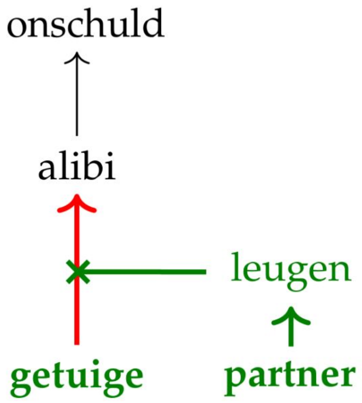
Explainable AI: Learning Arguments

Jonas Bei, David Pomerenke, Sepideh Sharbaf, Lukas Schreiner
Supervisors: Nico Roos & Pieter Collins

Agenda

- Introduction
- Learning Arguments
- Discretization
- Experiments
- Future work
- Summary

Arguments

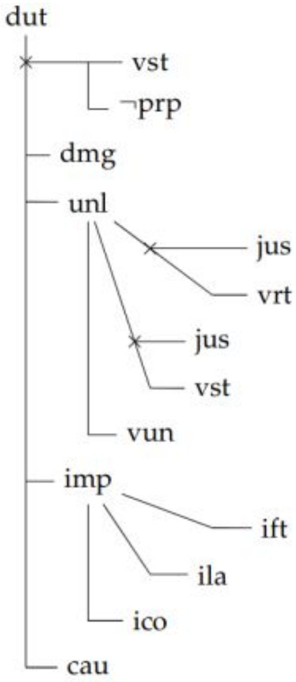


Introduction

Learning Arguments

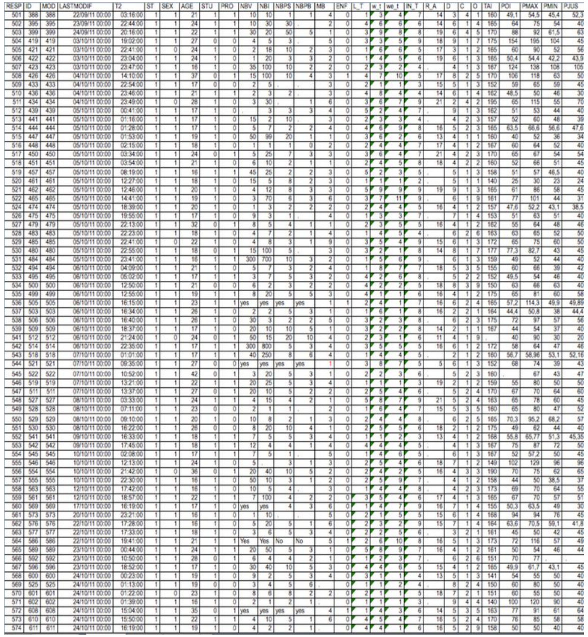
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	¬dut	¬dut	¬dut	dut	dut	dut	dut	dut	dut	dut	dut	dut	¬dut	¬dut	¬dut
¬dmg	dmg	dmg	dmg	dmg	dmg	dmg	dmg	dmg	dmg	dmg	dmg	dmg	dmg	dmg	dmg
	¬unl	unl	unl	unl	unl	unl	unl	unl	unl	unl	unl	unl	¬unl	¬unl	unl
		¬imp	imp	imp	imp	imp	imp	imp	imp	imp	imp	imp			imp
			¬cau	cau	cau	cau	cau	cau	cau	cau	cau	cau			cau
	¬vrt			vrt	vrt	vrt	¬vrt	¬vrt	¬vrt	¬vrt	¬vrt	¬vrt	vrt	¬vrt	
	¬vst			¬vst	¬vst	¬vst	vst	vst	vst	¬vst	¬vst	¬vst	¬vst	vst	vst
	¬vun			¬vun	¬vun	¬vun	¬vun	¬vun	¬vun	vun	vun	vun			
		¬ift		ift	¬ift	¬ift	ift	¬ift	¬ift	ift	¬ift	¬ift			
		¬ila		¬ila	ila	¬ila	¬ila	ila	¬ila	¬ila	ila	¬ila			
		¬ico		¬ico	¬ico	ico	¬ico	¬ico	ico	¬ico	¬ico	ico			
				¬jus	¬jus	¬jus	¬jus	¬jus	¬jus	¬jus	¬jus	¬jus	jus	jus	
				prp	prp	prp									¬prp

1 > 2 > 3 > 4 > 5 ∼ 6 ∼ 7 ∼ 8 ∼ 9 ∼ 10 ∼ 11 ∼ 12 ∼ 13 > 14 ∼ 15 ∼ 16



Explainable AI: Learning Arguments. Figure reproduced from Verheij 2018.

