Probabilistic reasoning in type systems

Silvia Ghilezan[1], Jelena Ivetić[1], Zoran Ognjanović[2], **Nenad Savić**[1]

1 : Faculty of Technical Sciences, University of Novi Sad
2 : Mathematical Institute of SASA

LAP 2016.

Outline of the talk

Outline of the talk

- The Idea

Outline of the talk

- The Idea
- Syntax

Outline of the talk

- The Idea
- Syntax
- Semantics

Outline of the talk

- The Idea
- Syntax
- Semantics
- Axiomatization

Outline of the talk

- The Idea
- Syntax
- Semantics
- Axiomatization
- Construction of the canonical model

Outline of the talk

- The Idea
- Syntax
- Semantics
- Axiomatization
- Construction of the canonical model
- Strong completeness

The Idea

Probabilistic Logic (Extension of the classical propositional logic)

The Idea

Probabilistic Logic (Extension of the classical propositional logic)

$$(P_{\leq \frac{1}{3}} p \wedge P_{\leq \frac{1}{4}} q) \Rightarrow (P_{\leq \frac{1}{4}} (p \wedge q))$$

The Idea

Probabilistic Logic (Extension of the classical propositional logic)

$$(P_{\leq \frac{1}{3}} p \wedge P_{\leq \frac{1}{4}} q) \Rightarrow (P_{\leq \frac{1}{4}} (p \wedge q))$$

- $p \wedge P_{\geq \frac{1}{2}} q$

The Idea

Probabilistic Logic (Extension of the classical propositional logic)

$$(P_{\leq \frac{1}{3}} p \wedge P_{\leq \frac{1}{4}} q) \Rightarrow (P_{\leq \frac{1}{4}} (p \wedge q))$$

- $p \wedge P_{\geq \frac{1}{2}} q$
- $P_{\geq \frac{1}{3}} P_{\geq \frac{1}{2}} p$

The Idea

Probabilistic Logic (Extension of the classical propositional logic)

$$(P_{\leq \frac{1}{3}} p \wedge P_{\leq \frac{1}{4}} q) \Rightarrow (P_{\leq \frac{1}{4}} (p \wedge q))$$

- $p \wedge P_{\geq \frac{1}{2}} q$
- $P_{\geq \frac{1}{3}} P_{\geq \frac{1}{2}} p$

Lambda Calculus with Intersection Types

The Idea

Probabilistic Logic (Extension of the classical propositional logic)

$$(P_{\leq \frac{1}{3}} p \wedge P_{\leq \frac{1}{4}} q) \Rightarrow (P_{\leq \frac{1}{4}} (p \wedge q))$$

- $p \wedge P_{\geq \frac{1}{2}} q$
- $P_{\geq \frac{1}{3}} P_{\geq \frac{1}{2}} p$

Lambda Calculus with Intersection Types

$$\dfrac{\dfrac{x : \sigma \qquad x : \tau}{x : \sigma \cap \tau}\,(\cap)}{\lambda x.x : (\sigma \cap \tau) \to (\sigma \cap \tau)}\,(\to_I)$$

The Idea

Probabilistic Logic (Extension of the classical propositional logic)

$$(P_{\leq \frac{1}{3}} p \wedge P_{\leq \frac{1}{4}} q) \Rightarrow (P_{\leq \frac{1}{4}} (p \wedge q))$$

- $p \wedge P_{\geq \frac{1}{2}} q$
- $P_{\geq \frac{1}{3}} P_{\geq \frac{1}{2}} p$

Lambda Calculus with Intersection Types

$$\dfrac{\dfrac{x : \sigma \qquad x : \tau}{x : \sigma \cap \tau}\,(\cap)}{\lambda x.x : (\sigma \cap \tau) \to (\sigma \cap \tau)}\,(\to_I)$$

