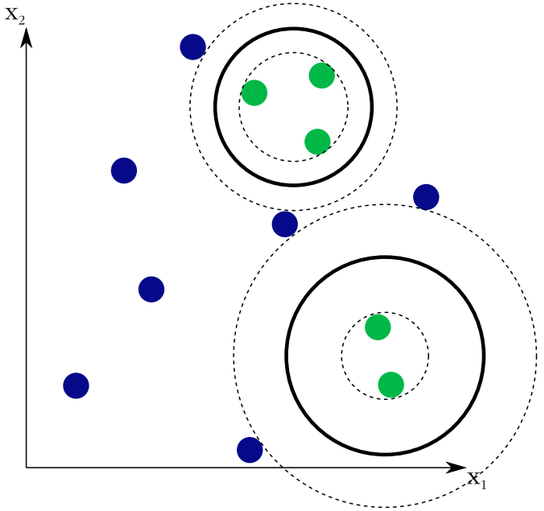
linear



polynomial
degree 2



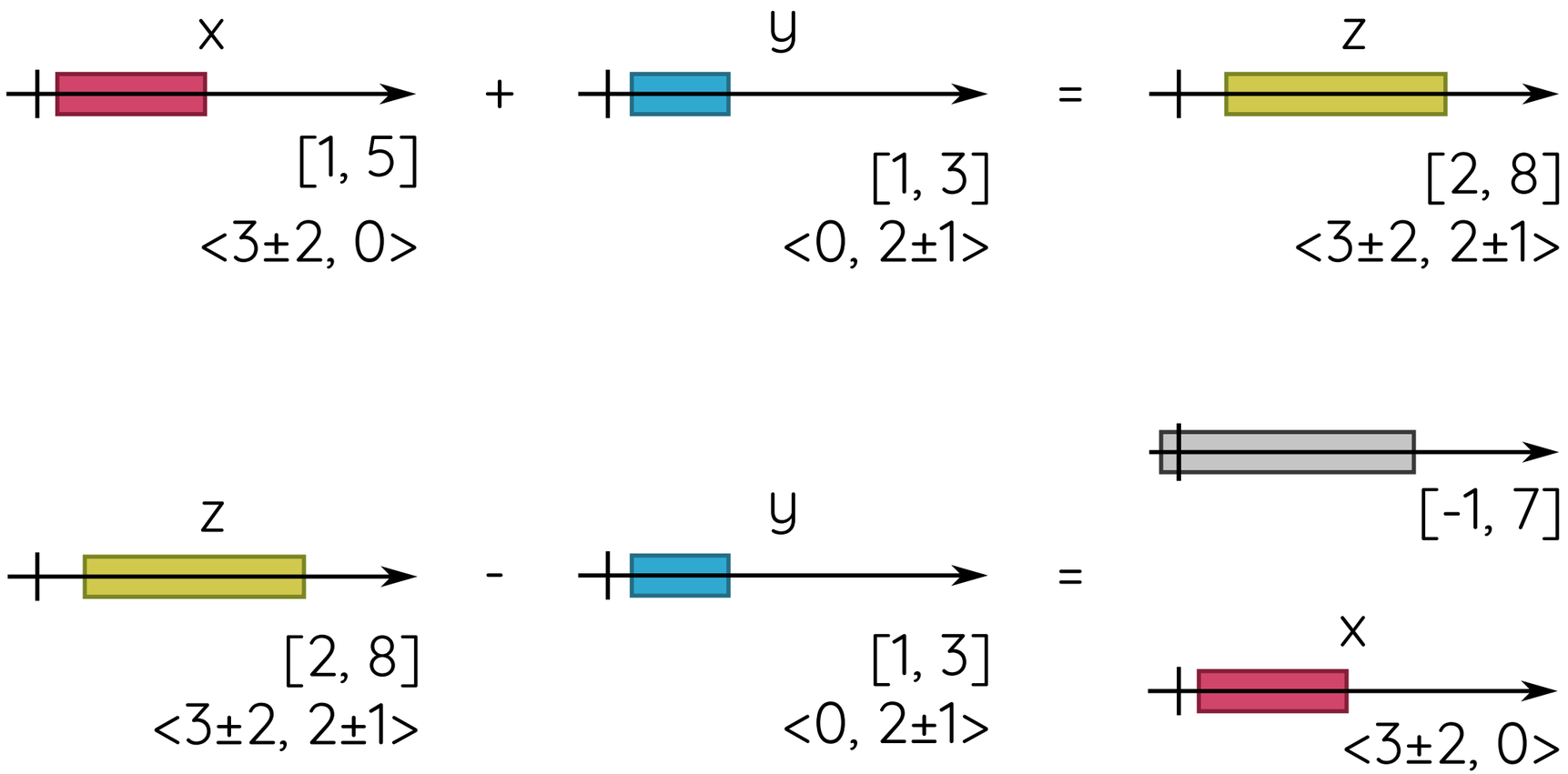
polynomial
degree 3



RBF

Verification - Support Vector Machine

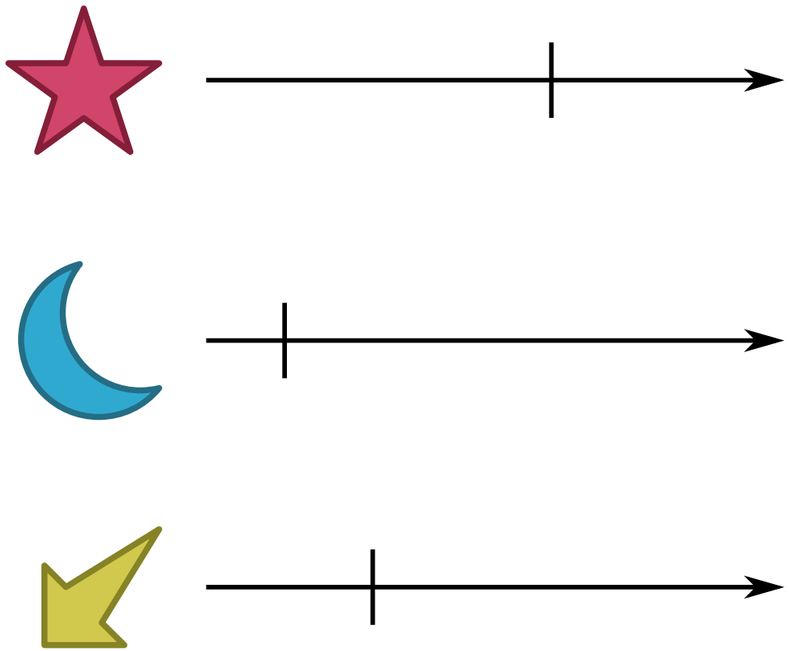
abstraction



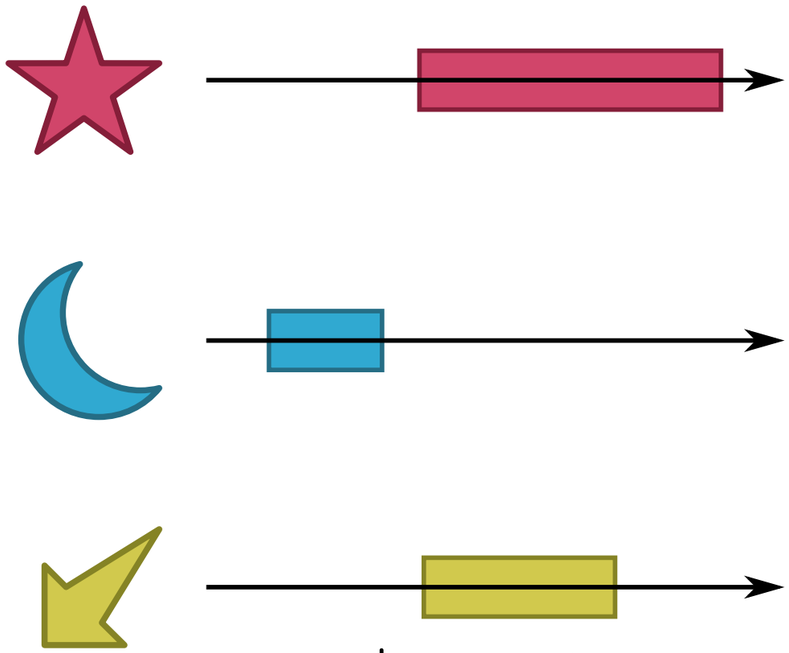
Verification - Support Vector Machine

non binary

use $O(n^2)$ binary SVMs



concrete



abstract

tournament scheme

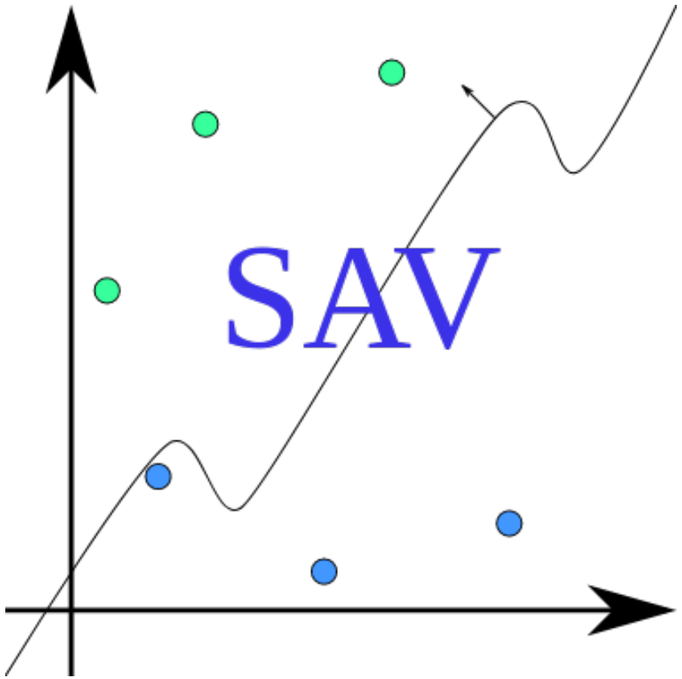
winner 

winner {  ,  }

Verification - Support Vector Machine

SAVer

SVM **A**bstract **V**erifier



https://github.com/abstract-machine-learning/saver

Verification - Support Vector Machine

SAVer

support for different kernels

P_δ^∞	Provable Robustness %				
	Linear	**Poly2**	**Poly3**	**Poly9**	**RBF**
0.01	82.23	98.64	99.07	98.51	99.83
0.02	38.95	94.82	96.96	96.34	99.57
0.03	12.77	82.14	91.80	92.85	99.19
0.04	3.22	57.44	78.95	87.33	97.27
0.05	0.71	30.52	57.31	77.69	93.58
0.06	0.13	14.89	34.80	61.12	82.21
0.07	0.00	7.89	18.36	39.75	67.76
0.08	0.00	4.08	10.64	23.70	48.02
0.09	0	1.61	6.28	12.86	28.10
0.10	0	0.58	3.33	7.18	16.38