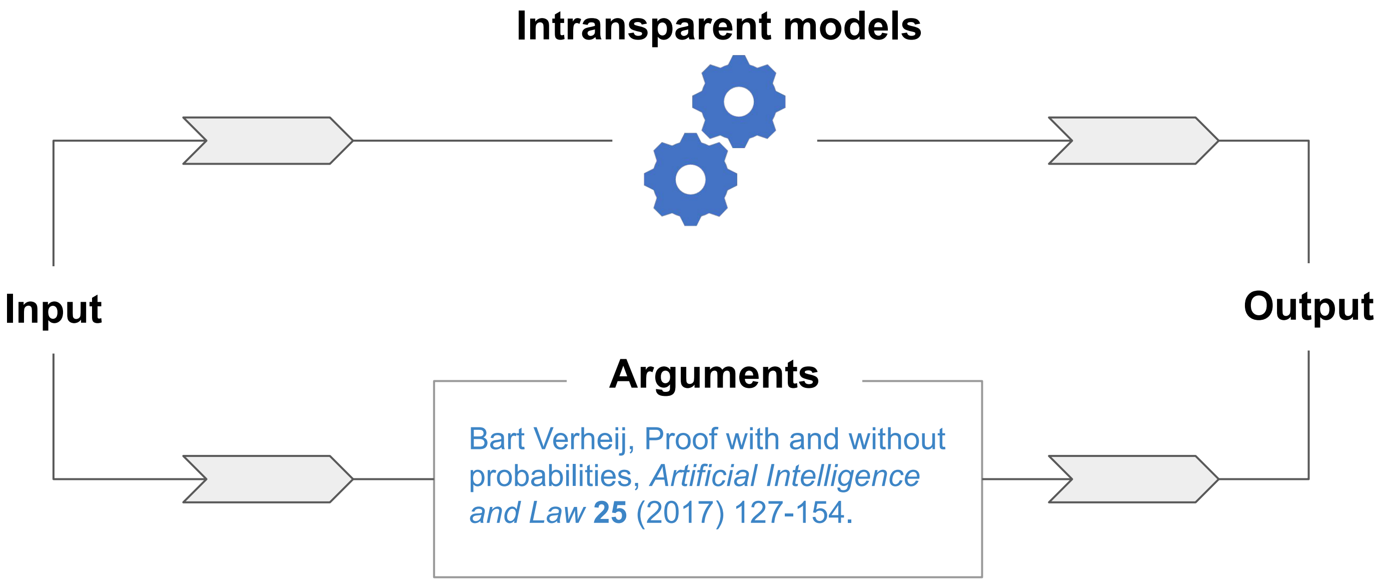
Machine learning setting



RESP	ID	MOD	LASTMODIF	T2	ST	SEX	AGE	STU	PRO	NBV	NBI	NBPS	NBPB	MB	ENF	L_T	w_t	we_t	IN_T	R_A	D	C	O	TAI	POI	PMAX	PMIN	PJUS
501	388	388	22/09/11 00:00	03:16:00	1	1	21	1	1	10	10	1	1	4	0	1	3	4	7	14	3	4	1	160	49,1	54,5	45,4	52,3
502	395	395	23/09/11 00:00	22:44:00	1	1	24	1	1	10	30	30	.	2	0	4	6	6	6	14	6	1	4	165	64	75	54	40
503	399	399	24/09/11 00:00	20:16:00	1	1	22	1	1	30	20	50	5	1	0	3	9	8	8	19	6	4	5	170	88	92	61,5	63
504	419	419	03/10/11 00:00	19:02:00	1	1	27	0	0	4	5	3	.	5	0	3	3	5	9	18	9	1	7	175	154	195	104	45
505	421	421	03/10/11 00:00	22:41:00	1	0	24	1	0	2	18	10	2	3	0	1	6	6	5	17	3	1	2	165	60	90	52	56
506	422	422	03/10/11 00:00	23:04:00	1	1	24	1	0	1	20	3	3	2	0	1	4	5	6	19	6	1	3	165	50,4	54,4	42,2	43,9
507	423	423	03/10/11 00:00	23:47:00	1	1	16	1	0	35	100	10	2	2	0	2	3	2	4	.	4	1	3	167	124	138	108	105
508	426	426	04/10/11 00:00	14:10:00	1	1	37	0	1	15	100	10	4	3	1	4	7	10	5	17	8	2	5	170	106	118	63	50
509	433	433	04/10/11 00:00	22:54:00	1	1	17	0	0	2	5	.	.	3	0	2	1	1	3	15	5	1	3	152	59	65	59	45
510	436	436	04/10/11 00:00	23:46:00	1	1	21	1	1	2	3	2	.	3	0	4	8	4	4	14	6	1	4	162	48,5	50	46	30
511	434	434	04/10/11 00:00	23:49:00	1	0	28	1	0	3	30	.	1	6	0	3	6	7	9	21	2	4	2	195	65	115	55	70
512	439	439	05/10/11 00:00	00:41:00	1	1	17	1	0	.	3	3	3	4	0	5	2	4	7	.	9	1	3	162	51	53	44	40
513	441	441	05/10/11 00:00	01:16:00	1	1	17	1	0	15	2	10	.	3	0	2	3	3	4	.	4	2	3	157	52	60	48	39
514	444	444	05/10/11 00:00	01:28:00	1	1	17	1	0	5	7	2	2	4	0	4	6	9	8	16	5	2	3	165	63,5	66,6	56,6	47,6
515	447	447	05/10/11 00:00	01:53:00	1	1	19	1	0	50	99	20	1	1	0	3	6	2	6	13	4	1	1	160	40	52	36	34
516	448	448	05/10/11 00:00	02:15:00	1	1	18	1	0	1	1	1	0	2	0	2	4	4	7	17	4	1	2	167	60	64	52	40
517	450	450	05/10/11 00:00	03:34:00	1	1	24	0	1	5	25	7	3	3	0	3	6	4	7	21	4	2	3	170	65	67	54	54
518	451	451	05/10/11 00:00	03:54:00	1	1	21	1	0	6	10	2	1	1	0	2	4	5	8	18	4	2	2	160	52	66	51	45
519	457	457	05/10/11 00:00	08:19:00	1	1	16	1	1	45	25	2	2	3	0	5	2	3	5	.	5	1	3	158	51	57	46,5	40
520	461	461	05/10/11 00:00	12:27:00	1	1	18	1	0	15	5	8	2	3	0	1	1	1	2	.	5	1	1	140	25	30	23	24
521	462	462	05/10/11 00:00	12:46:00	1	1	20	1	0	4	12	8	3	3	0	5	9	9	9	19	9	1	3	165	61	86	58	45
522	465	465	05/10/11 00:00	14:41:00	1	1	19	1	0	3	70	6	3	6	0	3	7	11	9	.	6	1	9	161	77	101	44	31
524	474	474	05/10/11 00:00	18:39:00	1	1	20	1	0	1	3	2	2	2	0	2	4	4	5	16	4	1	2	157	47,6	52,2	43,1	38,5
526	475	475	05/10/11 00:00	19:55:00	1	1	17	1	0	9	3	1	.	4	0	1	3	3	7	.	7	1	4	153	51	63	51	40
527	479	479	05/10/11 00:00	22:13:00	1	1	32	0	1	8	5	4	1	1	0	2	3	5	5	16	4	1	2	162	55	64	48	46
528	483	483	05/10/11 00:00	22:23:00	1	1	18	1	0	4	7	2	1	4	0	1	4	5	4	.	6	2	6	163	63	65	52	50
529	485	485	05/10/11 00:00	22:41:00	1	0	22	1	0	4	8	3	1	9	0	3	5	4	9	15	6	1	3	172	65	75	60	50
530	480	480	05/10/11 00:00	22:55:00	1	1	18	0	1	15	100	5	.	3	0	3	2	1	8	14	8	1	7	177	77,3	82,7	43	45
531	484	484	05/10/11 00:00	23:41:00	1	1	16	1	1	300	700	10	3	2	0	1	5	6	9	.	6	1	3	159	49	52	44	40
532	494	494	06/10/11 00:00	04:09:00	1	1	21	1	0	5	7	3	2	4	0	1	8	7	7	18	5	3	5	155	60	66	39	42
533	495	495	06/10/11 00:00	05:02:00	1	1	17	1	1	3	7	5	3	2	0	4	2	6	8	.	5	2	2	152	49,5	54	46	40
534	500	500	06/10/11 00:00	12:50:00	1	1	21	0	0	6	2	3	2	2	0	4	2	2	5	18	8	3	9	150	63	66	63	40
535	499	499	06/10/11 00:00	12:55:00	1	1	19	1	1	8	20	5	5	3	0	4	1	1	6	16	4	1	2	175	65	81	60	58
536	505	505	06/10/11 00:00	16:15:00	1	1	23	1	1	yes	yes	yes	yes	1	1	2	4	1	7	16	6	2	4	165	57,2	114,3	49,9	49,89
537	503	503	06/10/11 00:00	16:34:00	1	1	26	1	0	2	2	5	3	1	0	1	6	1	8	16	2	2	1	164	44,4	50,8	38	44,4
538	506	506	06/10/11 00:00	16:40:00	1	1	26	1	0	30	3	2	2	5	0	3	2	3	8	.	6	2	3	175	72	97	57	56
539	509	509	06/10/11 00:00	18:37:00	1	1	17	1	0	20	10	10	5	1	0	3	2	2	8	14	2	1	1	167	44	54	37	40
541	512	512	06/10/11 00:00	21:24:00	1	0	24	0	1	50	15	20	10	4	0	2	3	1	6	11	4	1	9	.	40	90	30	20
542	514	514	06/10/11 00:00	22:35:00	1	1	17	1	1	300	800	5	3	4	0	3	5	5	5	16	6	1	2	172	58	64	47	46
543	518	518	07/10/11 00:00	01:01:00	1	1	17	1	1	40	250	8	6	4	0	1	4	4	5	.	2	1	2	160	56,7	58,96	53,1	52,16
544	521	521	07/10/11 00:00	09:35:00	1	1	27	0	0	yes	yes	yes	yes	1	0	3	8	7	5	5	6	1	3	152	68	74	39	43
545	522	522	07/10/11 00:00	10:52:00	1	1	42	0	1	3	20	5	3	1	0	2	2	3	6	.	5	2	3	160	.	67	43	47
546	519	519	07/10/11 00:00	13:21:00	1	1	22	1	1	20	25	5	3	4	0	1	1	1	3	19	2	1	2	159	55	80	50	50
547	511	511	07/10/11 00:00	13:37:00	1	1	27	0	1	20	10	5	2	2	0	2	5	4	6	.	5	2	4	170	67	70	64	60
548	527	527	08/10/11 00:00	03:33:00	1	1	24	1	1	4	15	4	2	1	0	5	8	7	9	21	5	2	4	163	65	78	60	45
549	528	528	08/10/11 00:00	07:11:00	1	1	23	0	0	2	1	1	.	2	0	1	6	4	7	15	5	3	5	160	65	80	47	52
550	529	529	08/10/11 00:00	09:10:00	1	1	20	1	0	10	8	2	1	3	0	2	4	4	8	.	6	2	5	165	70,3	95,2	68,2	57
551	530	530	08/10/11 00:00	16:22:00	1	1	26	0	0	8	20	10	4	1	0	2	5	5	6	18	2	1	2	175	49	62	44	40
552	541	541	09/10/11 00:00	16:33:00	1	1	18	1	1	7	5	5	3	4	0	1	1	2	3	13	4	1	2	168	55,8	65,77	51,3	45,35
553	542	542	09/10/11 00:00	17:45:00	1	1	18	1	1	12	4	4	1	4	0	4	4	4	5	.	4	1	3	167	75	87	72	50
554	545	545	10/10/11 00:00	02:08:00	1	1	17	1	1	7	5	1	.	5	0	1	4	6	5	.	6	1	3	167	52	57,2	50	45
555	546	546	10/10/11 00:00	12:13:00	1	1	24	1	0	5	.	3	1	3	0	2	5	4	6	18	7	1	2	149	102	129	96	96
556	554	554	10/10/11 00:00	21:42:00	1	0	36	0	1	20	40	10	5	2	0	2	4	2	5	16	4	3	3	190	70	75	62	65
557	555	555	10/10/11 00:00	22:30:00	1	1	16	1	0	50	10	3	1	2	0	2	5	9	7	.	4	1	2	158	44	50	38,5	37
558	563	563	12/10/11 00:00	17:42:00	1	1	16	1	0	10	5	4	.	3	0	1	4	4	8	.	4	2	3	173	69	70	64	55
559	561	561	12/10/11 00:00	18:57:00	1	1	22	1	1	7	100	4	2	2	0	3	5	4	6	17	4	1	3	165	67	70	57	51
560	569	569	17/10/11 00:00	16:19:00	1	1	17	1	0	yes	yes	4	3	6	0	1	4	7	9	16	7	1	4	155	50,3	63,5	49	30
561	573	573	20/10/11 00:00	23:21:00	1	1	16	1	0	1	10	.	.	5	0	2	1	5	5	15	6	1	4	168	94	94	76	45
562	576	576	22/10/11 00:00	17:28:00	1	1	16	1	0	5	20	5	1	6	0	2	3	2	9	15	7	1	4	164	63,6	70,5	59,1	41,8
563	577	577	22/10/11 00:00	17:33:00	1	1	18	1	0	3	6	5	3	4	0	2	2	5	6	.	3	2	1	161	45	50	42	45
564	586	586	22/10/11 00:00	19:41:00	1	1	21	1	1	Yes	Yes	No	No	5	1	2	6	10	8	16	5	1	3	173	72	116	57	49
565	589	589	23/10/11 00:00	00:44:00	1	1	24	1	1	20	50	5	3	1	0	5	8	9	7	16	4	1	2	161	50	54	46	44
566	592	592	23/10/11 00:00	10:50:00	1	1	28	0	1	6	4	4	2	1	0	1	2	3	7	.	6	2	6	151	70	77	.	.
567	596	596	23/10/11 00:00	18:52:00	1	1	17	1	0	30	40	10	5	3	0	4	4	6	5	15	4	1	2	165	49,9	61,7	43,1	45
568	600	600	24/10/11 00:00	00:23:00	1	1	19	1	0	9	2	5	3	4	0	3	1	1	4	13	5	1	3	141	54	55	50	40
569	525	525	24/10/11 00:00	01:13:00	1	1	19	0	0	3	4	5	6	.	0	1	1	2	4	.	8	2	4	150	60	80	50	40
570	601	601	24/10/11 00:00	01:22:00	1	0	23	1	0	8	6	8	2	2	0	2	1	1	5	18	7	2	1	151	60	55	50	40
571	602	602	24/10/11 00:00	01:39:00	1	1	16	1	0	2	1	2	1	.	0	1	1	1	3	.	9	4	4	140	100	120	90	40
572	608	608	24/10/11 00:00	15:04:00	1	1	35	0	1	yes	yes	yes	yes	4	1	1	4	3	6	14	5	3	5	163	77	91	61	64
573	610	610	24/10/11 00:00	15:50:00	1	1	22	1	1	4	10	5	1	6	0	4	5	4	7	16	5	2	4	170	76	85	58	58
574	611	611	24/10/11 00:00	16:19:00	1	1	19	1	0	4	2	2	1	.	0	4	4	1	6	16	5	2	9	158	50	50	40	40

- Prediction
- Categorization

Explainable artificial intelligence



Arguments in Verheij 2017

- Coherent arguments:

  Conclusion holds sometime when the premises hold.
- Presumptively valid arguments:

  Conclusion holds in the most likely case where the premises hold.
- Conclusive arguments:

  Conclusion always holds when the premises hold.