Completeness theorem for the Lambda Calculus with Intersection Types

Completeness theorem for the Lambda Calculus with Intersection Types

With the notion of filter lambda model, completeness of the type assignment was proved:

Theorem (Completeness)

$\Gamma \vdash M : \sigma \Leftrightarrow \Gamma \models M : \sigma.$

Syntax of $P\Lambda^{\cap}$

Syntax of $\mathsf{P}\Lambda^{\cap}$

Let $\mathsf{S} = [0,1] \cap \mathbb{Q}$. The *alphabet* of the logic $\mathsf{P}\Lambda^{\cap}$ consists of

- all symbols needed to define lambda terms with intersection types,
- the classical propositional connectives $\neg$ and $\wedge$,
- the list of probability operators $P_{\geq s}$, for every $s \in \mathsf{S}$.

Syntax of $P\Lambda^{\cap}$

Let $S = [0,1] \cap \mathbb{Q}$. The *alphabet* of the logic $P\Lambda^{\cap}$ consists of

- all symbols needed to define lambda terms with intersection types,
- the classical propositional connectives $\neg$ and $\wedge$,
- the list of probability operators $P_{\geq s}$, for every $s \in S$.

Remark: Using $P_{\geq s}\alpha$ we can define other inequalities:

$$\begin{array}{lll} P_{<s}\alpha & \text{stands for} & \neg P_{\geq s}\alpha, \\ P_{\leq s}\alpha & \text{stands for} & P_{\geq 1-s}\neg\alpha, \\ P_{>s}\alpha & \text{stands for} & \neg P_{\leq s}\alpha, \\ P_{=s}\alpha & \text{stands for} & P_{\geq s}\alpha \wedge \neg P_{>s}\alpha. \end{array}$$

Basic and Probabilistic Formulas

Basic and Probabilistic Formulas

Basic formulas:

$$\mathsf{For_B} \quad \alpha ::= M : \sigma \mid \alpha \wedge \alpha \mid \neg\alpha.$$

Basic and Probabilistic Formulas

Basic formulas:

$$\mathsf{For}_{\mathsf{B}} \quad \alpha ::= M : \sigma \mid \alpha \wedge \alpha \mid \neg\alpha.$$

Example:

- $x : \sigma$

Basic and Probabilistic Formulas

Basic formulas:

$$\mathsf{For}_\mathsf{B} \quad \alpha ::= M : \sigma \mid \alpha \wedge \alpha \mid \neg\alpha.$$

Example:

- $x : \sigma$, $\qquad \lambda x.xy : \sigma \to \tau$

Basic and Probabilistic Formulas

Basic formulas:

$$\mathrm{For_B} \quad \alpha ::= M : \sigma \mid \alpha \wedge \alpha \mid \neg\alpha.$$

Example:

- $x : \sigma$, $\qquad \lambda x.xy : \sigma \to \tau$
- $(x : \sigma) \wedge (y : \sigma \cap \tau)$

Basic and Probabilistic Formulas

Basic formulas:

$$\mathrm{For_B} \quad \alpha ::= M : \sigma \mid \alpha \wedge \alpha \mid \neg\alpha.$$

Example:

- $x : \sigma,$ $\quad \lambda x.xy : \sigma \to \tau$
- $(x : \sigma) \wedge (y : \sigma \cap \tau),$ $\quad (x : \sigma \wedge y : \sigma \to \tau) \Rightarrow yx : \tau$

Basic and Probabilistic Formulas

Basic formulas:

$$\mathrm{For_B} \quad \alpha ::= M : \sigma \mid \alpha \wedge \alpha \mid \neg\alpha.$$

Example:

- $x : \sigma,$ $\qquad \lambda x.xy : \sigma \to \tau$
- $(x : \sigma) \wedge (y : \sigma \cap \tau),$ $\qquad (x : \sigma \wedge y : \sigma \to \tau) \Rightarrow yx : \tau$