analysis is **sound**, not complete
results are **lowerbounds**

box perturbation

handwritten digit recognition (MNIST)

https://github.com/abstract-machine-learning/saver

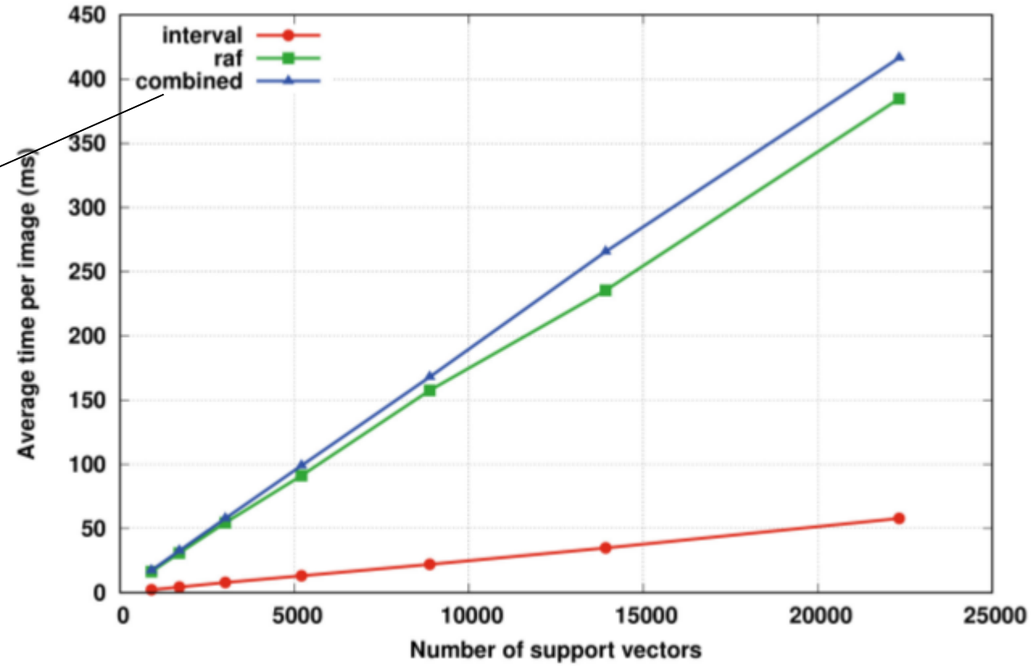
Verification - Support Vector Machine

SAVer

support for different abstractions

P_δ^∞	Provable Robustness %		
	Interval	RAF	Combined
0.01	47.73	99.83	99.83
0.02	14.95	99.57	99.57
0.03	6.26	99.19	99.19
0.04	2.42	97.27	97.27
0.05	0.82	93.58	93.58
0.06	0.17	82.21	82.21
0.07	0.04	67.76	67.76
0.08	0	48.02	48.02
0.09	0	28.10	28.10
0.10	0	16.38	16.38

cartesian product
interval x RAF



https://github.com/abstract-machine-learning/saver

Verification - Support Vector Machine

SAVer

support for different indicators

P_δ^∞	Provable Robustness %		Provable Vulnerability %	
	MNIST	F-MNIST	MNIST	F-MNIST
0.01	99.83	88.59	94.48	39.20
0.02	99.57	60.63	73.62	11.80
0.03	99.19	42.13	48.47	5.50
0.04	97.27	27.24	32.51	3.00
0.05	93.58	18.36	20.25	1.50
0.06	82.21	12.18	9.86	0.90
0.07	67.76	8.22	3.68	0.60
0.08	48.02	5.23	0.61	0.40
0.09	28.10	1.96	0	0.10
0.10	16.38	0.48	0	0

less stability may imply
less vulnerability

clothes recognition (Fashion MNIST)

https://github.com/abstract-machine-learning/saver

Verification - Fairness

a "new" property

are **similar individuals** treated in the **same way**?

Alex	Alice
bachelor	bachelor
25 y.o.	24 y.o.
male	female
2 y. work. exp	2 y. work. exp
hire	**don't hire**

Verification - Fairness

a "new" property

are **similar individuals** treated in the **same way**?

$\delta(x_1, x_2) < \varepsilon$ $\qquad$ $C(x_1) = C(x_2)$

same as **stability**

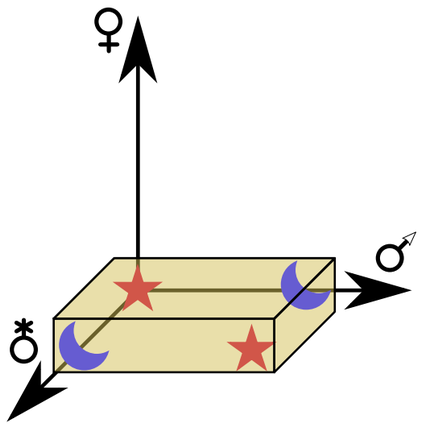
issue

categorical features gender in {male, female, non-binary}

one-hot encoding is_male, is_female, is_non_binary in {0, 1}

Verification - Fairness

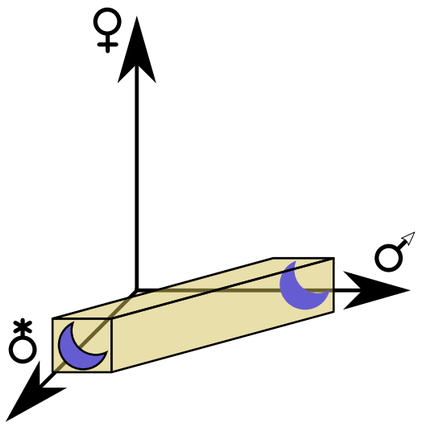
a "new" property



$\gamma([0, 1], [0, 0], [0, 1])$

male **and** non-binary person

either male **or** non-binary



issue

categorical features gender in {male, female, non-binary}

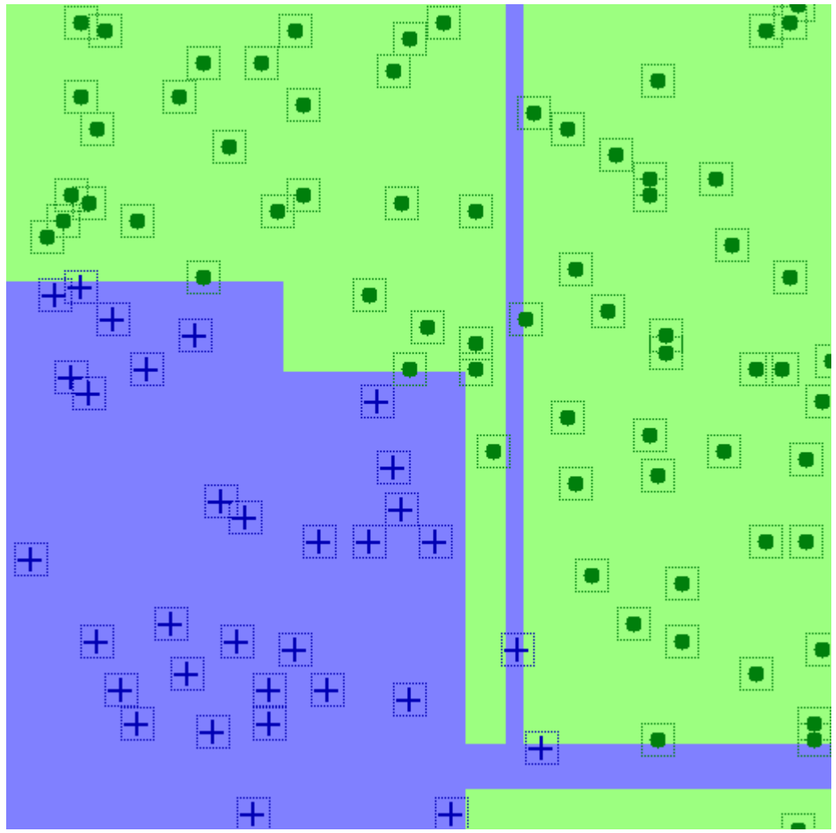
one-hot encoding is_male, is_female, is_non_binary in {0, 1}