Research questions

1. Can we **reproduce** the examples from Verheij 2017 and Verheij 2020?
2. Can we **find an (efficient) algorithm** for learning arguments with this approach? How do we decide which arguments are relevant and which ones can be discarded?
3. Can we transfer the approach to a **general attribute-value classification** machine learning setting?
4. What **existing techniques** are there for learning arguments, and how do they relate to each other? What insights can we transfer to the implementation of the approach by Verheij 2017?
5. Can we show the (in)**applicability of the approach on a real-world dataset**?
   How does the approach compare with similar rule-based approaches in terms of
   (a) accuracy and (b) runtime on real world datasets?
   What can we infer about explainability by looking at the theories generated by the algorithms?

Review of progress

1. Implementation of Verheij's approach
2. Transfer to machine learning
3. Evaluation

Review of progress

1

Implementation of Verheij's approach

2

Transfer to machine learning

Evaluation

Review of progress

1. Implementation of Verheij's approach
2. Transfer to machine learning
3. Evaluation

Learning Arguments

Learning Arguments

1. Naive search
2. Pruned search
3. HeRO algorithm
4. Decision trees

Learning Arguments

1. Naive search
2. Pruned search
3. HeRO algorithm
4. Decision trees

Require categorical data

Learning Arguments

1. Naive search
2. Pruned search
3. HeRO algorithm
4. Decision trees

Require categorical data

Hyperparameter optimization

Learning arguments
(1) Naive search

1. Enumerate all possible arguments:

$$\rightarrow c$$
$$a \rightarrow c$$
$$\neg a \rightarrow c$$
$$b \rightarrow c$$
$$\neg b \rightarrow c$$
$$a,\ b \rightarrow c$$
…

Learning arguments
(1) Naive search

1. Enumerate all possible arguments:

   $\rightarrow c$

   $a \rightarrow c$

   $\neg a \rightarrow c$

   $b \rightarrow c$

   $\neg b \rightarrow c$

   $a, \quad b \rightarrow c$

   …

2. Filtering irrelevant arguments

Learning arguments
(1) Naive search

1. Enumerate all possible arguments:

$$\left.\begin{array}{r} \longrightarrow c \\ a \longrightarrow c \\ \neg a \longrightarrow c \\ b \longrightarrow c \\ \neg b \longrightarrow c \\ a,\ \ b \longrightarrow c \\ \dots \end{array}\right\} \quad O(2^k)$$

2. Filtering irrelevant arguments

Learning arguments
(1) Naive search

Filtering irrelevant arguments

- Discard overly specific arguments
- Keep them if they are an exception
- Merge arguments with identical premises
- Eliminate arguments that do not affect the closure of arguments

Learning arguments
(1) Naive search

Filtering irrelevant arguments

- Discard overly specific arguments
- Keep them if they are an exception
- Merge arguments with identical premises
- Eliminate arguments that do not affect the closure of arguments

$$a \longrightarrow b$$
$$b \longrightarrow c$$
$$\cancel{a \longrightarrow c}$$

Learning arguments
(1) Naive search

Filtering irrelevant arguments

- Discard overly specific arguments
- Keep them if they are an exception
- Merge arguments with identical premises
- Eliminate arguments that do not affect the closure of arguments

$$a \rightarrow b$$
$$b \rightarrow c$$
$$\cancel{a \rightarrow c}$$

$$a \rightsquigarrow b$$
$$b \rightsquigarrow c$$
$$a \rightsquigarrow c$$
$$a \land b \rightsquigarrow \neg c$$

Learning arguments
(1) Naive search

```
case_model = CaseModel.fromStr([
    (3, 'inn, ¬gui, ¬evi'),
    (2, "¬inn, gui, evi, ¬evi'"),
    (1, "inn, ¬gui, evi, ¬evi'"),
    (0, "¬inn, gui, evi, evi'"),
])
```

```
evi ← ¬evi'
evi ∧ gui ← ¬inn
evi ∧ gui ∧ ¬inn ← evi'
evi ∧ ¬inn ← gui
inn ← ¬gui
inn ∧ ¬gui ← ¬evi
¬evi' ← evi ∧ inn
¬evi' ← evi ∧ ¬gui
¬gui ← inn
gui ∧ ¬evi' ∧ ¬inn ↜ evi
gui ∧ ¬inn ↜ evi ∧ ¬evi'
gui ∧ ¬inn ↜ ¬evi'
inn ∧ ¬evi ∧ ¬gui ↜
¬evi' ↜ gui
¬evi' ↜ ¬inn
```

Learning arguments
(2) Pruned search

Observations:

(A, B) coherent

$\Rightarrow$

(S, B) coherent for all S $\subseteq$ A

(A, B) conclusive

$\Rightarrow$

(S, B) conclusive for all S $\supseteq$ A

where (S, B) is coherent

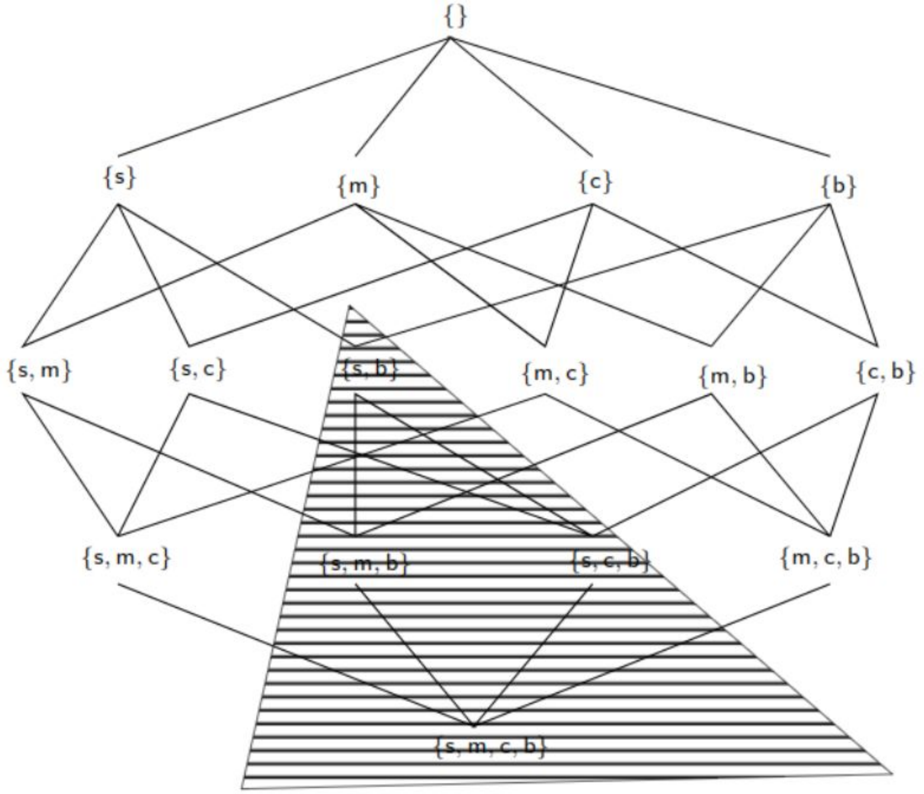
Learning arguments (2) Pruned search

Observations:

(A, B) coherent

$\Rightarrow$

(S, B) coherent for all $S \subseteq A$



Learning arguments
(2) Pruned search

nox	rm	age	dis	rad	tax	ptratio	b	lstat	medv
0.538	6.575	65.2	4.09	1	296	15.3	396.9	4.98	24
0.469	6.421	78.9	4.9671	2	242	17.8	396.9	9.14	21.6
0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
0.458	7.147	54.2	6.0622	3	222	18.7	396.9	5.33	36.2
0.458	6.43	58.7	6.0622	3	222	18.7	394.12	5.21	28.7
0.524	6.012	66.6	5.5605	5	311	15.2	395.6	12.43	22.9
0.524	6.172	96.1	5.9505	5	311	15.2	396.9	19.15	27.1
0.524	5.631	100	6.0821	5	311	15.2	386.63	29.93	16.5
0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.1	18.9
0.524	6.377	94.3	6.3467	5	311	15.2	392.52	20.45	15
0.524	6.009	82.9	6.2267	5	311	15.2	396.9	13.27	18.9
0.524	5.889	39	5.4509	5	311	15.2	390.5	15.71	21.7
0.538	5.949	61.8	4.7075	4	307	21	396.9	8.26	20.4
0.538	6.096	84.5	4.4619	4	307	21	380.02	10.26	18.2
0.538	5.834	56.5	4.4986	4	307	21	395.62	8.47	19.9
0.538	5.935	29.3	4.4986	4	307	21	386.85	6.58	23.1
0.538	5.99	81.7	4.2579	4	307	21	386.75	14.67	17.5
0.538	5.456	36.6	3.7965	4	307	21	288.99	11.69	20.2
0.538	5.727	69.5	3.7965	4	307	21	390.95	11.28	18.2
0.538	5.57	98.1	3.7979	4	307	21	376.57	21.02	13.6

```
¬high_rad <- ¬high_lstat ∧ ¬high_medv ∧ ¬high_ptratio ∧ ¬high_tax
¬high_tax <- high_b ∧ high_ptratio ∧ high_rad ∧ ¬high_lstat ∧ ¬high_medv
¬high_tax <- high_b ∧ high_ptratio ∧ ¬high_lstat ∧ ¬high_medv
¬high_tax <- high_b ∧ high_ptratio ∧ ¬high_lstat ∧ ¬high_medv ∧ ¬high_rad
¬high_tax <- high_lstat ∧ high_medv ∧ high_ptratio ∧ ¬high_b ∧ ¬high_rad
¬high_tax <- high_lstat ∧ high_medv ∧ high_rad ∧ ¬high_b ∧ ¬high_ptratio
high_b <~ high_lstat ∧ high_rad ∧ high_tax ∧ ¬high_ptratio
high_b <~ high_lstat ∧ ¬high_ptratio ∧ ¬high_rad ∧ ¬high_tax
high_b <~ high_lstat ∧ ¬high_ptratio ∧ ¬high_tax
high_b <~ high_medv ∧ high_ptratio ∧ high_rad ∧ ¬high_tax
high_b <~ high_medv ∧ high_ptratio ∧ high_tax ∧ ¬high_lstat ∧ ¬high_rad
```

Learning arguments
(2) Pruned search

nox	rm	age	dis	rad	tax	ptratio	b	lstat	medv
0.538	6.575	65.2	4.09	1	296	15.3	396.9	4.98	24
0.469	6.421	78.9	4.9671	2	242	17.8	396.9	9.14	21.6
0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
0.458	7.147	54.2	6.0622	3	222	18.7	396.9	5.33	36.2
0.458	6.43	58.7	6.0622	3	222	18.7	394.12	5.21	28.7
0.524	6.012	66.6	5.5605	5	311	15.2	395.6	12.43	22.9
0.524	6.172	96.1	5.9505	5	311	15.2	396.9	19.15	27.1
0.524	5.631	100	6.0821	5	311	15.2	386.63	29.93	16.5
0.524	6.004	85.9	6.5921	5	311	15.2	386.71	17.1	18.9
0.524	6.377	94.3	6.3467	5	311	15.2	392.52	20.45	15
0.524	6.009	82.9	6.2267	5	311	15.2	396.9	13.27	18.9
0.524	5.889	39	5.4509	5	311	15.2	390.5	15.71	21.7
0.538	5.949	61.8	4.7075	4	307	21	396.9	8.26	20.4
0.538	6.096	84.5	4.4619	4	307	21	380.02	10.26	18.2
0.538	5.834	56.5	4.4986	4	307	21	395.62	8.47	19.9
0.538	5.935	29.3	4.4986	4	307	21	386.85	6.58	23.1
0.538	5.99	81.7	4.2579	4	307	21	386.75	14.67	17.5
0.538	5.456	36.6	3.7965	4	307	21	288.99	11.69	20.2
0.538	5.727	69.5	3.7965	4	307	21	390.95	11.28	18.2
0.538	5.57	98.1	3.7979	4	307	21	376.57	21.02	13.6

```
¬high_rad <- ¬high_lstat ∧ ¬high_medv ∧ ¬high_ptratio ∧ ¬high_tax
¬high_tax <- high_b ∧ high_ptratio ∧ high_rad ∧ ¬high_lstat ∧ ¬high_medv
¬high_tax <- high_b ∧ high_ptratio ∧ ¬high_lstat ∧ ¬high_medv
¬high_tax <- high_b ∧ high_ptratio ∧ ¬high_lstat ∧ ¬high_medv ∧ ¬high_rad
¬high_tax <- high_lstat ∧ high_medv ∧ high_ptratio ∧ ¬high_b ∧ ¬high_rad
¬high_tax <- high_lstat ∧ high_medv ∧ high_rad ∧ ¬high_b ∧ ¬high_ptratio
high_b <~ high_lstat ∧ high_rad ∧ high_tax ∧ ¬high_ptratio
high_b <~ high_lstat ∧ ¬high_ptratio ∧ ¬high_rad ∧ ¬high_tax
high_b <~ high_lstat ∧ ¬high_ptratio ∧ ¬high_tax
high_b <~ high_medv ∧ high_ptratio ∧ high_rad ∧ ¬high_tax
high_b <~ high_medv ∧ high_ptratio ∧ high_tax ∧ ¬high_lstat ∧ ¬high_rad
```

Parameters:
- maximum size of premises
- maximum depth of exceptions

HeRO algorithm

- Incremental approach

- Information gain
- Maximum information gain

An algorithm for the induction of defeasible logic theories from databases

Benjamin Johnston **Guido Governatori**

School of Information Technology and Electrical Engineering
The University of Queensland
Brisbane, Queensland, Australia
Email: superhero@benjaminjohnston.com.au, guido@itee.uq.edu.au

Abstract

Defeasible logic is a non-monotonic logic with applications in rule-based domains such as law. To ease the development and improve the accuracy of expert systems based on defeasible logic, it is desirable to automatically induce a theory of the logic from a training set of precedent data. Empirical evidence suggests that minimal theories that describe the training set tend to be more faithful representations of reality. We show via transformation from the hitting set problem that this global minimization problem is intractable, belonging to the class of NP optimisation problems. Given the inherent difficulty of finding the optimal solution, we instead use heuristics and demonstrate that a best-first, greedy, branch and bound algorithm can be used to find good theories in short time. This approach displays significant improvements in both accuracy and theory size as compared to recent work in the area that post-processed the output of an Apriori association rule-mining algorithm, with comparable execution times.

Keywords: Defeasible Logic, Machine Learning, Association Rules

1 Introduction

Expert and decision support systems are slowly making inroads into the legal community, but unfortunately they currently appear to be limited in terms of either the difficulty of their construction or their inability to justify their reasoning processes to the user. Existing systems could be roughly classified into two broad categories: expert systems that are constructed by manual encoding of knowledge (Zeleznikow & Hunter 1994), and classification tools that are automatically trained from precedent data using machine learning or data mining techniques (Zeleznikow & Stranieri 1997, Brüninghaus & Ashley 1999). Unsurprisingly, the expense involved in employing human experts and the difficulty that experts have in expressing the reasoning behind their "intuition" can eliminate the option of building expert systems in spite expert systems, but facilitating construction via automatic induction.

Recent work in the field of non-monotonic logics suggests the suitability of the formalism as an underlying model for such reasoning, that turns out (as we will show) to be conducive to automatic induction. Non-monotonic logics, such as defeasible logic, were originally developed to simplify reasoning with incomplete information (Ginsberg 1993). In contrast to monotonic logics whereby a conclusion of a theory remains valid irrespective of how many assertions are added to the theory, non-monotonic logics can reach tentative conclusions that may be overridden (and replaced with a contrary conclusion) in light of additional information. Defeasible logic is one of many non-monotonic logics in use, but is particularly desirable for use in information systems because it matches the non-monotonicity of legal reasoning and is computationally efficient without sacrificing too much expressiveness. The extension of a defeasible logic theory has been shown to be computable in linear time (Antoniou, Billington, Governatori & Maher 2001, Maher 2001), as opposed to the NP-hardness or even undecidability of most non-monotonic and monotonic logics (Prakken 1997, Ginsberg 1993). While some expressiveness is sacrificed in using defeasible logic over first order logic, it still remains quite suited to legal domains. Antoniou et al (1999) have demonstrated the extremely high correspondence between regulatory documents and their equivalent expression as defeasible logic theories, in some cases the correspondence is almost 1-1 between sentences in legal documents and logical rules.

2 Defeasible Logic

In this section we will present a formal explanation of defeasible logic. Because we are focussing our attention to a specific application of defeasible logic, for simplicity our terminology slightly devi-