Probabilistic formulas:

$$\mathrm{For_P} \quad \phi ::= P_{\geq s}\alpha \mid \phi \wedge \phi \mid \neg\phi.$$

Basic and Probabilistic Formulas

Basic formulas:

$$\mathrm{For_B} \quad \alpha ::= M : \sigma \mid \alpha \wedge \alpha \mid \neg\alpha.$$

Example:

- $x : \sigma, \qquad \lambda x.xy : \sigma \to \tau$
- $(x : \sigma) \wedge (y : \sigma \cap \tau), \qquad (x : \sigma \wedge y : \sigma \to \tau) \Rightarrow yx : \tau$

Probabilistic formulas:

$$\mathrm{For_P} \quad \phi ::= P_{\geq s}\alpha \mid \phi \wedge \phi \mid \neg\phi.$$

Example:

- $P_{=\frac{1}{3}} x : \sigma$

Basic and Probabilistic Formulas

Basic formulas:

$$\mathrm{For}_{\mathrm{B}} \quad \alpha ::= M : \sigma \mid \alpha \wedge \alpha \mid \neg\alpha.$$

Example:

- $x : \sigma,$ $\qquad \lambda x.xy : \sigma \to \tau$
- $(x : \sigma) \wedge (y : \sigma \cap \tau),$ $\qquad (x : \sigma \wedge y : \sigma \to \tau) \Rightarrow yx : \tau$

Probabilistic formulas:

$$\mathrm{For}_{\mathrm{P}} \quad \phi ::= P_{\geq s}\alpha \mid \phi \wedge \phi \mid \neg\phi.$$

Example:

- $P_{=\frac{1}{3}} x : \sigma,$ $\qquad P_{\geq \frac{1}{4}}(\lambda x.xy : \sigma \to \tau)$

Basic and Probabilistic Formulas

Basic formulas:

$$\mathsf{For_B} \quad \alpha ::= M : \sigma \mid \alpha \wedge \alpha \mid \neg\alpha.$$

Example:

- $x : \sigma$, $\quad \lambda x.xy : \sigma \to \tau$
- $(x : \sigma) \wedge (y : \sigma \cap \tau)$, $\quad (x : \sigma \wedge y : \sigma \to \tau) \Rightarrow yx : \tau$

Probabilistic formulas:

$$\mathsf{For_P} \quad \phi ::= P_{\geq s}\alpha \mid \phi \wedge \phi \mid \neg\phi.$$

Example:

- $P_{=\frac{1}{3}} x : \sigma$, $\quad P_{\geq\frac{1}{4}}(\lambda x.xy : \sigma \to \tau)$
- $P_{\leq 0.2}(x : \sigma) \vee P_{\geq 0.8}(y : \sigma \cap \tau)$

Basic and Probabilistic Formulas

Basic formulas:

$$\mathrm{For}_{\mathrm{B}} \quad \alpha ::= M : \sigma \mid \alpha \wedge \alpha \mid \neg\alpha.$$

Example:

- $x : \sigma,\qquad \lambda x.xy : \sigma \to \tau$
- $(x : \sigma) \wedge (y : \sigma \cap \tau),\qquad (x : \sigma \wedge y : \sigma \to \tau) \Rightarrow yx : \tau$

Probabilistic formulas:

$$\mathrm{For}_{\mathrm{P}} \quad \phi ::= P_{\geq s}\alpha \mid \phi \wedge \phi \mid \neg\phi.$$

Example:

- $P_{=\frac{1}{3}} x : \sigma,\qquad P_{\geq\frac{1}{4}}(\lambda x.xy : \sigma \to \tau)$
- $P_{\leq 0.2}(x : \sigma) \vee P_{\geq 0.8}(y : \sigma \cap \tau),\qquad P_{=1}(x : \sigma \wedge y : \sigma \to \tau) \Rightarrow P_{=1}(yx : \tau)$