solution

custom abstraction: box $\sqcap \Sigma_i x_i = 1$

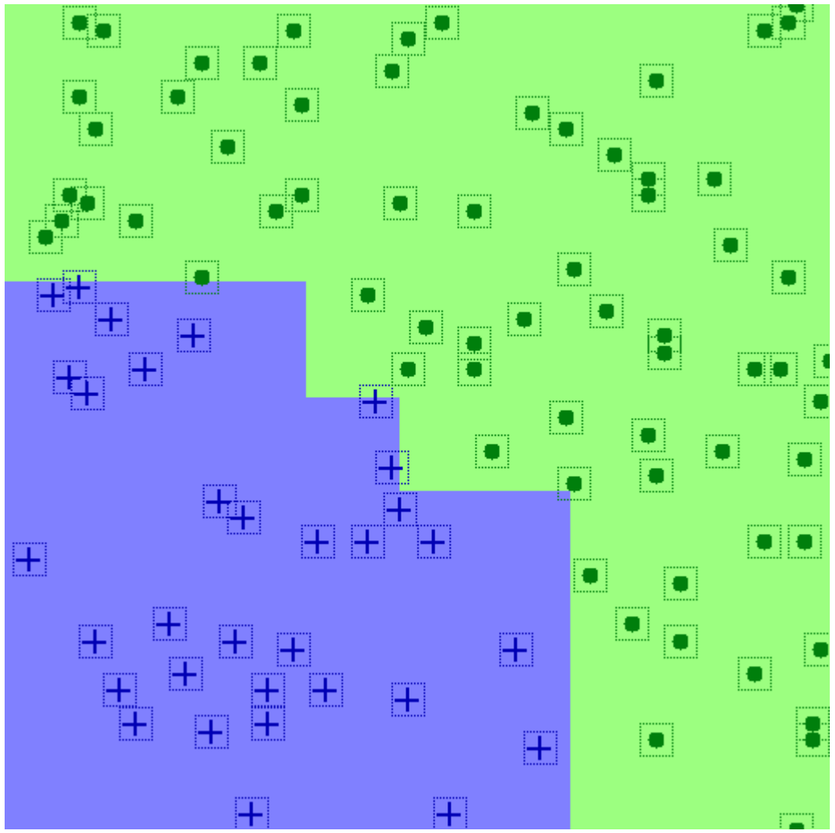
Training

trainer: $\mathbb{R}^n \times L \rightarrow (\mathbb{R}^n \rightarrow L)$



standard techniques

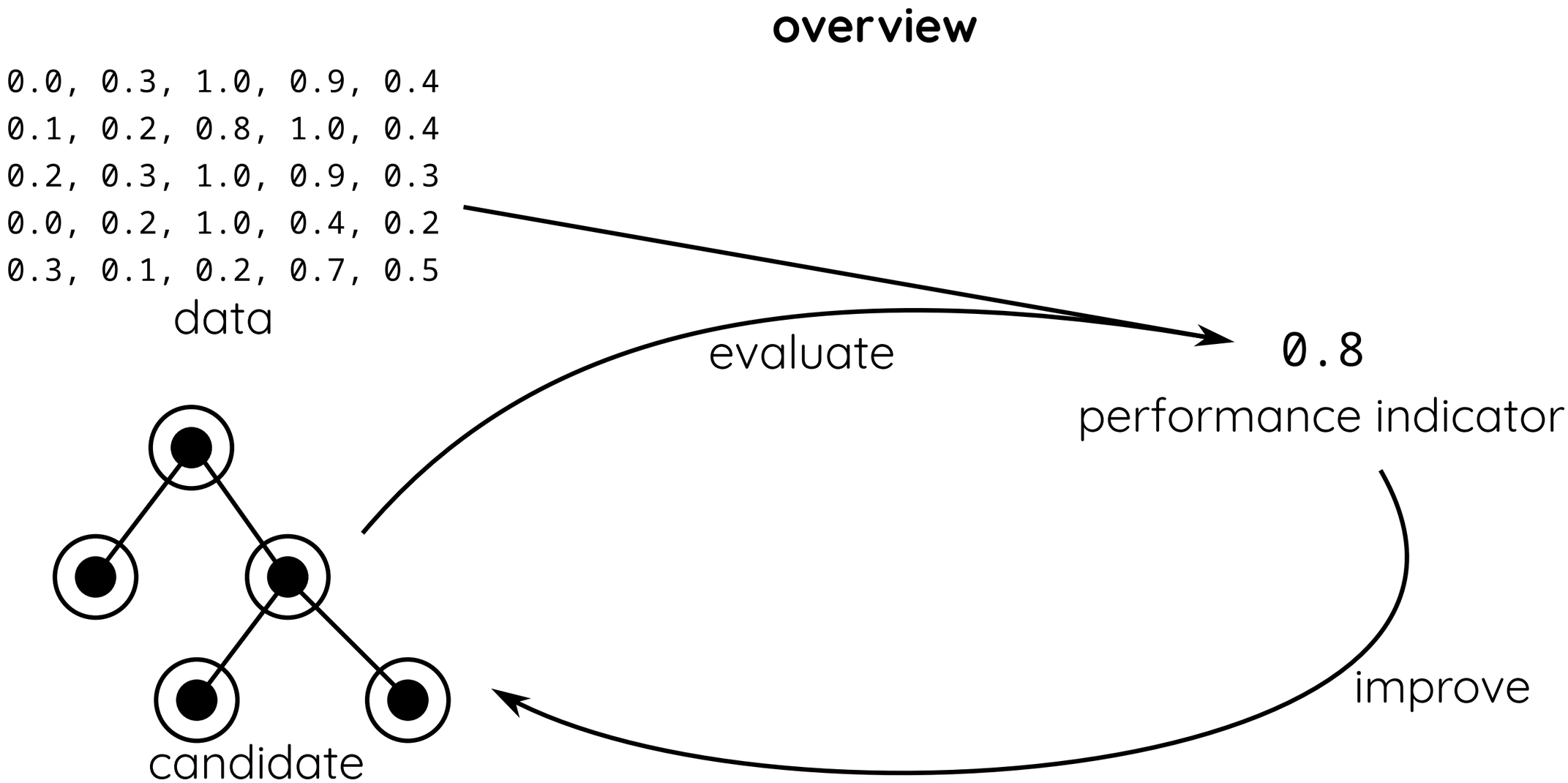
CART



stability aware

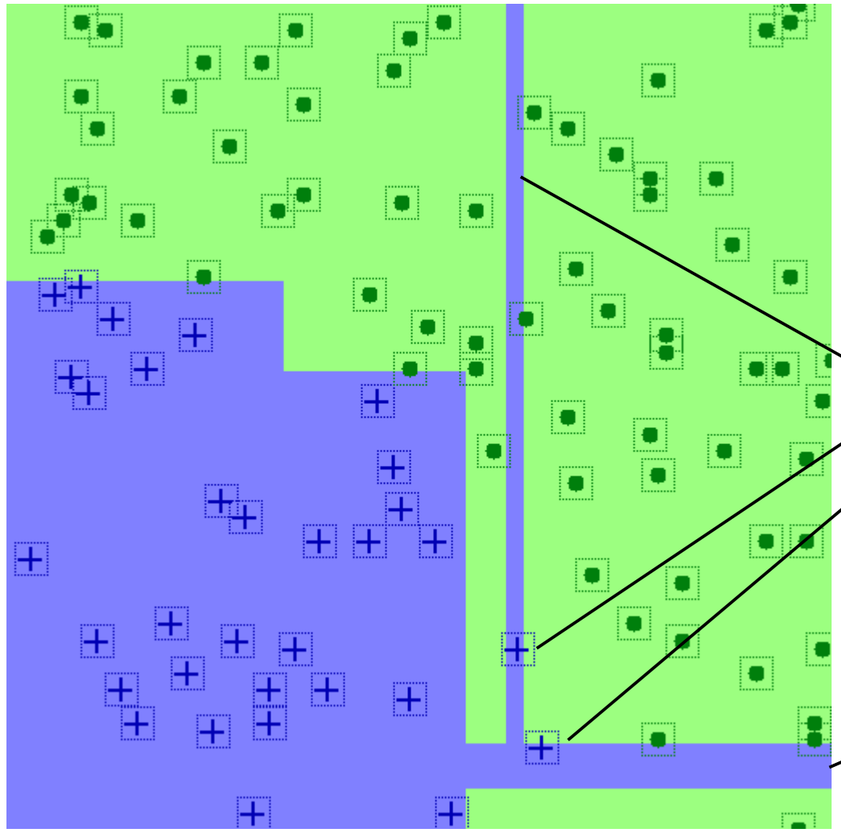
meta-silvae

Training - Standard Training



Training - Standard Training

overfitting



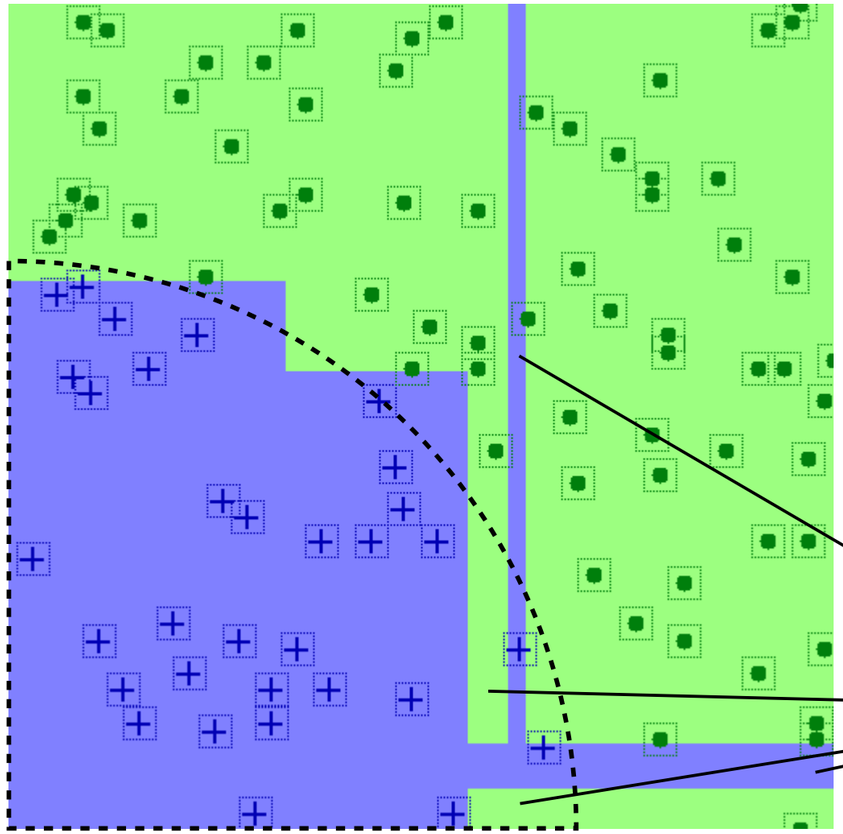
artificial strands
to "rescue" samples

100% accuracy
makes **no sense** for data

trained with standard CART algorithm

Training - Standard Training

overfitting



common symptoms:

$C = \text{trainer}(D_1)$

$\text{accuracy}_{D_1}(C) = 0.99$

$\text{accuracy}_{D_2}(C) = 0.41$

$\text{accuracy}_{D_3}(C) = 0.43$

"bad" areas

Training - Standard Training

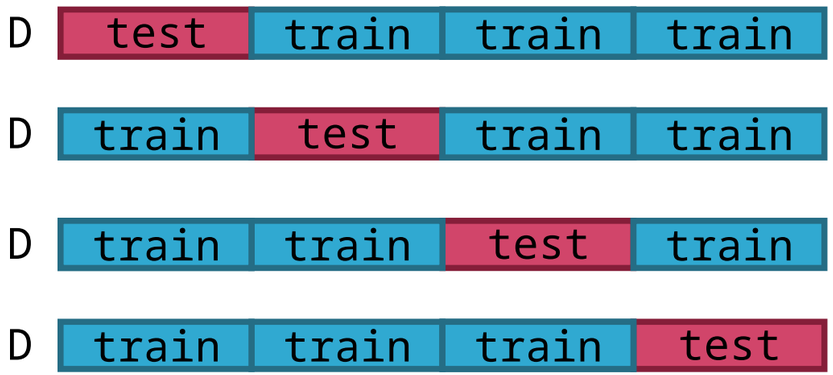
overfitting

counter measures

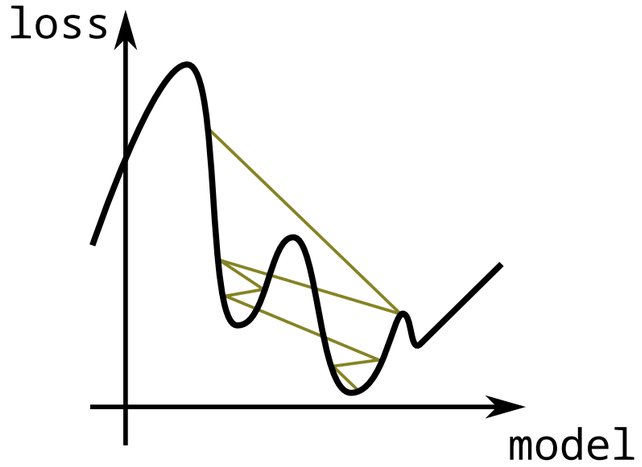
cross-validation train on D_1. improve on D_2

regularization add noise to D

model-specific dropout for ANN, stochastic gradient descend for SVM...




```
0.0, 0.3, 1.0, 0.9, 0.4
0.1, 0.4, 1.1, 0.8, 0.3
0.0, 0.2, 1.0, 1.0, 0.4
0.1, 0.3, 1.1, 1.0, 0.5
```



Training - Standard Training

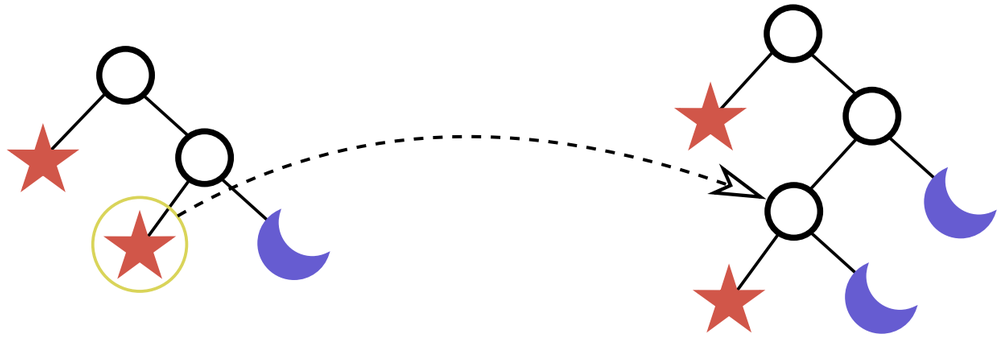
problem

standard techniques use **correctness** (or similar)
to address problems not related to correctness