HeRO algorithm

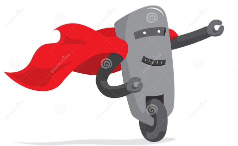
rule 2

HeRO algorithm

rule 2

rule 3

HeRO algorithm

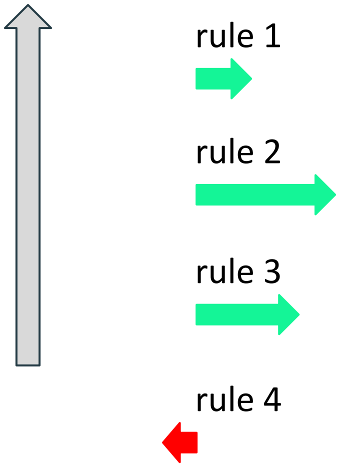


rule 1

rule 2

rule 3

HeRO algorithm



HeRO algorithm

rule 1

rule 2

rule 3

Learning arguments
(4) Decision trees

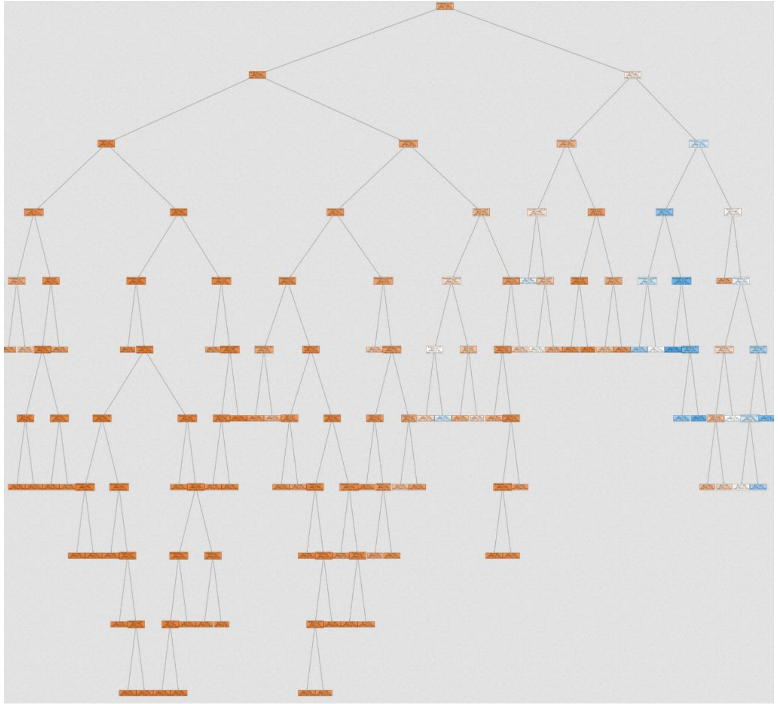
- Efficient on large data sets
- Entropy-based discretization
- Small number of arguments thanks to pruning

- No exceptions

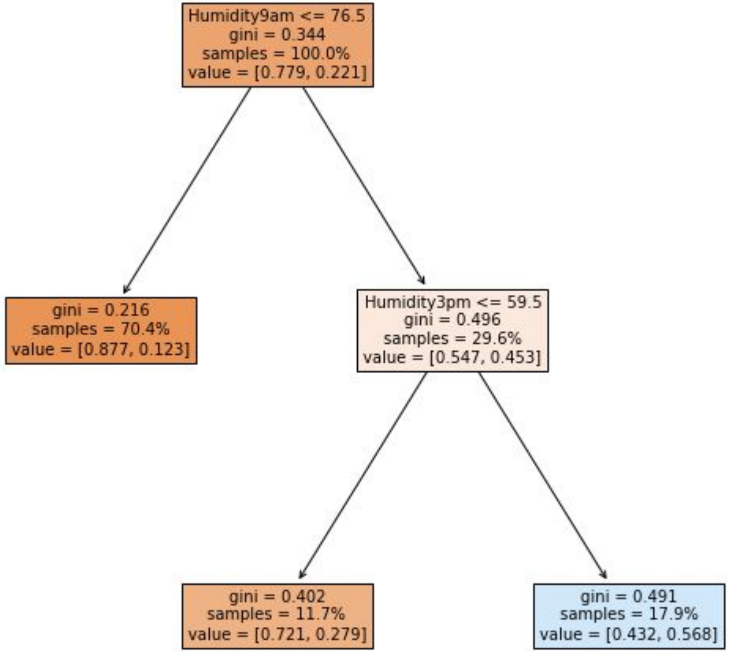
Learning arguments
(4) Decision trees

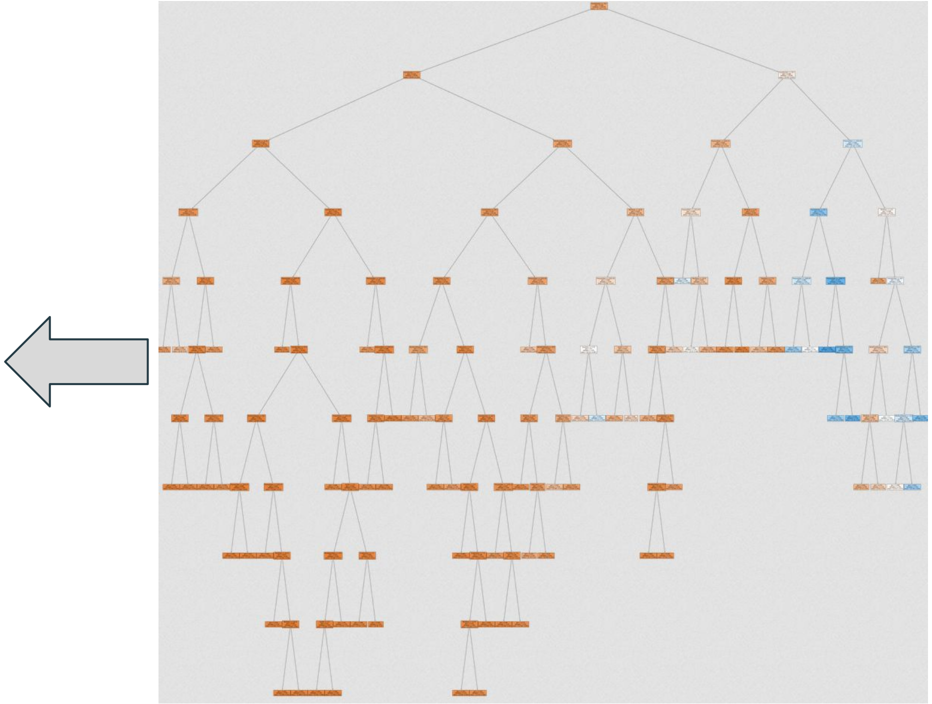
	MinTemp	MaxTemp	Evaporation	Sunshine	WindGustSpeed	WindSpeed9am	WindSpeed3pm	Humidity
6049	17.9	35.2	12.0	12.3	48.0	6.0	20.0	2
6050	18.4	28.9	14.8	13.0	37.0	19.0	19.0	3
6052	19.4	37.6	10.8	10.6	46.0	30.0	15.0	4
6053	21.9	38.4	11.4	12.2	31.0	6.0	6.0	3
6054	24.2	41.0	11.2	8.4	35.0	17.0	13.0	1
...	...	...	...	...	...	...	...	
142298	19.3	33.4	6.0	11.0	35.0	9.0	20.0	6
142299	21.2	32.6	7.6	8.6	37.0	13.0	11.0	5
142300	20.7	32.8	5.6	11.0	33.0	17.0	11.0	4
142301	19.5	31.8	6.2	10.6	26.0	9.0	17.0	6
142302	20.2	31.7	5.6	10.7	30.0	15.0	7.0	7

56420 rows × 92 columns

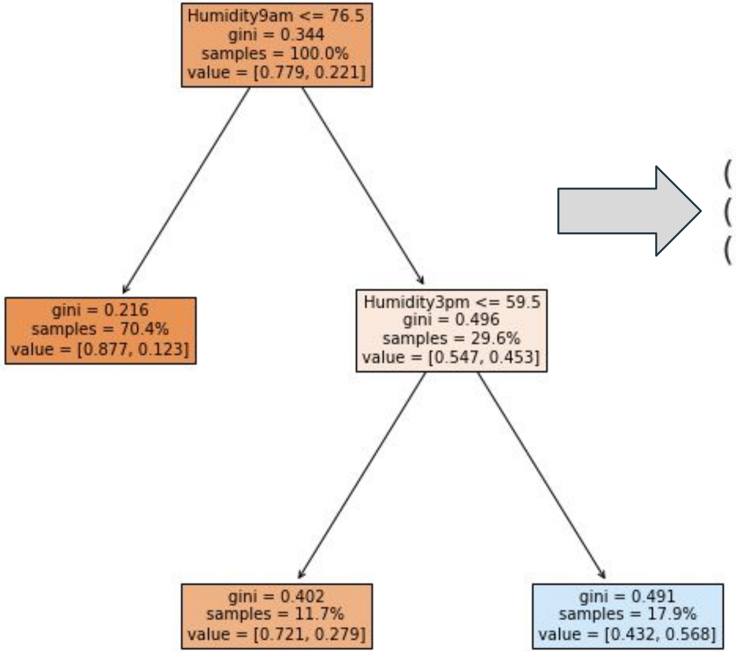


Learning arguments
(4) Decision trees



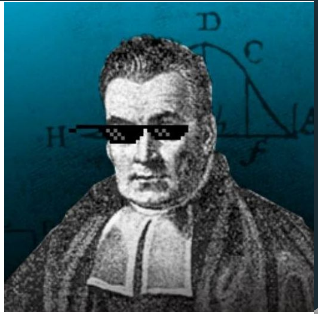


Learning arguments
(4) Decision trees



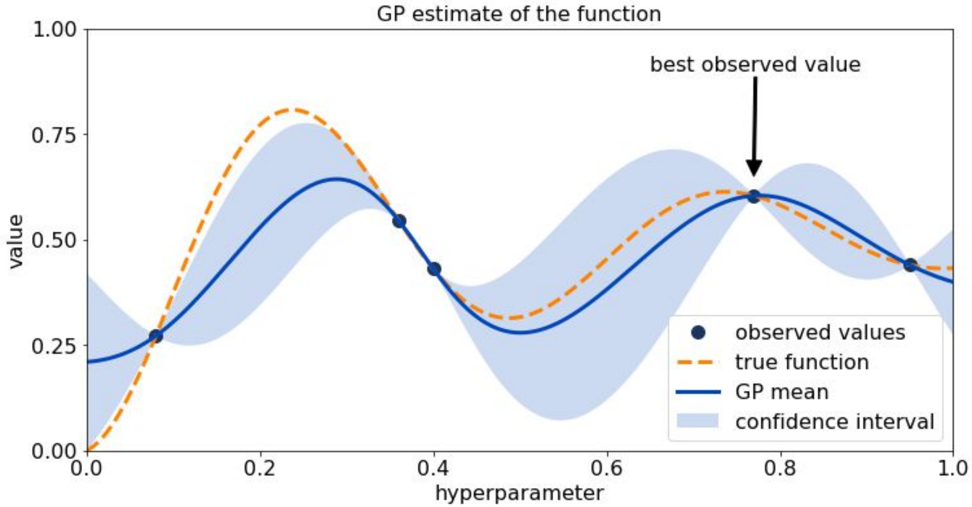

```
(Humidity9am <= 76.5) -> NoRain
(Humidity9am > 76.5), (Humidity3pm > 59.5) -> Rain
(Humidity9am > 76.5), (Humidity3pm <= 59.5) -> NoRain
```

