Naïve Inference viewed as Computation: CogSci 2011 Presentation

Chris Thornton
Informatics
University of Sussex
Brighton
BN1 9QH
UK
c.thornton@sussex.ac.uk

July 12, 2011

General motivation

Probabilities are an intuitive and precise way to represent knowledge.

```
P(rain) = 0.2
P(mud|rain) = 0.5
```

Laws of probability yield inferences.

```
P(rain) x P(mud|rain) --> P(mud) = 0.1
```

But with non-trivial representations, inferential threads tend to interact.

```
If we add P(mud|leak) = 0.8
what is P(mud) ?
```

Classical probabilistic reasoning

Classical probabilistic reasoning

- Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo: Morgan and Kaufman.

Classical probabilistic reasoning

- Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo: Morgan and Kaufman.

Brings the problem under control through ‘divide and conquer’.

Classical probabilistic reasoning

- Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo: Morgan and Kaufman.

Brings the problem under control through 'divide and conquer'.

- View probabilities as constraints applying to an underlying, static, joint distribution on all variables.

Classical probabilistic reasoning

- Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo: Morgan and Kaufman.

Brings the problem under control through 'divide and conquer'.

- View probabilities as constraints applying to an underlying, static, joint distribution on all variables.
- Infer unknown values in the joint distribution from known values, by deploying inference rules in a way that takes advantage of conditional independencies (i.e., factorization).

Classical probabilistic reasoning

- Pearl, J. (1988). *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo: Morgan and Kaufman.

Brings the problem under control through 'divide and conquer'.

- View probabilities as constraints applying to an underlying, static, joint distribution on all variables.
- Infer unknown values in the joint distribution from known values, by deploying inference rules in a way that takes advantage of conditional independencies (i.e., factorization).

Bayesian networks force the user to state probability information in a way that guarantees this can be done for all values.

Drawbacks

Drawbacks

- The approach cannot be used if there *is no* static joint distribution.

Drawbacks

- The approach cannot be used if there *is no* static joint distribution.
- This is more or less inevitable with domains which have dynamic properties.

Drawbacks

- The approach cannot be used if there *is no* static joint distribution.
- This is more or less inevitable with domains which have dynamic properties.
- It is guaranteed if there are any conditional cycles.

Drawbacks

- The approach cannot be used if there *is no* static joint distribution.
- This is more or less inevitable with domains which have dynamic properties.
- It is guaranteed if there are any conditional cycles.

Bayesian networks make conditional cycles 'illegal'.

Naïve inferential reasoning

Data-driven alternative that doesn't require there to be a static joint distribution.

Naïve inferential reasoning

Data-driven alternative that doesn't require there to be a static joint distribution.

- View probabilities as directly mandating acts of inference.

Naïve inferential reasoning

Data-driven alternative that doesn't require there to be a static joint distribution.

- View probabilities as directly mandating acts of inference.
- Inferred values assumed to trigger new inferences in a data-driven way.

Naïve inferential reasoning

Data-driven alternative that doesn't require there to be a static joint distribution.

- View probabilities as directly mandating acts of inference.
- Inferred values assumed to trigger new inferences in a data-driven way.
- Normalization applied at each stage (as usual).

Naïve inferential reasoning

Data-driven alternative that doesn't require there to be a static joint distribution.

- View probabilities as directly mandating acts of inference.
- Inferred values assumed to trigger new inferences in a data-driven way.
- Normalization applied at each stage (as usual).
- Process continues until no new values produced.

Naïve inferential reasoning

Data-driven alternative that doesn't require there to be a static joint distribution.

- View probabilities as directly mandating acts of inference.
- Inferred values assumed to trigger new inferences in a data-driven way.
- Normalization applied at each stage (as usual).
- Process continues until no new values produced.

Inference becomes the behaviour of a parallel, data-driven 'naïve machine'.

Naïve inferential reasoning

Data-driven alternative that doesn't require there to be a static joint distribution.

- View probabilities as directly mandating acts of inference.
- Inferred values assumed to trigger new inferences in a data-driven way.
- Normalization applied at each stage (as usual).
- Process continues until no new values produced.

Inference becomes the behaviour of a parallel, data-driven 'naïve machine'.