Training - Decision Tree

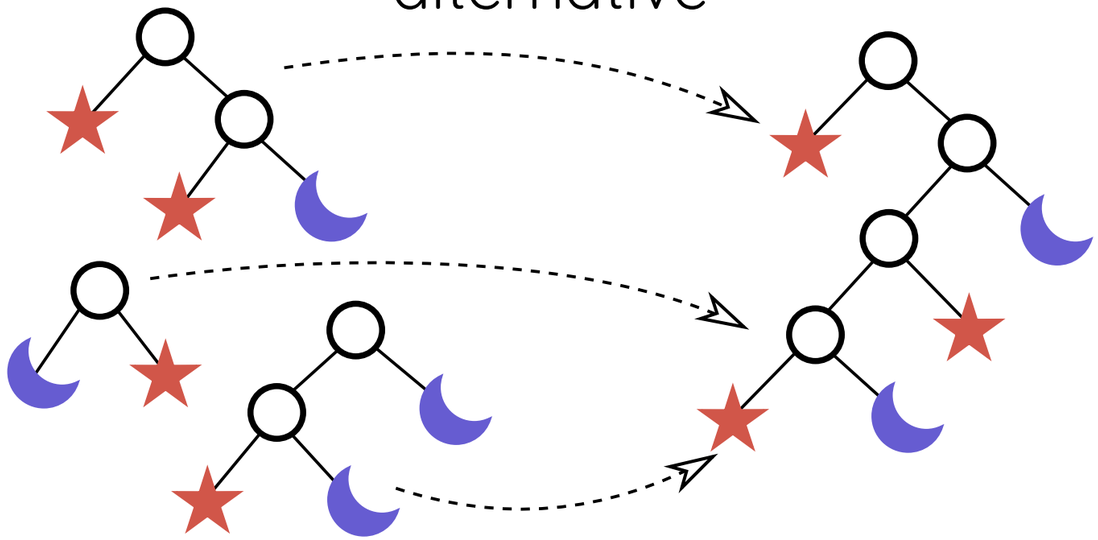
training as optimization

CART



local
greedy
accuracy-based

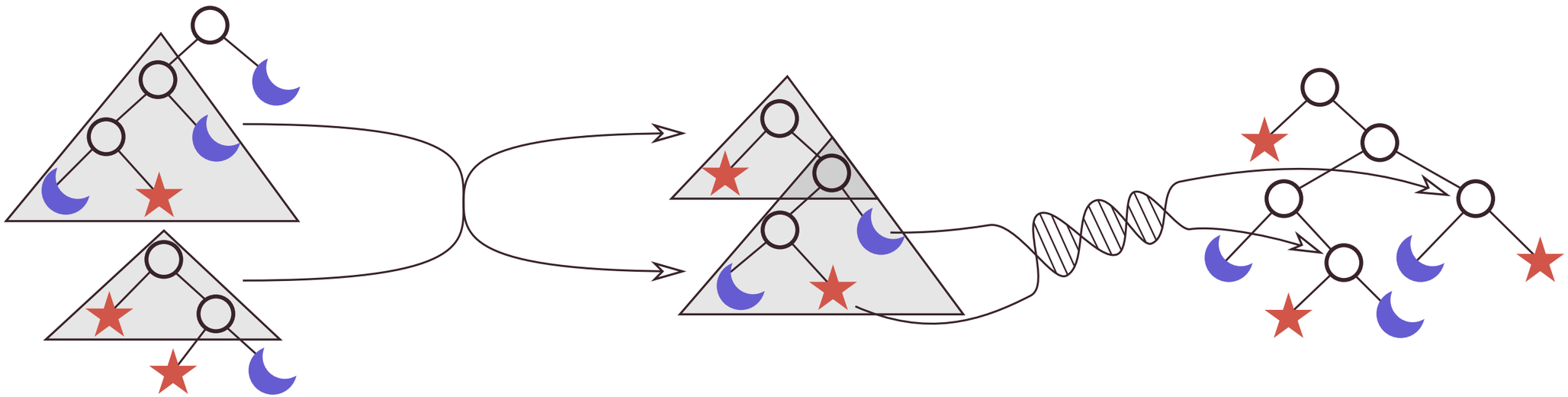
alternative



global
evolutionary
stability-aware

Training - Decision Tree

genetic algorithm

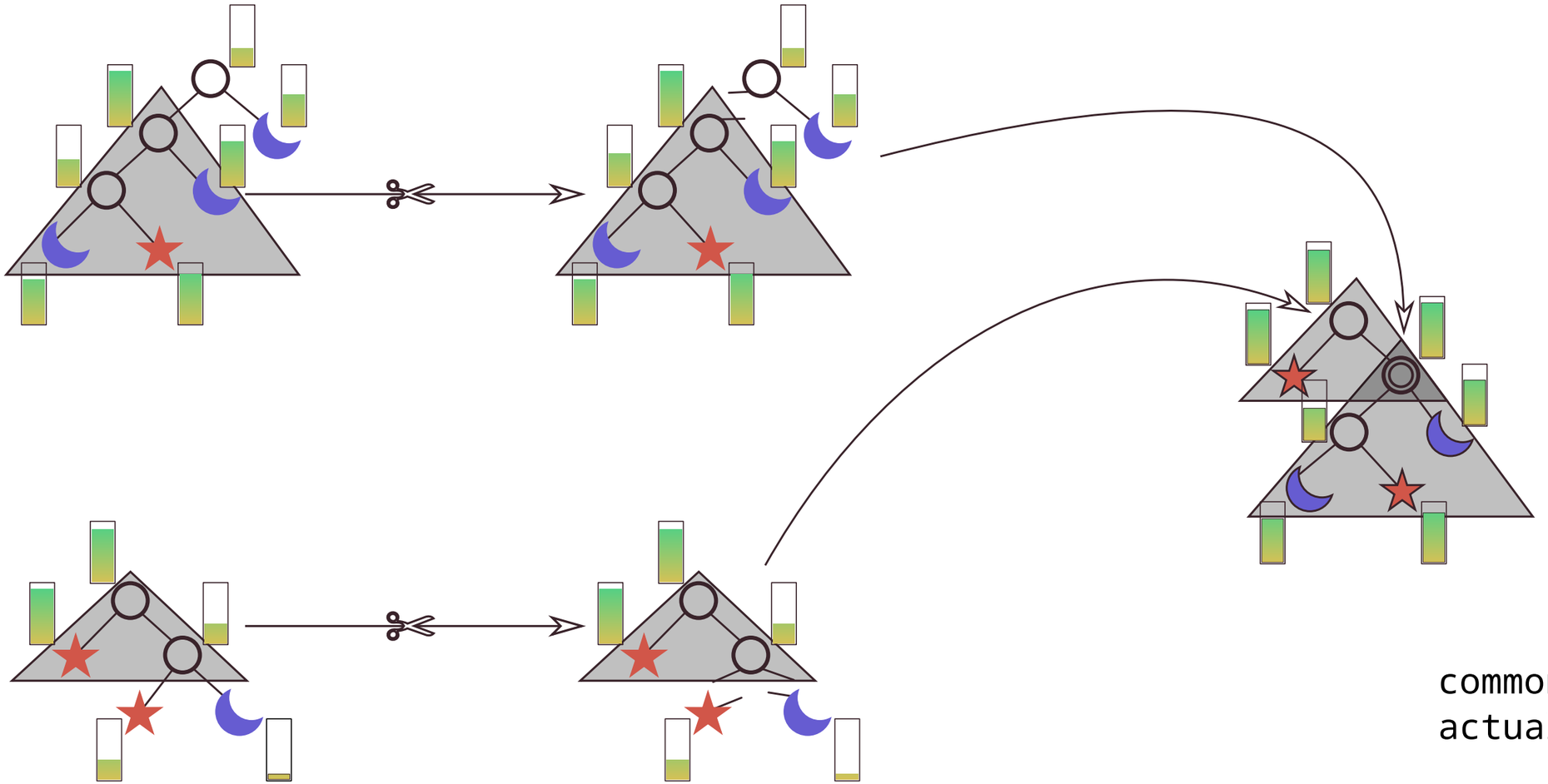


crossover $\mathsf{T} \times \mathsf{T} \to \mathsf{T}$ **mutation** $\mathsf{T} \to \mathsf{T}$