Bayesian Optimization

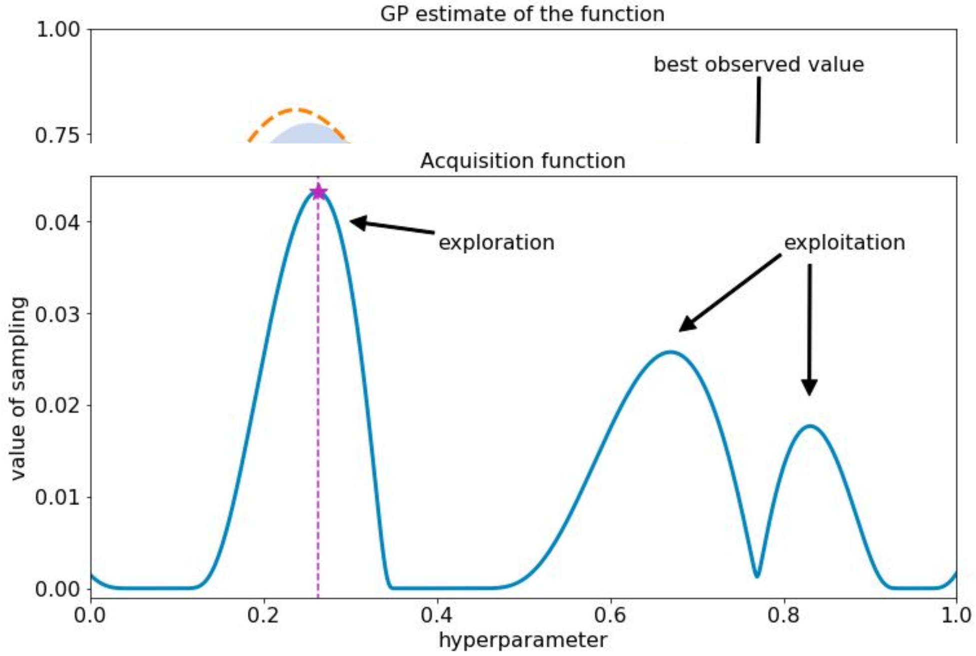


- Sampled points are used to estimate the objective function (prior)
- Points are sampled using an acquisition function and the prior is updated
  - Acquisition function balances exploration & exploitation via uncertainty in the prior
- Prior is updated
- After a given number of iterations, use a numerical method to find the estimated optimum

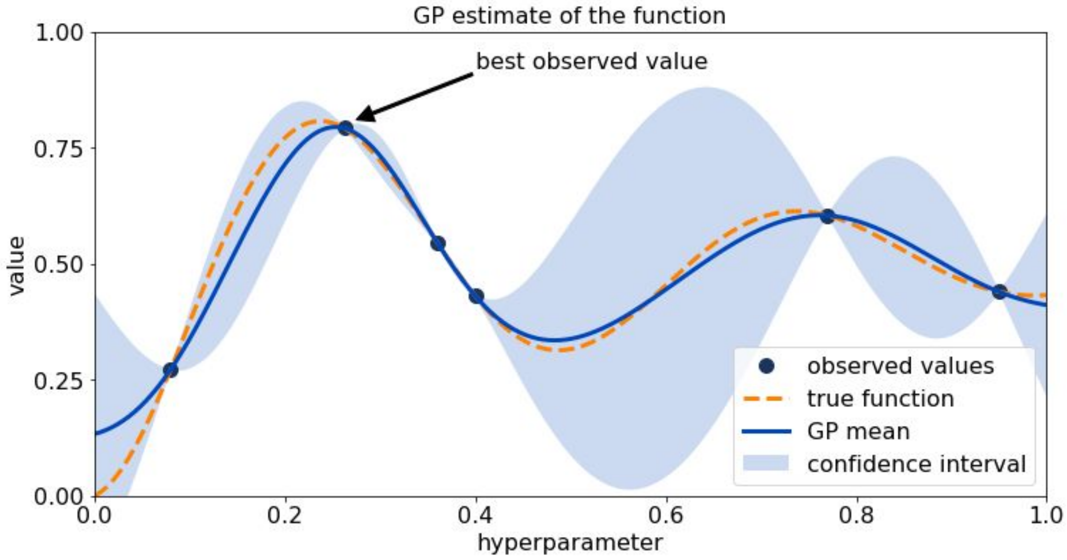
Bayesian Optimization



Bayesian Optimization



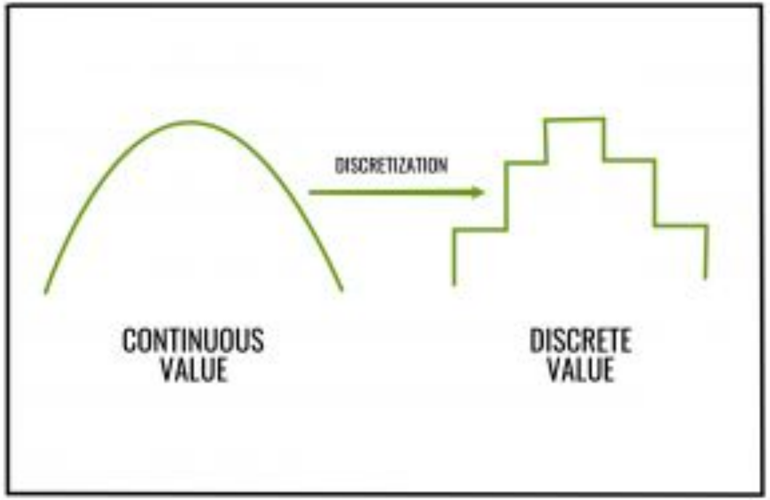
Bayesian Optimization



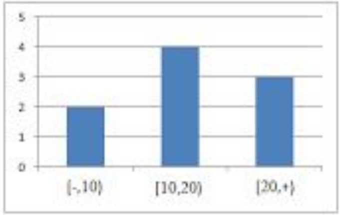
Discretization Algorithms

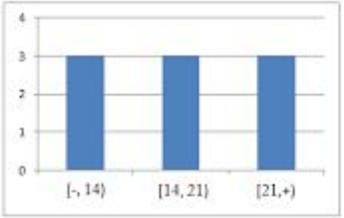
Techniques used

With the exception of decision trees, the rule-mining algorithms cannot be trained on continuous data. Therefore, in order to apply the rule-mining algorithms to datasets, we must rely on data discretization techniques to preprocess the data before mining the rules.



Discretization Techniques



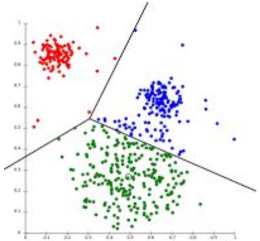


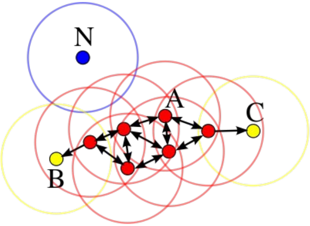
Equal width binning

Equal depth binning

K-means clustering

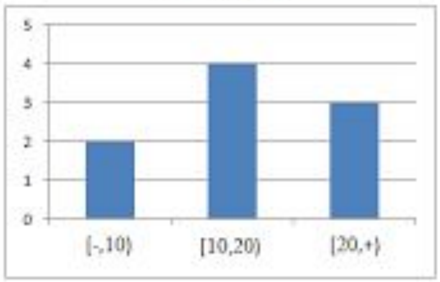
DBSCAN





Discretization Techniques

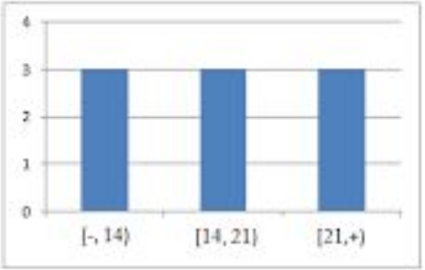
Equal Width Binning



- This algorithm is a comparatively simple binning technique.
- Assuming each cluster, same diameter, each of bins have size max-min/K.
- To discretize, values are assigned to the respective bin they fall into.

Discretization Techniques

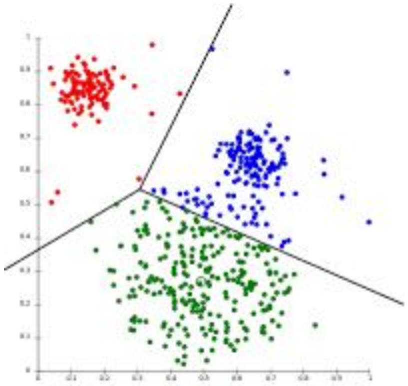
Equal Depth Binning



- Equal-depth or equal-frequency binning is another simple discretization approach.
- Each bin approximately holds the same number of instances.
- This is done by sorting the values of the feature.