- *Bad news*: this is much less like logical inference.

Naïve inferential reasoning

Data-driven alternative that doesn't require there to be a static joint distribution.

- View probabilities as directly mandating acts of inference.
- Inferred values assumed to trigger new inferences in a data-driven way.
- Normalization applied at each stage (as usual).
- Process continues until no new values produced.

Inference becomes the behaviour of a parallel, data-driven 'naïve machine'.

- *Bad news*: this is much less like logical inference.
- *Good news*: it's much more like computation.

Naïve inferential reasoning

Data-driven alternative that doesn't require there to be a static joint distribution.

- View probabilities as directly mandating acts of inference.
- Inferred values assumed to trigger new inferences in a data-driven way.
- Normalization applied at each stage (as usual).
- Process continues until no new values produced.

Inference becomes the behaviour of a parallel, data-driven 'naïve machine'.

- *Bad news*: this is much less like logical inference.
- *Good news*: it's much more like computation.

Formalities

A *probabilistic model* is here a set of (conditional and unconditional) probability values for some set of random discrete variables.

The unnormalized inferred probability for X_i in model M:

$$P_M(X_i) = \sum_{c \in C(X_i, M)} P_M(X_i|c) P_M(c)$$

Given $C(x, M)$ as the set of conditions figuring in conditional probabilities asserted for values of X in model M, the distribution inferred for X is then

$$\mathbf{P}'_M(X) = \alpha \langle P_M(X_1), ..., P_M(X_n) \rangle$$

Model revision

The naïve-inferential revision M' of model M is the complete set of inferences obtained.

$$M' = \{\ \mathbf{P}'_M(X)\ \mid\ X \in M \wedge P'_M(X) \neq P_M(X)\ \}$$

Naïve machines

Recursive evaluation of M' yields a sequence of revisions:

$$M_i = M'_{i-1} \Longrightarrow M_{i-1}$$

where the '$\Longrightarrow$' operator denotes model imposition.
Sequence of revisions for model M is then the behaviour of the naïve machine $N(M)$ defined by M.

$$N(M) = (\ M'_0,\ \ldots,\ M'_n\ )$$

Oscillator example

$M =$

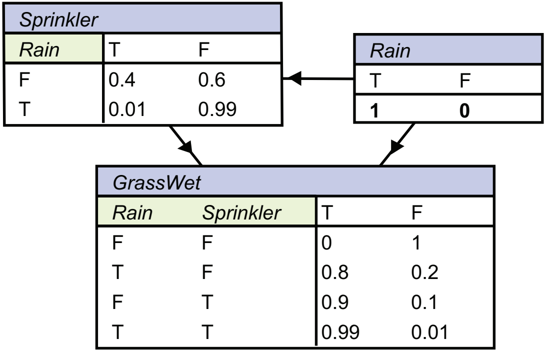
$$\begin{array}{l|l} P(X=1)=1 & P(X=0)=0 \\ P(X=1|X=1)=0 & P(X=0|X=1)=1 \\ P(X=1|X=0)=1 & P(X=0|X=0)=0 \end{array}$$

Naïve-machine behaviour:

$$N(M) = \begin{pmatrix} \{P(X)=1\}, \\ \{P(X)=0\}, \\ \{P(X)=1\}, \\ \{P(X)=0\}, \\ \dots \end{pmatrix}$$

Emulation of ideal Bayesian inference

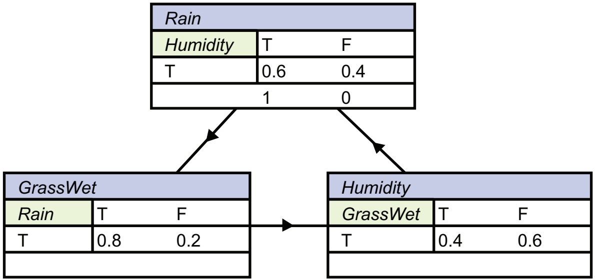
Naïve machines emulate Bayesian networks in some simple cases.



Discovery that $P(Rain) = 1.0$ yields $P(Sprinker) = 0.01$ and $P(GrassWet) = 0.802$.

Top-down revision of the Bayesian network reproduced by the naïve machine.