No mixing and no iterations

No mixing and no iterations

The next two formulas are **NOT** the **FORMULAS** of our logic:

No mixing and no iterations

The next two formulas are **NOT** the **FORMULAS** of our logic:

- $(x : \sigma) \wedge P_{\geq \frac{1}{2}}(y : \tau_1 \cap \tau_2)$

No mixing and no iterations

The next two formulas are **NOT** the **FORMULAS** of our logic:

- $(x : \sigma) \wedge P_{\geq \frac{1}{2}}(y : \tau_1 \cap \tau_2)$
- $P_{\geq \frac{1}{3}} P_{\geq \frac{1}{2}}(xy : \sigma)$

Kripke-style Semantics of $P\Lambda^{\cap}$

Kripke-style Semantics of $P\Lambda^{\cap}$

Definition ($P\Lambda^{\cap}$-structure)

A $P\Lambda^{\cap}$-structure is a tuple $\mathcal{M} = \langle W, \rho, \xi, H, \mu \rangle$, where:

(i) W is a nonempty set of *worlds*, where each world is one lambda model, i.e. for every $w \in W$, $w = \langle \mathcal{L}(w), \cdot_w, [\![\ ]\!]_w \rangle$;

(ii) $\rho : \mathtt{V}_\Lambda \times \{w\} \longrightarrow \mathcal{L}(w)$, $w \in W$;

(iii) $\xi : \mathtt{V}_{\mathtt{Type}} \times \{w\} \longrightarrow \mathcal{P}(\mathcal{L}(w))$, $w \in W$;

(iv) H is an *algebra of subsets* of W, i.e. $H \subseteq \mathcal{P}(W)$ such that
- $W \in H$,
- if $U, V \in H$, then $W \setminus U \in H$ and $U \cup V \in H$;

(v) μ is a *finitely additive probability measure* defined on H, i.e.
- $\mu(W) = 1$,
- if $U \cap V = \emptyset$, then $\mu(U \cup V) = \mu(U) + \mu(V)$, for all $U, V \in H$.

Satisfiability of a formula

Satisfiability of a formula

We say that a lambda statement $M : \sigma$ holds in a world w, denoted by $w \models M : \sigma$, iff

$$[\![M]\!]_{\rho}^{w} \in [\![\sigma]\!]_{\xi}^{w}.$$

Satisfiability of a formula

We say that a lambda statement $M : \sigma$ holds in a world w, denoted by $w \models M : \sigma$, iff

$$[\![M]\!]^w_\rho \in [\![\sigma]\!]^w_\xi .$$

Definition (Satisfiability relation)

The satisfiability relation $\models \subseteq P\Lambda^{\cap}_{\mathrm{Meas}} \times \mathrm{For}_{P\Lambda^{\cap}}$ is defined in the following way:

- $\mathcal{M} \models M : \sigma$ iff $w \models M : \sigma$, for all $w \in W$;
- $\mathcal{M} \models P_{\geq s}\alpha$ iff $\mu([\alpha]) \geq s$;
- $\mathcal{M} \models \neg A$ iff it is not the case that $\mathcal{M} \models A$;
- $\mathcal{M} \models A_1 \wedge A_2$ iff $\mathcal{M} \models A_1$ and $\mathcal{M} \models A_2$.

Example

Example

Example

Consider following two models with three worlds, i.e., let $\mathcal{M}_i = \langle W_i, \rho_i, \xi_i, H_i, \mu_i \rangle$, $i = 1, 2$, where:

- $W_i = \{w_1, w_2, w_3\}$, $i = 1, 2$,
- $H_i = \mathcal{P}(W_i)$, $i = 1, 2$,
- $\mu_i(\{w_j\}) = \frac{1}{3}$, $j = 1, 2, 3$, $i = 1, 2$,

and ρ_i and ξ_i are defined such that $\mathcal{M}_i \models P_{=\frac{1}{3}}(x : \sigma \to \tau)$ and $\mathcal{M}_i \models P_{=\frac{2}{3}}(y : \sigma)$, but in the model $\mathcal{M}_1$: $w_1 \models x : \sigma \to \tau$, $w_2 \models y : \sigma$ and $w_3 \models y : \sigma$ (Figure 1), while in the model $\mathcal{M}_2$: $w_1 \models x : \sigma \to \tau$, $w_1 \models y : \sigma$ and $w_2 \models y : \sigma$ (Figure 2).
Considering the first model we obtain that $\mathcal{M}_1 \models P_{=0}(xy : \tau)$, but considering the second model we obtain that $\mathcal{M}_2 \models P_{=\frac{1}{3}}(xy : \tau)$.

$y : \sigma$ holds in w_2 and w_3:

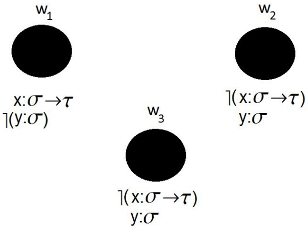

Figure:

$y : \sigma$ holds in w_1 and w_2:

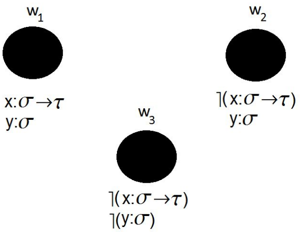

Figure:

Axiom schemes

(1) all instances of the classical propositional tautologies, (atoms are λ-statements or any $\mathsf{P}\Lambda^{\cap}$-formulas),

(2) $P_{\geq 0}\alpha$,

(3) $P_{\leq r}\alpha \Rightarrow P_{<s}\alpha$, $s > r$,

(4) $P_{<s}\alpha \Rightarrow P_{\leq s}\alpha$,

(5) $(P_{\geq r}\alpha \wedge P_{\geq s}\beta \wedge P_{\geq 1}(\neg\alpha \vee \neg\beta)) \Rightarrow P_{\geq min\{1,r+s\}}(\alpha \vee \beta)$,

(6) $(P_{\leq r}\alpha \wedge P_{<s}\beta) \Rightarrow P_{<r+s}(\alpha \vee \beta)$, $r + s \leq 1$,

(7) $P_{\geq 1}(\alpha \Rightarrow \beta) \Rightarrow (P_{\geq s}\alpha \Rightarrow P_{\geq s}\beta)$.

Inference Rules I

$$\frac{M:\sigma\to\tau \qquad N:\sigma}{MN:\tau}\ (\to_E)$$

$$\frac{\begin{array}{c}[x:\sigma]\\ \vdots\\ M:\tau\end{array}}{\lambda x.M:\sigma\to\tau}\ (\to_I)$$

$$\frac{M:\sigma\cap\tau}{M:\sigma}\ (\cap_E)$$

$$\frac{M:\sigma\cap\tau}{M:\tau}\ (\cap_E)$$

$$\frac{M:\sigma \qquad M:\tau}{M:\sigma\cap\tau}\ (\cap_I)$$

$$\frac{}{M:\omega}\ (\omega)$$

$$\frac{M:\sigma \qquad \sigma\leq\tau}{M:\tau}\ (\leq)$$

Inference Rules II

(1) From A_1 and $A_1 \Rightarrow A_2$ infer A_2,

(2) from α infer $P_{\geq 1}\alpha$,

(3) from the set of premises

$$\{\phi \Rightarrow P_{\geq s-\frac{1}{k}}\alpha \mid k \geq \frac{1}{s}\}$$

infer $\phi \Rightarrow P_{\geq s}\alpha$.

Soundness and Strong Completeness

Soundness and Strong Completeness

Theorem (Soundness)

The axiomatic system $Ax_{\mathsf{P}\Lambda^{\cap}}$ is sound with respect to the class of $\mathsf{P}\Lambda^{\cap}_{\mathsf{Meas}}$-models.

