A simple method of proving logical constancy by consequence extraction

Tin Perkov

LAP 2016, Dubrovnik

Logical constants

A fundamental question: which symbols (or words) of a formal language are logical?

Logical constants

A fundamental question: which symbols (or words) of a formal language are logical?

Some approaches to answer this:

- grammatical (atomic sentences are non-logical, while complex sentences are built using logical constants)

Logical constants

A fundamental question: which symbols (or words) of a formal language are logical?
Some approaches to answer this:

- grammatical (atomic sentences are non-logical, while complex sentences are built using logical constants)
- proof-theoretical (applied to any reasoning, regardless of its subject: definability by inferential rules)

Logical constants

A fundamental question: which symbols (or words) of a formal language are logical?
Some approaches to answer this:

- grammatical (atomic sentences are non-logical, while complex sentences are built using logical constants)
- proof-theoretical (applied to any reasoning, regardless of its subject: definability by inferential rules)
- semantical (fixed meaning, not depending on properties of individuals: invariance under permutations, isomorphisms etc.)

Logical constants and logical consequence

We focus on a recent development (Bonnay-Westerståhl 2012) in the semantical approach, which explores the close relation between logical constants and logical consequence ($S \Rightarrow F$ iff there is no interpretation of non-logical symbols such that S is true and F is false).

Logical constants and logical consequence

We focus on a recent development (Bonnay-Westerståhl 2012) in the semantical approach, which explores the close relation between logical constants and logical consequence ($S \Rightarrow F$ iff there is no interpretation of non-logical symbols such that S is true and F is false). Goals:

- ambitious: find the proper notion of logical constants – probably no answer
- less ambitious: understand how a *choice* of constants generates a consequence relation, and vice versa.

Abstract definition of consequence relation

To reason about fundamental question of logical constants, we need an abstract definition of logical consequence.

Abstract definition of consequence relation

To reason about fundamental question of logical constants, we need an abstract definition of logical consequence.

Definition

A *language* is a triple $L = (Symb, Sent, Tr)$, where:

- $Symb$ is a countable set of *symbols*, partitioned into categories
- $Sent$ is a set of *sentences* over an alphabet containing $Symb$
- $Tr \subseteq Sent$ is the set of *true sentences*.

Abstract definition of consequence relation

To reason about fundamental question of logical constants, we need an abstract definition of logical consequence.

Definition

A *language* is a triple $L = (Symb, Sent, Tr)$, where:

- $Symb$ is a countable set of *symbols*, partitioned into categories
- $Sent$ is a set of *sentences* over an alphabet containing $Symb$
- $Tr \subseteq Sent$ is the set of *true sentences*.

A *consequence relation* is $\Rightarrow \subseteq \mathcal{P}(Sent) \times Sent$ s.t.

- if $\varphi \in \Gamma$, then $\Gamma \Rightarrow \varphi$
- if $\Delta \Rightarrow \varphi$ and $\Gamma \Rightarrow \psi$ for all $\psi \in \Delta$, then $\Gamma \Rightarrow \varphi$.

Abstract definition of consequence relation

To reason about fundamental question of logical constants, we need an abstract definition of logical consequence.

Definition

A *language* is a triple $L = (Symb, Sent, Tr)$, where:

- $Symb$ is a countable set of *symbols*, partitioned into categories
- $Sent$ is a set of *sentences* over an alphabet containing $Symb$
- $Tr \subseteq Sent$ is the set of *true sentences*.

A *consequence relation* is $\Rightarrow \subseteq \mathcal{P}(Sent) \times Sent$ s.t.

- if $\varphi \in \Gamma$, then $\Gamma \Rightarrow \varphi$
- if $\Delta \Rightarrow \varphi$ and $\Gamma \Rightarrow \psi$ for all $\psi \in \Delta$, then $\Gamma \Rightarrow \varphi$.

A consequence relation is *truth-preserving* if for all $\Gamma \subseteq Tr$ we have: if $\Gamma \Rightarrow \varphi$, then $\varphi \in Tr$.

Consequence from constants

In this abstract setting we use replacement instead of reinterpretation.

Consequence from constants

In this abstract setting we use replacement instead of reinterpretation.

Definition

A *replacement* is a function $\rho : \mathit{Symb} \to \mathit{Symb}$ which respects categories, i.e. $\rho(u)$ is in the same category as u for all $u \in \mathit{Symb}$.

Consequence from constants

In this abstract setting we use replacement instead of reinterpretation.

Definition

A *replacement* is a function $\rho : \mathit{Symb} \to \mathit{Symb}$ which respects categories, i.e. $\rho(u)$ is in the same category as u for all $u \in \mathit{Symb}$. Denote by $\varphi[\rho]$ the result of replacing each occurence of any $u \in \mathit{Symb}$ in φ by $\rho(\varphi)$. Analogously, we use notation $\Gamma[\rho]$.