Bring in a conditional cycle



Now we get the following (finite) sequence:

$$N(M) = \left(\begin{array}{l} \{P(Rain) = 1\}, \\ \{P(GrassWet) = 0.8\}, \\ \{P(Humidity) = 0.4\}, \\ \{P(Rain) = 0.6\}, \\ \{P(GrassWet) = 0.8\}, \end{array} \right)$$

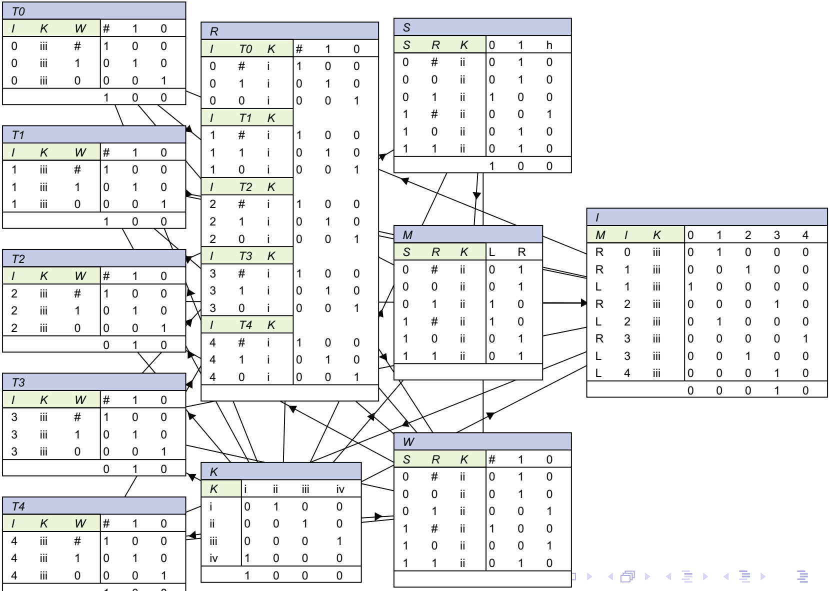
Emulation of Turing machines

For any Turing machine, we can build a naïve machine emulation.

State	Read	Write	Move	New state
0	#	1	R	1
0	0	1	R	1
0	1	0	L	0
1	#	#	L	h
1	0	0	R	1
1	1	1	R	1

Table: Incrementing Turing Machine

Incrementing naïve machine



Conclusions

Conclusions

- Naïve inference is a plausible alternative for use in dynamic domains, or any situation where we do not have a static joint distribution.

Conclusions

- Naïve inference is a plausible alternative for use in dynamic domains, or any situation where we do not have a static joint distribution.

Link from probabilistic inference to computation may have implications for

Conclusions

- Naïve inference is a plausible alternative for use in dynamic domains, or any situation where we do not have a static joint distribution.

Link from probabilistic inference to computation may have implications for

- notions of 'bounded rationality'

Conclusions

- Naïve inference is a plausible alternative for use in dynamic domains, or any situation where we do not have a static joint distribution.

Link from probabilistic inference to computation may have implications for

- notions of ‘bounded rationality’
- intractability of ideal Bayesian inference

Conclusions

- Naïve inference is a plausible alternative for use in dynamic domains, or any situation where we do not have a static joint distribution.

Link from probabilistic inference to computation may have implications for

- notions of 'bounded rationality'
- intractability of ideal Bayesian inference
- application of probabilistic methods to structured representation

Conclusions

- Naïve inference is a plausible alternative for use in dynamic domains, or any situation where we do not have a static joint distribution.

Link from probabilistic inference to computation may have implications for

- notions of 'bounded rationality'
- intractability of ideal Bayesian inference
- application of probabilistic methods to structured representation
- ways of connecting probability with logic

Conclusions

- Naïve inference is a plausible alternative for use in dynamic domains, or any situation where we do not have a static joint distribution.

Link from probabilistic inference to computation may have implications for

- notions of ‘bounded rationality’
- intractability of ideal Bayesian inference
- application of probabilistic methods to structured representation
- ways of connecting probability with logic

Any questions?

Visit www.christhornton.eu for more on this.