Soundness and Strong Completeness

Theorem (Soundness)

The axiomatic system $Ax_{P\Lambda^{\cap}}$ is sound with respect to the class of $P\Lambda^{\cap}_{\mathrm{Meas}}$-models.

Strong Completeness

Soundness and Strong Completeness

Theorem (Soundness)

The axiomatic system $Ax_{\mathrm{P}\Lambda^{\cap}}$ *is sound with respect to the class of* $\mathrm{P}\Lambda^{\cap}_{\mathrm{Meas}}$*-models.*

Strong Completeness

We need a few auxiliary lemmas in order to prove the strong completeness theorem:

Soundness and Strong Completeness

Theorem (Soundness)

The axiomatic system $Ax_{P\Lambda^{\cap}}$ is sound with respect to the class of $P\Lambda^{\cap}_{\mathrm{Meas}}$-models.

Strong Completeness

We need a few auxiliary lemmas in order to prove the strong completeness theorem:

Theorem

Every consistent set can be extended to a maximal consistent set.

Construction of the canonical model

Construction of the canonical model

Definition

If $T^\star$ is the maximally consistent set of formulas, then a tuple $\mathcal{M}_{T^\star} = \langle W, \rho, \xi, H, \mu \rangle$ is defined:

- $W = \{w = \langle \mathcal{F}(w), \cdot_w, [\![\ ]\!]_w \rangle \mid w \models Cn_B(T)\}$ contains all **filter** lambda models that satisfy the set $Cn_B(T)$,
- $\rho_w(x) = \{\sigma \in Type \mid w \models x : \sigma\}$,
- $\xi_w(\sigma) = \{d \in \mathcal{F}(w) \mid \sigma \in d\}$,
- $H = \{[\alpha] \mid \alpha \in \mathrm{For_B}\}$, where $[\alpha] = \{w \in W \mid w \models \alpha\}$,
- $\mu([\alpha]) = \sup\{s \mid P_{\geq s}\alpha \in T^\star\}$.

Construction of the canonical model

Definition

If $T^\star$ is the maximally consistent set of formulas, then a tuple $\mathcal{M}_{T^\star} = \langle W, \rho, \xi, H, \mu \rangle$ is defined:

- $W = \{w = \langle \mathcal{F}(w), \cdot_w, [\![\ ]\!]_w \rangle \mid w \models Cn_B(T)\}$ contains all **filter** lambda models that satisfy the set $Cn_B(T)$,
- $\rho_w(x) = \{\sigma \in \mathit{Type} \mid w \models x : \sigma\}$,
- $\xi_w(\sigma) = \{d \in \mathcal{F}(w) \mid \sigma \in d\}$,
- $H = \{[\alpha] \mid \alpha \in \mathrm{For_B}\}$, where $[\alpha] = \{w \in W \mid w \models \alpha\}$,
- $\mu([\alpha]) = \sup\{s \mid P_{\geq s}\alpha \in T^\star\}$.

Theorem (Strong completeness)

Every consistent set of formulas T is $\mathrm{P}\Lambda^{\cap}_{\mathrm{Meas}}$-satisfiable.

Further Work

- Intuitionistic instead of classical propositional calculus

Further Work

- Intuitionistic instead of classical propositional calculus
- Restriction to the finite case

References

H. Barendregt, M. Coppo, M. Dezani-Ciancaglini.
A filter lambda model and the completeness of type assignment.
The Journal of Symbolic Logic, 48: 931–940, 1983.

Z. Ognjanović, M. Rašković, Z. Marković.
Probability logics.
Zbornik radova, subseries Logic in computer science, 12 (20), 35–111, Matematički institut, 2009.

S. Ghilezan, S. Likavec.
Computational interpretations of logics.
Zbornik radova, special issue Logic in Computer Science 12(20) 159–215, Matematički institut, 2009.