Consequence from constants

In this abstract setting we use replacement instead of reinterpretation.

Definition

A *replacement* is a function $\rho : Symb \to Symb$ which respects categories, i.e. $\rho(u)$ is in the same category as u for all $u \in Symb$. Denote by $\varphi[\rho]$ the result of replacing each occurence of any $u \in Symb$ in φ by $\rho(\varphi)$. Analogously, we use notation $\Gamma[\rho]$.

Fix $X \subseteq Symb$ (a choice of constants). Put $\Gamma \Rightarrow_X \varphi$ iff for each replacement ρ s.t. $\rho|_X = id_X$ we have: if $\Gamma[\rho] \subseteq Tr$, then $\varphi[\rho] \in Tr$. Then $\Rightarrow_X$ is a truth-preserving consequence relation.

Consequence from constants

In this abstract setting we use replacement instead of reinterpretation.

Definition

A *replacement* is a function $\rho : \mathit{Symb} \to \mathit{Symb}$ which respects categories, i.e. $\rho(u)$ is in the same category as u for all $u \in \mathit{Symb}$. Denote by $\varphi[\rho]$ the result of replacing each occurence of any $u \in \mathit{Symb}$ in φ by $\rho(\varphi)$. Analogously, we use notation $\Gamma[\rho]$.

Fix $X \subseteq \mathit{Symb}$ (a choice of constants). Put $\Gamma \Rightarrow_X \varphi$ iff for each replacement ρ s.t. $\rho|_X = id_X$ we have: if $\Gamma[\rho] \subseteq \mathit{Tr}$, then $\varphi[\rho] \in \mathit{Tr}$. Then $\Rightarrow_X$ is a truth-preserving consequence relation.

Examples

- the standard consequence relation of propositional logic $\models_{PL}$ equals $\Rightarrow_{\{\neg,\vee,\wedge\}}$
- the standard consequence relation of first-order logic $\models_{FO}$ is a subset of $\Rightarrow_{\{\neg,\vee,\wedge,\forall,\exists\}}$

Constants from consequence

Idea: given a language and a consequence relation, a symbol is a constant if replacing it with another symbol of the same category destroys at least one valid inference of that consequence relation.

Constants from consequence

Idea: given a language and a consequence relation, a symbol is a constant if replacing it with another symbol of the same category destroys at least one valid inference of that consequence relation.

Definition

Let $\Rightarrow$ be a truth-preserving consequence relation. We define the set of constants $C_{\Rightarrow}$ by putting $u \in C_{\Rightarrow}$ iff there are Γ, φ and ρ which is identity on $\mathit{Symb} \setminus \{u\}$ s.t. $\Gamma \Rightarrow \varphi$ and $\Gamma[\rho] \not\Rightarrow \varphi[\rho]$.

Constants from consequence

Idea: given a language and a consequence relation, a symbol is a constant if replacing it with another symbol of the same category destroys at least one valid inference of that consequence relation.

Definition

Let $\Rightarrow$ be a truth-preserving consequence relation. We define the set of constants $C_{\Rightarrow}$ by putting $u \in C_{\Rightarrow}$ iff there are Γ, φ and ρ which is identity on $Symb \setminus \{u\}$ s.t. $\Gamma \Rightarrow \varphi$ and $\Gamma[\rho] \not\Rightarrow \varphi[\rho]$.

It is easy to see that for all $X \subseteq Symb$ we have $C_{\Rightarrow_X} \subseteq X$.

Constants from consequence

Idea: given a language and a consequence relation, a symbol is a constant if replacing it with another symbol of the same category destroys at least one valid inference of that consequence relation.

Definition

Let $\Rightarrow$ be a truth-preserving consequence relation. We define the set of constants $C_{\Rightarrow}$ by putting $u \in C_{\Rightarrow}$ iff there are Γ, φ and ρ which is identity on $Symb \setminus \{u\}$ s.t. $\Gamma \Rightarrow \varphi$ and $\Gamma[\rho] \not\Rightarrow \varphi[\rho]$.

It is easy to see that for all $X \subseteq Symb$ we have $C_{\Rightarrow_X} \subseteq X$.

Examples

- $C_{\models_{PL}}$ is the standard set of logical constants of propositional logic
- $C_{\models_{FO}}$ is the standard set of logical constants of first-order logic

A technical difficulty

Example

To see that $\vee$ is in $C_{\models_{PL}}$, note that $p \models_{PL} p \vee q$, but $p \not\models_{PL} p \wedge q$.

A technical difficulty

Example

To see that $\vee$ is in $C_{\models_{PL}}$, note that $p \models_{PL} p \vee q$, but $p \not\models_{PL} p \wedge q$.