fitness $\mathsf{T} \to \mathbb{R}$

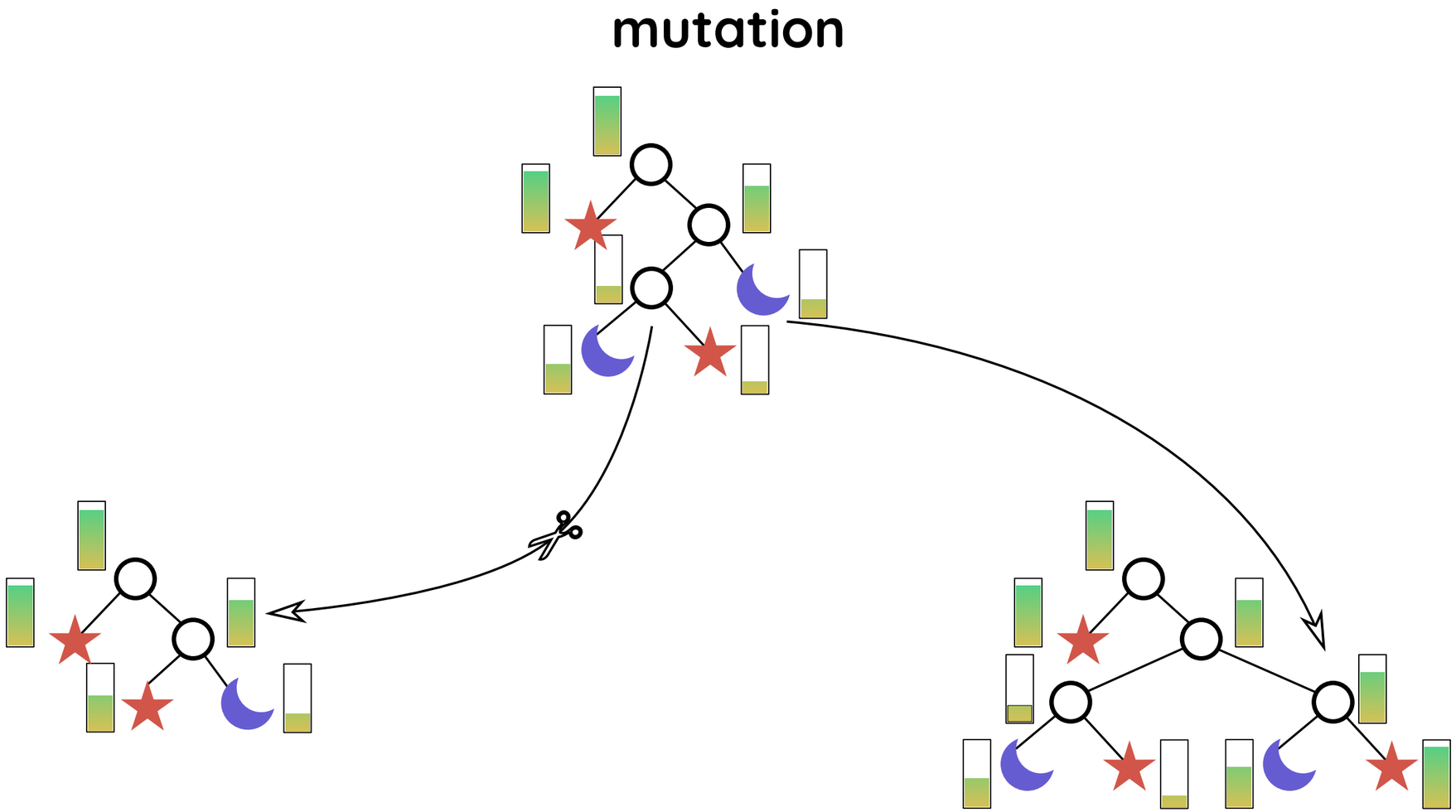
Training - Decision Tree

crossover: grafting



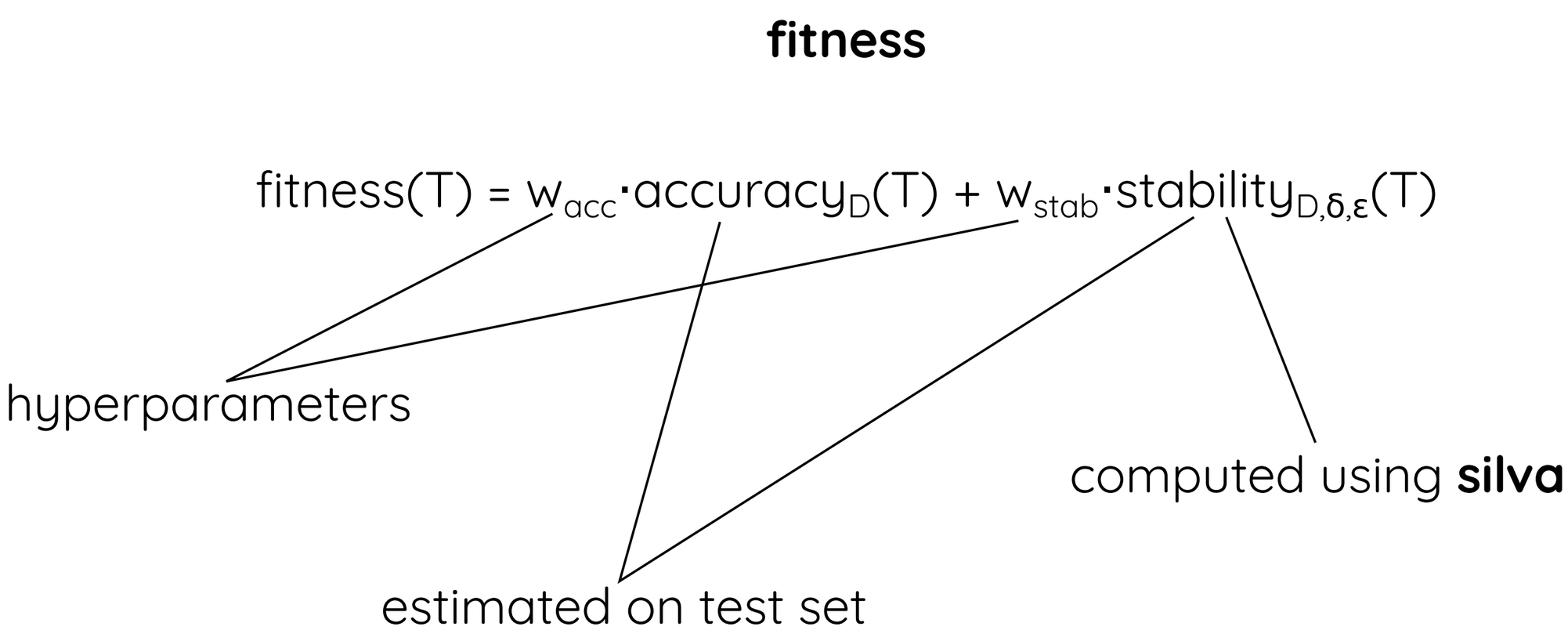
common among
actual trees

Training - Decision Tree



inspired by actual trees

Training - Decision Tree



requirements involving first derivative must be met

Training - Decision Tree

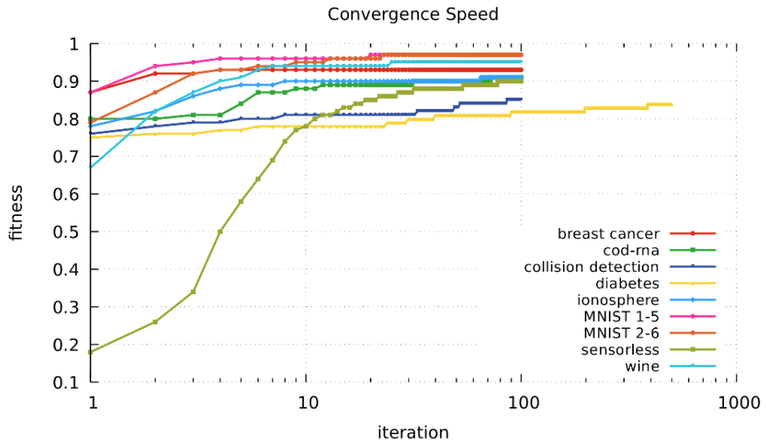
meta-silvae

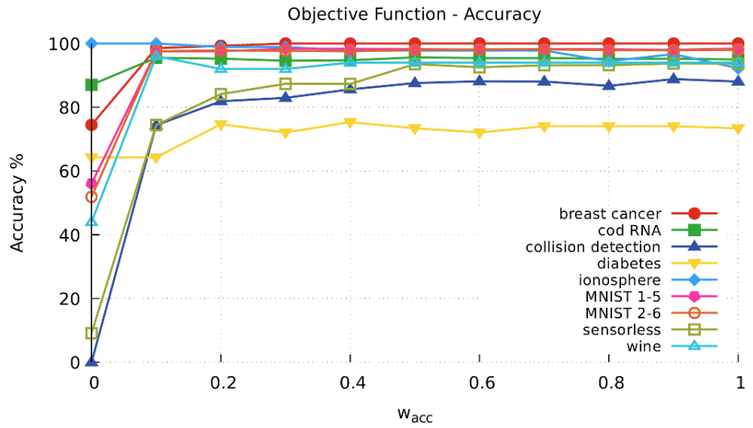
Magister **E**fficiens **T**emperat **A**rbore **silvae**

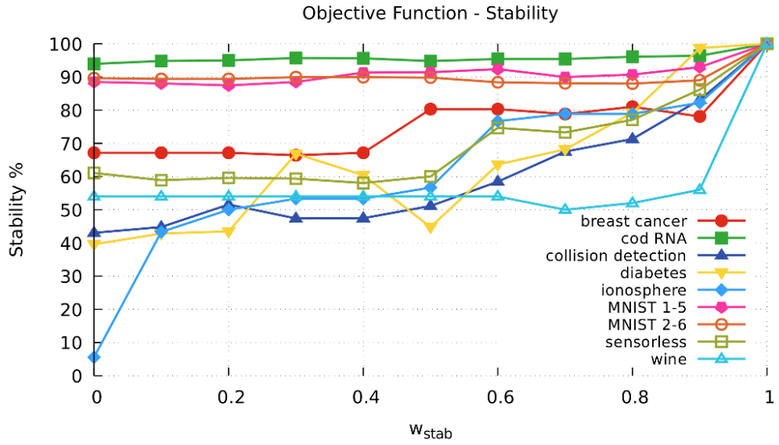
https://github.com/abstract-machine-learning/meta-silvae

Training - Decision Tree

meta-silvae







https://github.com/abstract-machine-learning/meta-silvae

Training - Decision Tree

meta-silvae

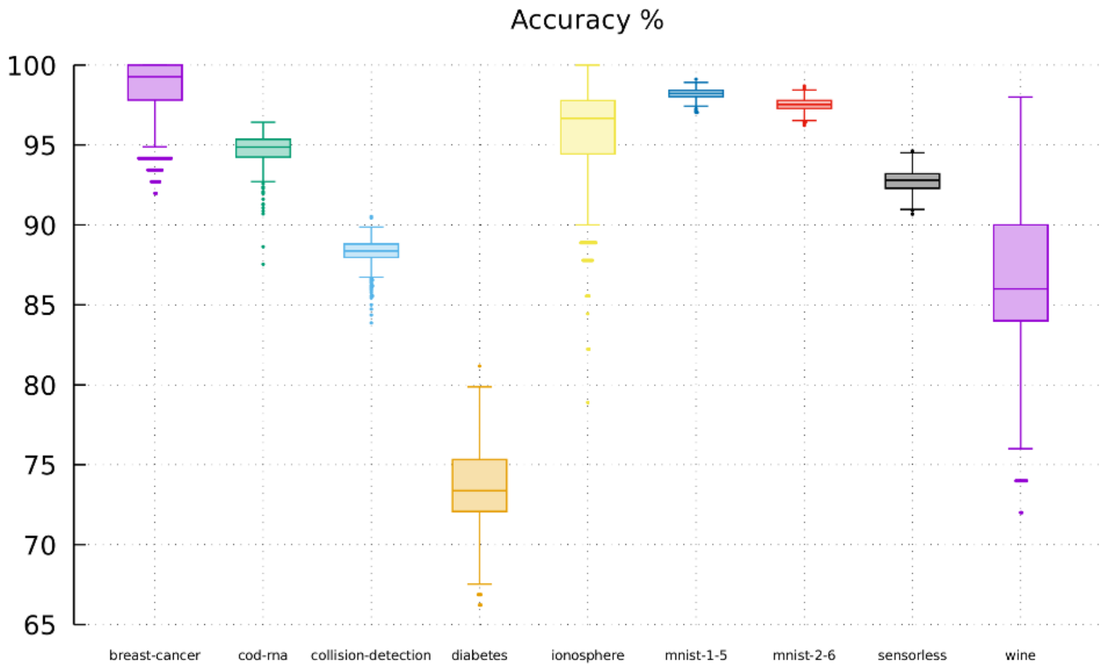
Domain	RF		MS		Stab. Gain	
	acc%	*st*%	*acc*%	*st*%	abs.	rel.
breast-cancer	**100.0**	10.2	**100.0**	**89.1**	+78.9%	8.7×
cod-rna	**97.9**	62.3	95.6	**89.9**	+27.6%	1.4×
collision-det.	**94.8**	21.0	87.5	**45.4**	+24.4%	2.2×
diabetes	**78.6**	20.8	76.0	**68.8**	+48.1%	3.3×
fashion-mnist	**86.4**	0.0	**86.4**	**46.9**	+46.9%	∞
ionosphere	96.7	0.0	**97.8**	**84.4**	+84.4%	∞
mnist	94.9	0.0	**95.6**	**81.7**	+81.7%	∞
mnist-1-5	**99.8**	19.0	97.9	**94.0**	+75.0%	4.9×
mnist-2-6	**99.2**	0.0	98.1	**88.7**	+88.7%	∞
sensorless	**99.9**	22.2	94.7	**57.4**	+35.2%	2.6×
wine	**94.0**	62.0	92.0	**68.0**	+6.0%	1.1×
Average	**94.7**	19.8	92.9	**74.0**	+54.3%	3.4×