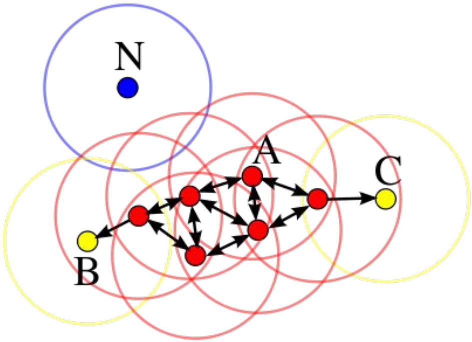
Discretization Techniques

k-means



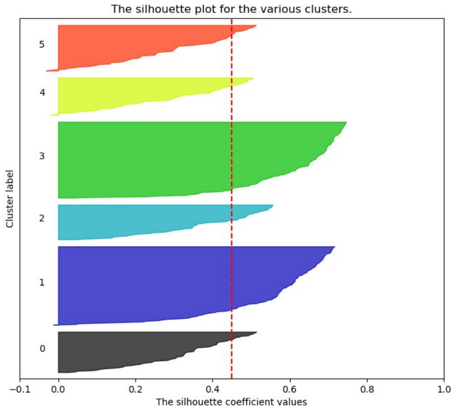
- K-Means is based on the idea of centroids, which are points in the centre of the cluster.
- K centroids are initialized randomly, and the instances are assigned to the cluster whose centroid is closest.
- The algorithm converges when the movement of centroids is below a certain threshold.
- Quite fast, sensitive to outliers

DBSCAN



- DBSCAN considers clusters to be regions of high density.
- For each instance, the algorithm counts the number of instances within a distance ε, also called the instance’s ε-neighbourhood.
- The neighbours of this core instance are considered to be in the same cluster, where some neighbours may also be core instances themselves.
- A cluster consists of a multitude of core instances.

Cluster Optimization: Silhouette score



- The silhouette score has been utilized to provide a metric for accuracy of clusters in this project.
- This score computes the mean silhouette coefficient of all samples.

silhouette_score=b-a/max(a,b)

- Clusters are optimized by exhaustive search this project, i.e., every combination of parameters is tested using the silhouette score, before returning the parameters resulting in the highest score.

Experimental Results

Set Up

1 **Discretize the Data**

- Equal Width
- Equal Depth
- k-Means
- DBSCAN

2 **Learning Arguments**

- Pruned Search
- Decision Trees
- HeRO Algorithm

3 **Make Predictions**

- Test Set
- Training Set

4 **Evaluate**

- Accuracy
- F1 Score

Hyperparamter

- Algorithm Type
- No. Bins (optional)
- Search Depth
- Max Premises

Evaluation Pruned Search

n=198	Acc	F1	No. bins	Depth	Runtime	Max premises
Acc	1,000					
F1	0,940	1,000				
No. bins	-0,008	0,057	1,000			
Depth	0,000	0,000	0,000	1,000		
Runtime	-0,173	-0,001	0,170	0,035	1,000	
Max premises	0,000	0,000			0,207	1,000

- Depth and Premise constraint do not affect precision
- Algorithms with higher accuracy have lower the runtime

Evaluation Decision Trees and HeRO

Decision Trees

- Very high accuracy and F1 score for all discretization techniques
- Reasonable runtime in comparison
- Work as well with undiscretized input data

HeRO

- Performance strongly depends on the discretization technique. Accuracy varies from 0.5 to 0.97
- Very high variations in runtime. Some configurations were 30x times slower
- Still has potential for improvements

Experimental Results

Discussion

Evaluation

- Accuracy does not draw a complete picture because of explainability also matters
- Number of bins has a tremendous impact on the difficulty of the problem (e.g. no. bins=1)

Future Work: Measure explainability of the algorithm

Runtime

- Runtime increases exponentially in search and discretization algorithms
- Experiments were run on reduced data sets
- Needs improvement before becoming applicable

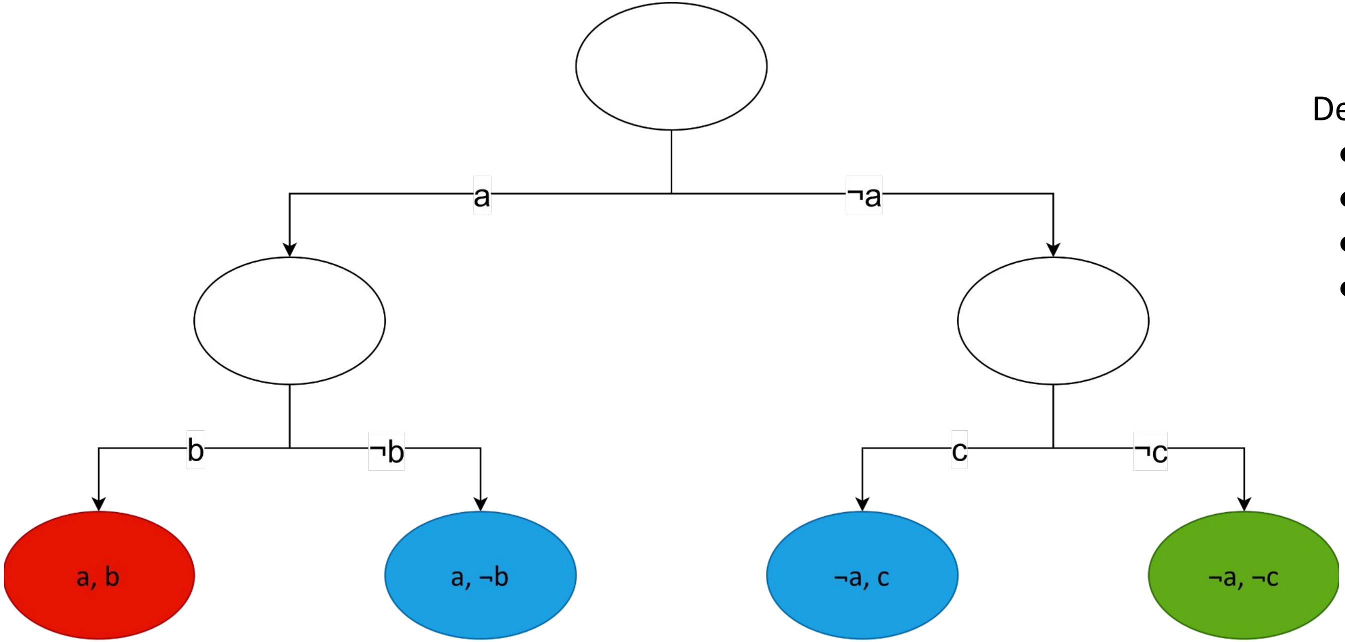
Future Work: Measure explainability of the algorithm

Future Work

Future Work

- Arguments with Exceptions for Decision Tree Rule-Mining
  - Derive arguments with exceptions from decision trees
  - Form arguments with exceptions, which allows prediction on incomplete data
- Optimizing Discretization
  - Choosing the columns to be discretized by selecting columns containing numerical values

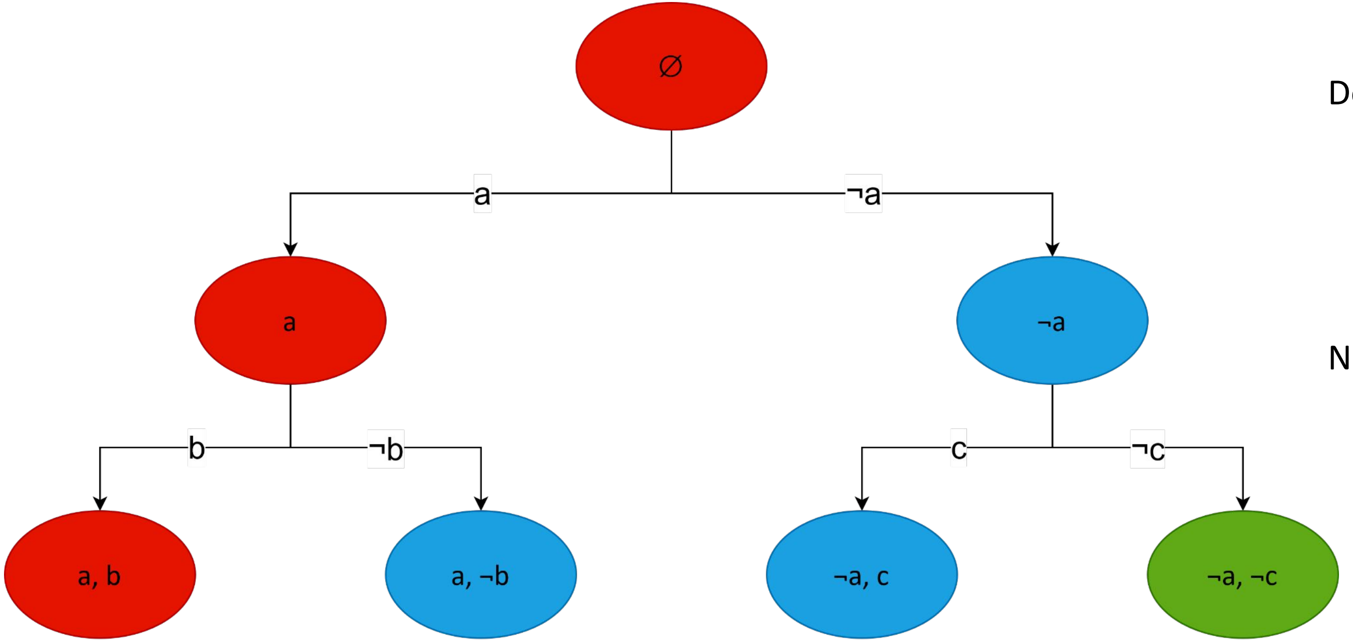
Arguments with exceptions from decision trees



Derived rules:

- a, b → red
- a, ¬b → blue
- ¬a, c → blue
- ¬a, ¬c → green

Arguments with exceptions from decision trees



Derived rules:

- a, b $\longrightarrow$ red
- a, ¬b $\longrightarrow$ blue
- ¬a, c $\longrightarrow$ blue
- ¬a, ¬c $\longrightarrow$ green

New rules:

- ∅ $\longrightarrow$ red
- a $\longrightarrow$ red
- ¬a $\longrightarrow$ blue

Arguments with exceptions from decision trees: Some rules imply others!