But often there is only one symbol of a given category in the alphabet. It is usually convenient not to have too many primitive symbols. Notably, regarding our example, we often have only one binary Boolean connective in the alphabet, while the others are defined as abbreviations.

A technical difficulty

Example

To see that $\vee$ is in $C_{\models_{PL}}$, note that $p \models_{PL} p \vee q$, but $p \not\models_{PL} p \wedge q$.

But often there is only one symbol of a given category in the alphabet. It is usually convenient not to have too many primitive symbols. Notably, regarding our example, we often have only one binary Boolean connective in the alphabet, while the others are defined as abbreviations.

There is a simple solution: in this case it is more convenient to have more symbols (at least two of each category), so we let them be in *Symb* – this does not essentially change the language, while it provides what we need to prove that a symbol is a constant.

Dual symbols

Recall we proved $\vee$ is a constant by replacing it with $\wedge$, which is its dual, i.e. $\varphi \wedge \psi$ is equivalent to $\neg(\neg\varphi \vee \neg\psi)$.

Dual symbols

Recall we proved $\vee$ is a constant by replacing it with $\wedge$, which is its dual, i.e. $\varphi \wedge \psi$ is equivalent to $\neg(\neg\varphi \vee \neg\psi)$.
Idea: if a symbol is unique of its category, introduce its dual to the language and use it to show it is a constant.

Dual symbols

Recall we proved $\vee$ is a constant by replacing it with $\wedge$, which is its dual, i.e. $\varphi \wedge \psi$ is equivalent to $\neg(\neg\varphi \vee \neg\psi)$.
Idea: if a symbol is unique of its category, introduce its dual to the language and use it to show it is a constant.
More general idea: why not always use duals, whether a symbol is unique of its category or not?

Dual symbols

Recall we proved $\vee$ is a constant by replacing it with $\wedge$, which is its dual, i.e. $\varphi \wedge \psi$ is equivalent to $\neg(\neg\varphi \vee \neg\psi)$.
Idea: if a symbol is unique of its category, introduce its dual to the language and use it to show it is a constant.
More general idea: why not always use duals, whether a symbol is unique of its category or not?
Consider further examples.

Examples

- Consider the basic modal logic with the standard (local) consequence relation $\Vdash_{ML}$. To show that $\Box \in C_{\Vdash_{ML}}$, include its dual $\Diamond$ in the language. From duality itself we have $\Box p \Vdash_{ML} \neg\Diamond\neg p$, but $\Diamond p \nVdash_{ML} \neg\Diamond\neg p$.

Dual symbols

Recall we proved $\vee$ is a constant by replacing it with $\wedge$, which is its dual, i.e. $\varphi \wedge \psi$ is equivalent to $\neg(\neg\varphi \vee \neg\psi)$.
Idea: if a symbol is unique of its category, introduce its dual to the language and use it to show it is a constant.
More general idea: why not always use duals, whether a symbol is unique of its category or not?
Consider further examples.

Examples

- Consider the basic modal logic with the standard (local) consequence relation $\Vdash_{ML}$. To show that $\Box \in C_{\Vdash_{ML}}$, include its dual $\Diamond$ in the language. From duality itself we have $\Box p \Vdash_{ML} \neg\Diamond\neg p$, but $\Diamond p \nVdash_{ML} \neg\Diamond\neg p$.
- We have $\forall \in C_{\models_{FO}}$, since $\forall x A \models_{FO} \neg\exists x \neg A$, but $\exists x A \not\models_{FO} \neg\exists x \neg A$.

Arity of a symbol

To apply the idea, we need more structure in the abstract definition of language.

Arity of a symbol

To apply the idea, we need more structure in the abstract definition of language.

Definition

Let $L = (\mathit{Symb}, \mathit{Sent}, \mathit{Tr})$ and let $u \in \mathit{Symb}$. For any $\varphi \in \mathit{Sent}$ in which u occurs, and for any occurrence of u in φ, let ψ be the subsentence which contains this occurrence of u, but no subsentence of ψ contains this occurrence of u.

Arity of a symbol

To apply the idea, we need more structure in the abstract definition of language.

Definition

Let $L = (Symb, Sent, Tr)$ and let $u \in Symb$. For any $\varphi \in Sent$ in which u occurs, and for any occurrence of u in φ, let ψ be the subsentence which contains this occurrence of u, but no subsentence of ψ contains this occurrence of u.
Arity of $u \in Symb$ is $k \in \mathbb{N}$ s.t. each such ψ has exactly k maximal proper subsentences $\psi_1, \ldots, \psi_k$. We denote ψ by $u(\psi_1, \ldots, \psi_k)$.

Arity of a symbol

To apply the idea, we need more structure in the abstract definition of language.