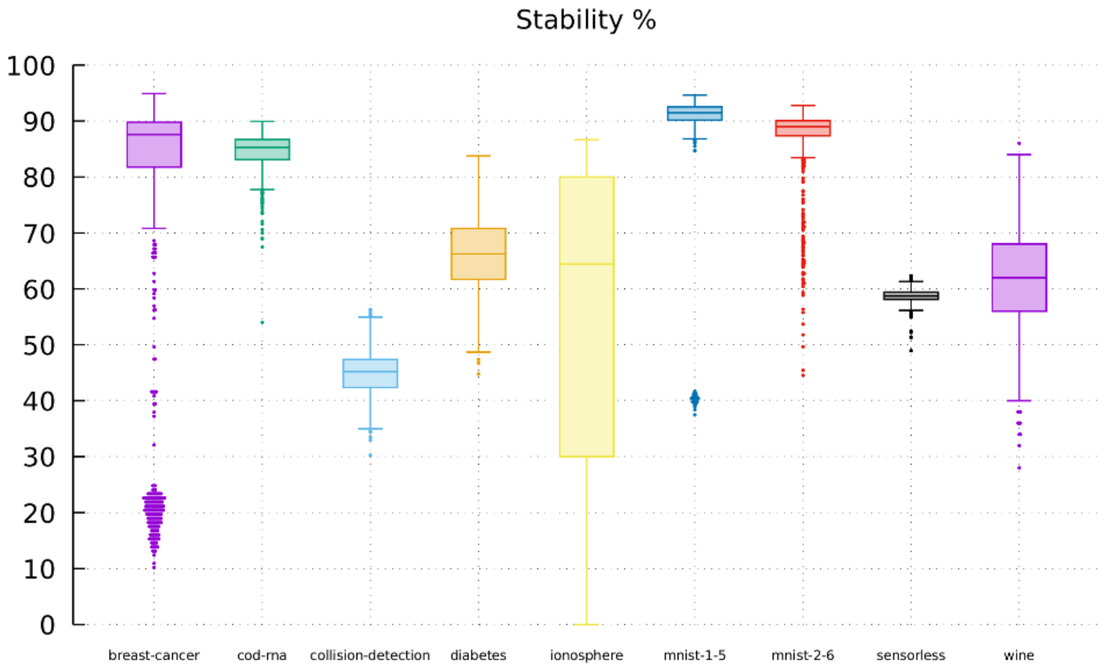
Domain	# Leaves		Eff_{acc}		Eff_{stab}	
	RF	MS	RF	MS	RF	MS
breast-cancer	641	**4**	$1.5{\cdot}10^{-3}$	$2.5{\cdot}10^{-1}$	$1.5{\cdot}10^{-4}$	$2.2{\cdot}10^{-1}$
cod-rna	89757	**85**	$1.0{\cdot}10^{-5}$	$1.1{\cdot}10^{-2}$	$6.9{\cdot}10^{-6}$	$1.0{\cdot}10^{-2}$
collision-det.	39678	**96**	$2.3{\cdot}10^{-5}$	$9.1{\cdot}10^{-3}$	$5.2{\cdot}10^{-6}$	$4.7{\cdot}10^{-3}$
diabetes	2583	**83**	$3.0{\cdot}10^{-4}$	$9.1{\cdot}10^{-3}$	$8.0{\cdot}10^{-5}$	$8.2{\cdot}10^{-3}$
fashion-mnist	**47549**	119986	$1.8{\cdot}10^{-3}$	$7.2{\cdot}10^{-4}$	$0.0{\cdot}10^{+0}$	$3.9{\cdot}10^{-4}$
ionosphere	493	**17**	$1.9{\cdot}10^{-3}$	$5.7{\cdot}10^{-2}$	$0.0{\cdot}10^{+0}$	$4.9{\cdot}10^{-2}$
mnist	**45268**	133652	$2.0{\cdot}10^{-3}$	$7.1{\cdot}10^{-4}$	$0.0{\cdot}10^{+0}$	$6.1{\cdot}10^{-4}$
mnist-1-5	3231	**30**	$3.0{\cdot}10^{-2}$	$3.2{\cdot}10^{+0}$	$5.8{\cdot}10^{-3}$	$3.1{\cdot}10^{+0}$
mnist-2-6	4881	**76**	$2.0{\cdot}10^{-2}$	$1.2{\cdot}10^{+0}$	$0.0{\cdot}10^{+0}$	$1.1{\cdot}10^{+0}$
sensorless	15704	**150**	$6.3{\cdot}10^{-5}$	$6.3{\cdot}10^{-3}$	$1.4{\cdot}10^{-5}$	$3.8{\cdot}10^{-3}$
wine	220	**15**	$4.2{\cdot}10^{-3}$	$6.1{\cdot}10^{-2}$	$2.8{\cdot}10^{-3}$	$4.5{\cdot}10^{-2}$

https://github.com/abstract-machine-learning/meta-silvae

Training - Decision Tree

meta-silvae





https://github.com/abstract-machine-learning/meta-silvae

Training - Decision Tree

meta-silvae

	Acc. %		B-Acc. %		Fairness %					
					CAT		NOISE		NOISE + CAT	
domain	RF	FATT	RF	FATT	RF	FATT	RF	FATT	RF	FATT
adult	82.7	80.8	70.2	61.8	91.7	100.0	85.4	95.2	77.5	95.2
compas	66.5	64.1	66.2	63.8	48.0	100.0	35.5	85.9	30.8	85.9
crime	80.9	79.4	80.9	79.4	86.2	100.0	31.8	75.1	32.0	75.1
german	76.5	72.0	63.6	52.5	91.5	100.0	92.0	99.5	90.0	99.5
health	85.2	77.8	83.2	73.5	7.8	99.9	47.6	97.0	2.9	97.0

fairness

https://github.com/abstract-machine-learning/meta-silvae

Training - Decision Tree

meta-silvae

	Size		Avg. Verification Time per Sample (ms)					
			CAT		NOISE		NOISE + CAT	
domain	RF	FATT	RF	FATT	RF	FATT	RF	FATT
adult	1427	43	0.0	0.0	0.0	0.0	0.0	0.0
compas	147219	75	0.3	0.1	0.4	0.1	0.6	0.1
crime	14148	11	0.1	0.1	2025.1	0.1	2028.4	0.1
german	5743	2	0.0	0.0	0.1	0.0	0.1	0.0
health	2558676	84	1.4	0.1	0.9	0.1	3.1	0.1

model size

https://github.com/abstract-machine-learning/meta-silvae

Training - Decision Tree

meta-silvae

	FATT			Standard			Hint		
domain	Acc. %	Fair. %	Size	Acc. %	Fair. %	Size	Acc. %	Fair. %	Size
adult	80.8	95.2	43	85.3	77.5	270	84.7	87.4	47
compas	64.1	85.9	75	65.9	22.2	56	65.9	22.2	56
crime	79.4	75.1	11	77.6	24.3	48	77.4	60.6	8
german	72.0	99.5	2	75.5	57.5	115	73.5	86.0	4
health	77.8	97.0	84	83.8	79.9	2371	82.2	93.6	100

meta training

https://github.com/abstract-machine-learning/meta-silvae

Training - Open Questions

role of abstraction

5.1, 3.5, 1.4, 0.2

entity
flower in a field

representation
vector

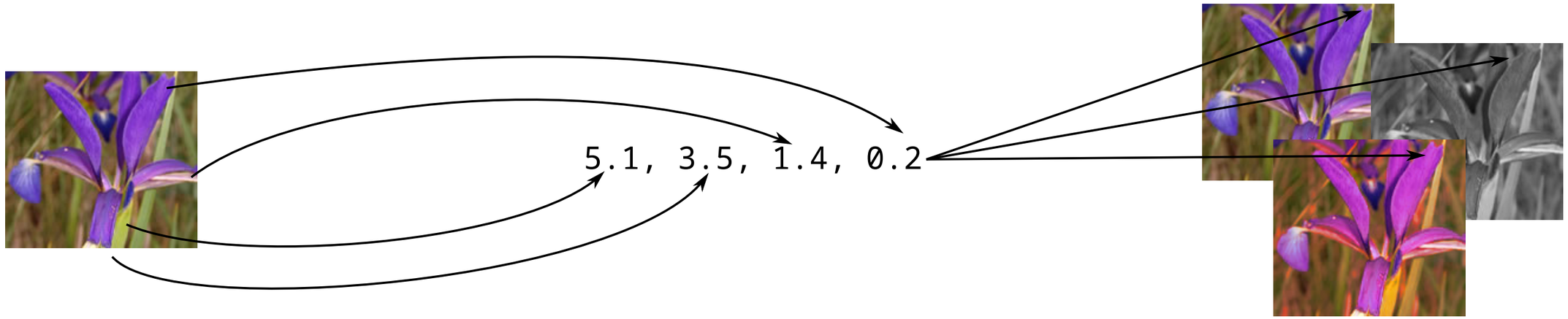
entities sharing
same **representation**

abstraction?

concretization?

Training - Open Questions

role of abstraction



entity
flower in a field

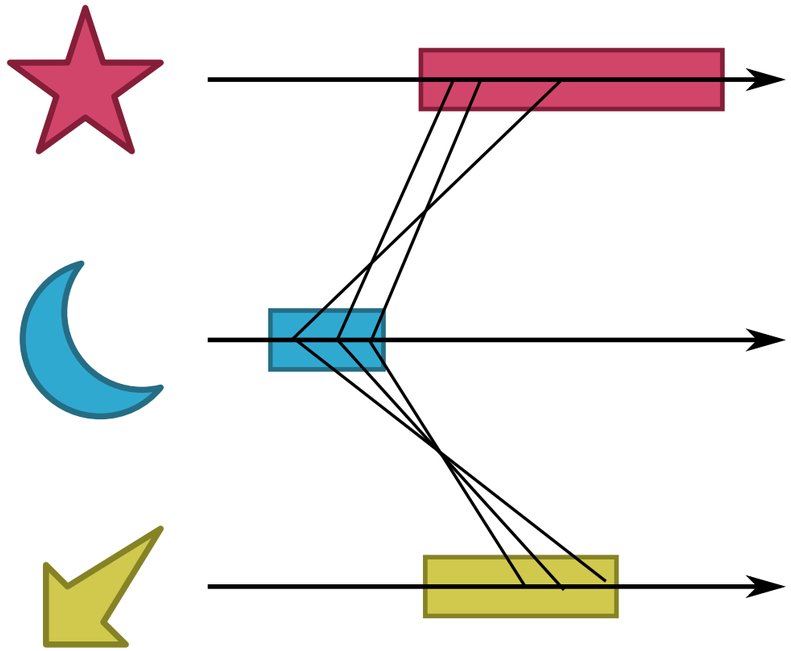
representation
vector

entities sharing
same **representation**

color? grain? scent? growing condition?