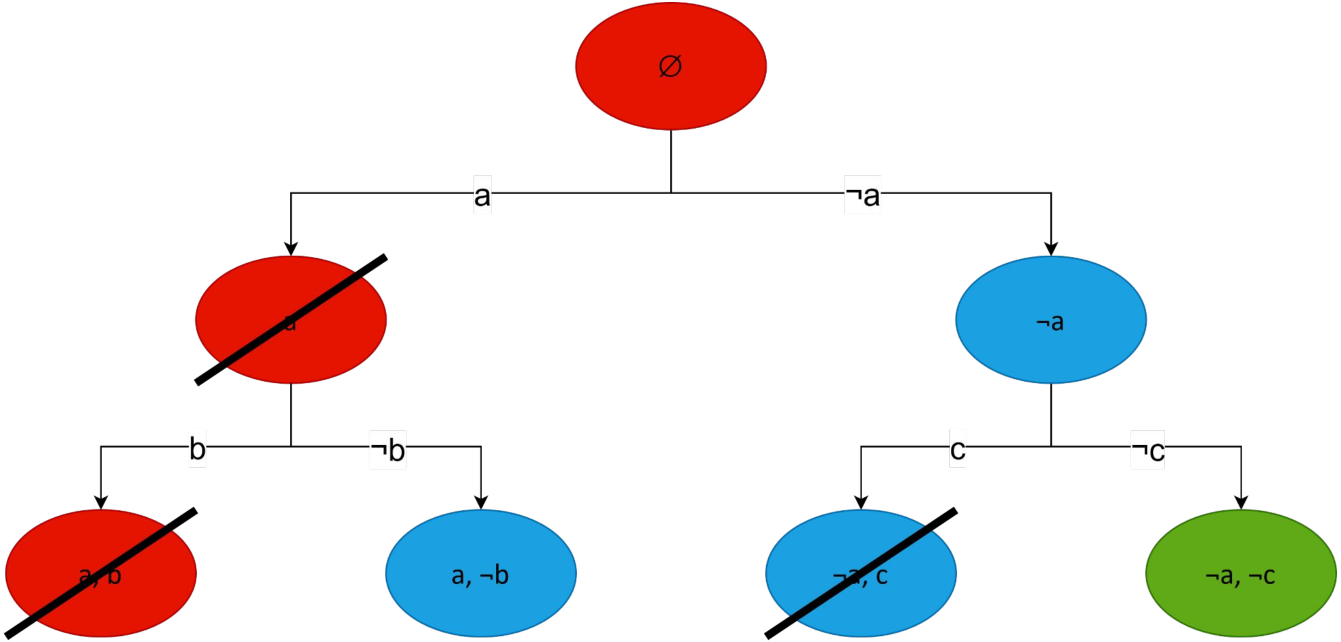
$\varnothing \longrightarrow$ red	**implies**	a, b $\longrightarrow$ red
¬a $\longrightarrow$ blue	**implies**	¬a, c $\longrightarrow$ blue

If rule a implies rule b:

- The premises of rules a are a subset of the premises of rule b
- The conclusions of rule a and b are the same

Since these cases are covered by the less specific rules, we can prune rules that are implied.

Arguments with exceptions from decision trees



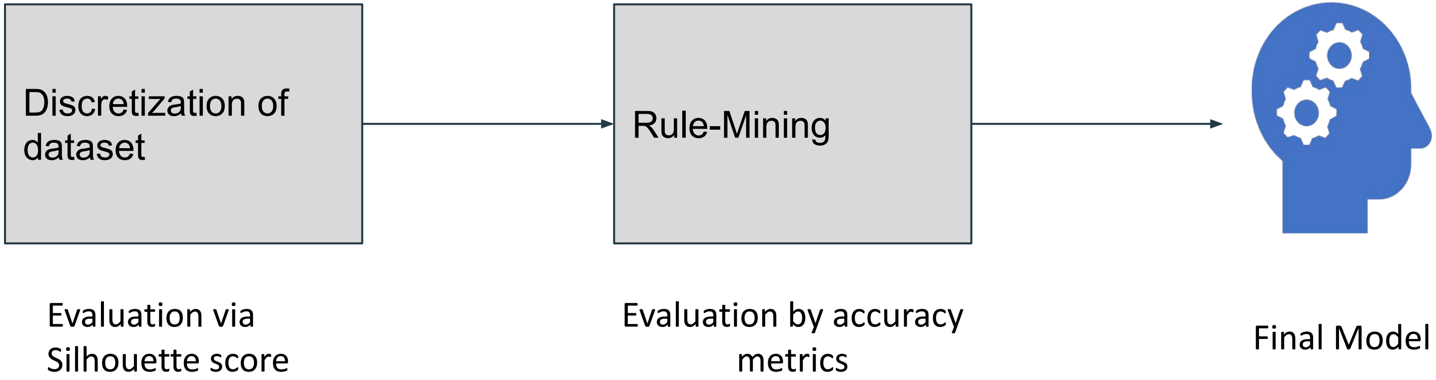
Derived rules:

- ~~a, b → red~~
- a, ¬b → blue
- ~~¬a, c → blue~~
- ¬a, ¬c → green

New rules:

- ∅ → red
- ~~a → red~~
- ¬a → blue

Merging discretization and rule-mining into one classifier



Merging discretization and rule-mining into one classifier

Discretization + Rule-Mining

Optimization of all parameters

- Parameters are combined from both techniques
- Can entail techniques used

Final Model

Possible problem with the merge

- Target feature is not taken into account for discretization
- Instead, discretization will aim to maximize clustering metrics by design
- Improvement may be negligible

Conclusion

Project Conclusion

1. Examples from Verheij are implemented and replicated
2. Four different algorithms for learning arguments have been investigated:
    - Naive Search
    - Pruned Search
    - HerO
    - Decision Tree
3. The approach has been transferred to an attribute-value dataset:
    - Equal-Width Binning
    - Equal-Depth Binning
    - K-Means Clustering
    - DBSCAN Clustering

Project Conclusion

4. We have given a survey over existing techniques for mining rules and arguments. We have applied ideas from logical learning and from the Apriori algorithm. We have implemented the HeRO algorithm. We also sketched an algorithm that allows learning arguments from decision trees

5. Applicability of the algorithms has been shown by experimentation

Thank you!
Questions?

Bart Verheij. ‘Proof with and without Probabilities’. *Artificial Intelligence and Law* 25, no. 1 (2017): 127–54.

Bart Verheij. ‘Arguments for Good Artificial Intelligence’. Inaugural lecture. Groningen: University of Groningen, 2018.

Benjamin Johnston and Guido Governatori. ‘An Algorithm for the Induction of Defeasible Logic Theories from Databases’. In *Proceedings of the 14th Australasian Database Conference-Volume 17*, 75–83, 2003.

Appendix

What specific algorithms are used in Bayesian Optimization?

- To create the prior: Kriging/Gaussian process regression
- For the acquisition function: Probabilistically, one of the following functions are chosen:
  - Lower confidence bound
  - Negative expected improvement
  - Negative probability of improvement
- An unknown numerical approach is used to find the optimum of the prior (not in documentation)

1	inn, $\neg$gui
0	$\neg$inn, gui, evi

(a) Case model

Verheij 2017	*Naive search*	*Pruned search*	*HeRO*
inn $\leftsquigarrow$	inn $\wedge \neg$gui $\leftsquigarrow$	inn $\wedge \neg$gui $\leftsquigarrow$	inn $\wedge \neg$gui $\wedge$evi $\leftsquigarrow$
$\neg$gui $\leftarrow$ inn		$\neg$gui $\leftarrow$ inn	
		gui $\leftarrow \neg$inn	
		evi $\wedge$ gui $\leftarrow \neg$inn	
		evi $\wedge \neg$inn $\leftarrow$ gui	
gui $\leftarrow$ evi		gui $\wedge \neg$inn $\leftarrow$ evi	gui $\wedge \neg$inn $\leftsquigarrow$ evi

(b) Learned arguments

Figure 5.1: Learning arguments in case model 1 from Verheij (2017): *Presumption of innocence.*

2	a	b	c	y
1	a	b	$\neg c$	$\neg y$
0	a	$\neg b$	$\neg c$	y

(a) Case model

Manual	*Naive search*	*Pruned search*	*HeRO*
$y \leftsquigarrow$	$y \leftsquigarrow$	$y \leftsquigarrow$	$y \leftsquigarrow$
		$y \leftarrow c$	
$\neg y \leftsquigarrow \neg c$		$\neg y \leftsquigarrow \neg c$	
		$\neg y \leftarrow b \wedge \neg c$	$\neg y \leftsquigarrow b \wedge \neg c$
		$\neg y \leftsquigarrow a \wedge \neg c$	
$y \leftarrow \neg b$		$y \leftarrow \neg b$	

(b) Learned arguments

Figure 5.2: Learning arguments in case model 1 from Verheij (2017): *Presumption of innocence.*