Training - Open Questions

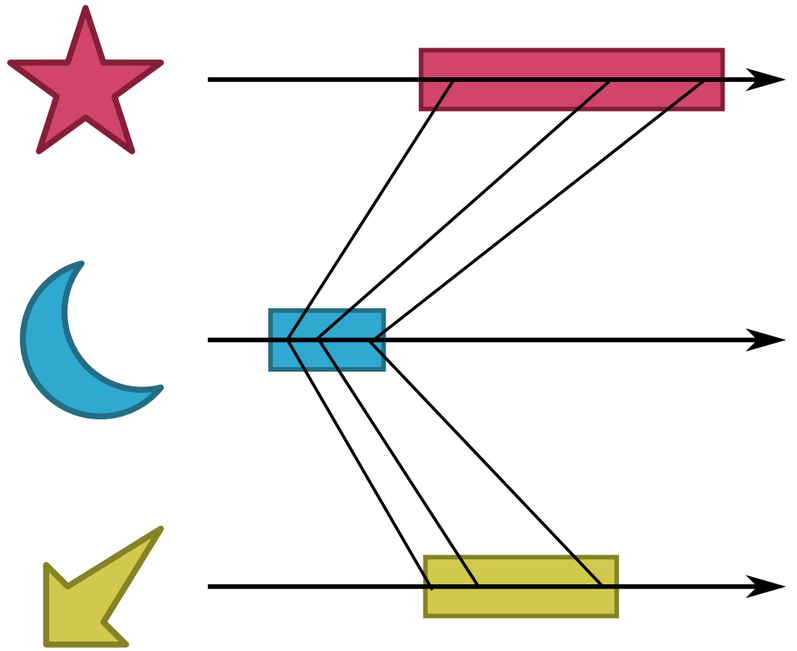
completeness for SVM



can analysis for SVM become **complete**?

minimum **refinement** of completeness?

relational information?

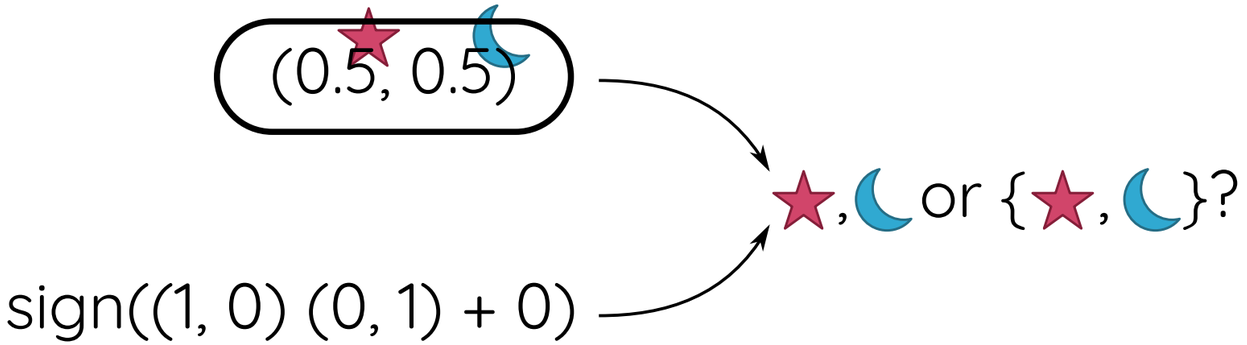


wins

wins

Training - Open Questions

L or $\wp$(L)?



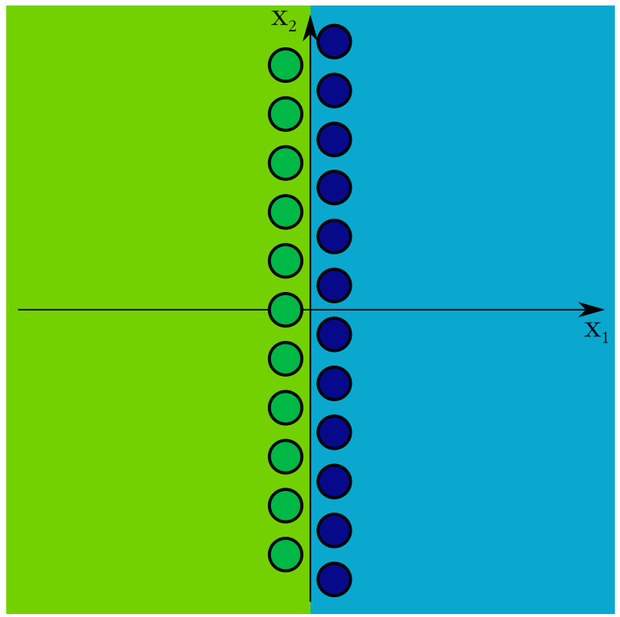
"In case that two classes have identical votes, though **it may not be a good strategy**, now we simply choose the class appearing first in the array of storing class names."
–libsvm documentation

https://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf pag. 30

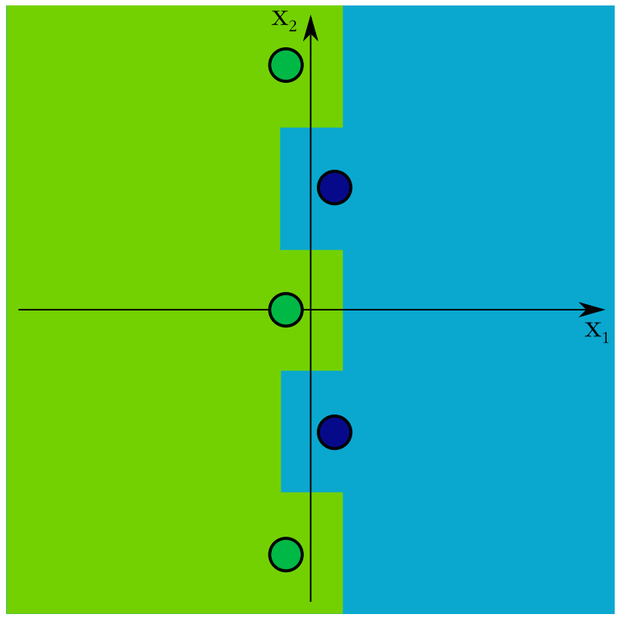
`(Development cost to handle this "not so good strategy" in SAVer: ≈ 2 weeks.)`

Training - Open Questions

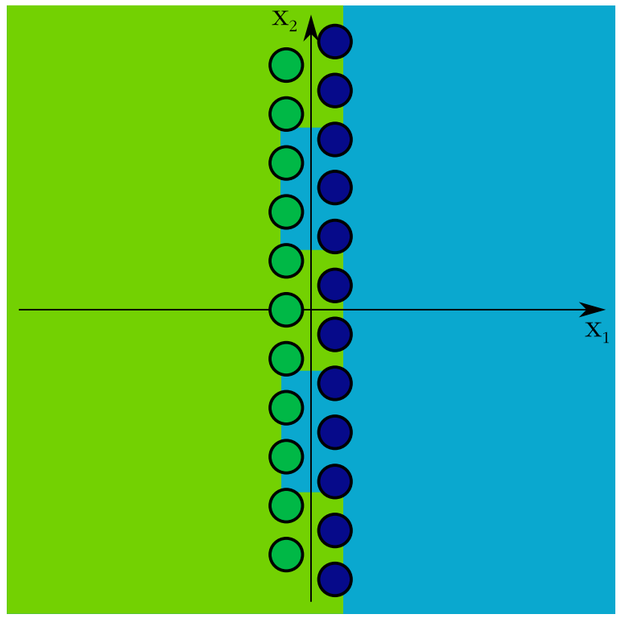
is stability good?



actual population
unstable



training set
stable

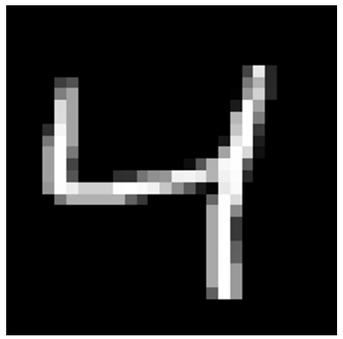


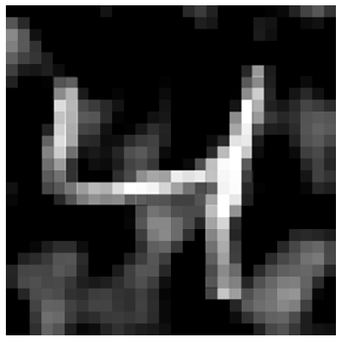
test set
incorrect

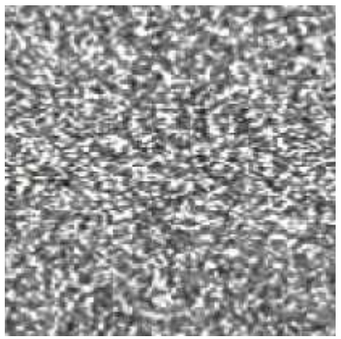
when to apply stability, and how much?

Training - Open Questions

total functions?







prediction	4	7	2
	`ok: correct`	`ok: tricked by noise`	`...just why?`

can we **abstract** away from noise?

Training - Ideas for Thesis

Abstract Interpretation for Machine Learning

give a look to "**open questions**"!

improve what's been done (new abstractions, speedup, new indicators...)

extend to different models (neural networks, KNN...)

Machine Learning for Abstract Interpretation

learn new abstract **domains**

learn abstract **transfer functions**

learn **widening**/narrowing

keep in touch!