Definition

Let $L = (Symb, Sent, Tr)$ and let $u \in Symb$. For any $\varphi \in Sent$ in which u occurs, and for any occurrence of u in φ, let ψ be the subsentence which contains this occurrence of u, but no subsentence of ψ contains this occurrence of u.
Arity of $u \in Symb$ is $k \in \mathbb{N}$ s.t. each such ψ has exactly k maximal proper subsentences $\psi_1, \ldots, \psi_k$. We denote ψ by $u(\psi_1, \ldots, \psi_k)$.

Examples

In the sense of the above definition:

- $\neg$, $\Diamond$, $\Box$, $\forall$, $\exists$ are unary
- $\vee$, $\wedge$, $\rightarrow$ are binary

Languages with duals

Definition

We say that $L = (Symb, Sent, Tr)$ is a *language with (classical) negation* if there is $\neg \in Symb$ s.t. for all $\varphi \in Sent$ we have also $\neg\varphi \in Sent$ and $\varphi \in Tr$ iff $\neg\varphi \notin Tr$.

Languages with duals

Definition

We say that $L = (Symb, Sent, Tr)$ is a *language with (classical) negation* if there is $\neg \in Symb$ s.t. for all $\varphi \in Sent$ we have also $\neg\varphi \in Sent$ and $\varphi \in Tr$ iff $\neg\varphi \notin Tr$.
We say that a language with negation L is a *language with duals* if for all $k > 0$ and for each k-ary symbol u, there is a k-ary symbol u' of the same category s.t. $u'(\psi_1, \ldots, \psi_k) \in Tr$ iff $\neg u(\neg\psi_1, \ldots, \neg\psi_k) \in Tr$.

Languages with duals

Definition

We say that $L = (Symb, Sent, Tr)$ is a *language with (classical) negation* if there is $\neg \in Symb$ s.t. for all $\varphi \in Sent$ we have also $\neg\varphi \in Sent$ and $\varphi \in Tr$ iff $\neg\varphi \notin Tr$.
We say that a language with negation L is a *language with duals* if for all $k > 0$ and for each k-ary symbol u, there is a k-ary symbol u' of the same category s.t. $u'(\psi_1, \ldots, \psi_k) \in Tr$ iff $\neg u(\neg\psi_1, \ldots, \neg\psi_k) \in Tr$.

Examples

- propositional logic: $\vee$ and $\wedge$, $\rightarrow$ and $\not\leftarrow$
- first-order logic: $\forall$ and $\exists$
- basic modal logic: $\Box$ and $\Diamond$

Conclusion

Theorem

Let L be a language with duals, $k > 0$, and u any k-ary symbol. Let $\Rightarrow$ be s.t. $u'(\psi_1, \ldots, \psi_k) \Leftrightarrow \neg u(\neg\psi_1, \ldots, \neg\psi_k)$ (in particular, this holds for the maximal truth-preserving $\Rightarrow$ on L). If u is not self-dual, then it is a constant.

Conclusion

Theorem

Let L be a language with duals, $k > 0$, and u any k-ary symbol. Let $\Rightarrow$ be s.t. $u'(\psi_1, \ldots, \psi_k) \Leftrightarrow \neg u(\neg\psi_1, \ldots, \neg\psi_k)$ (in particular, this holds for the maximal truth-preserving $\Rightarrow$ on L). If u is not self-dual, then it is a constant.

Proof.

Let ρ be a replacement s.t. $\rho(u) = u'$ which is identity on $Symb \setminus \{u\}$. Then $u'(\psi_1, \ldots, \psi_k) \not\Rightarrow \neg u'(\neg\psi_1, \ldots, \neg\psi_k)$ or $\neg u'(\neg\psi_1, \ldots, \neg\psi_k) \not\Rightarrow u'(\psi_1, \ldots, \psi_k)$ (otherwise u is self-dual). □

Conclusion

Theorem

Let L be a language with duals, $k > 0$, and u any k-ary symbol. Let $\Rightarrow$ be s.t. $u'(\psi_1, \ldots, \psi_k) \Leftrightarrow \neg u(\neg\psi_1, \ldots, \neg\psi_k)$ (in particular, this holds for the maximal truth-preserving $\Rightarrow$ on L). If u is not self-dual, then it is a constant.

Proof.

Let ρ be a replacement s.t. $\rho(u) = u'$ which is identity on $Symb \setminus \{u\}$. Then $u'(\psi_1, \ldots, \psi_k) \not\Rightarrow \neg u'(\neg\psi_1, \ldots, \neg\psi_k)$ or $\neg u'(\neg\psi_1, \ldots, \neg\psi_k) \not\Rightarrow u'(\psi_1, \ldots, \psi_k)$ (otherwise u is self-dual). □

Questions:

- What about self-duals? Idea: replace by a symbol of the same type which is not self-dual to prove it is a constant.

Conclusion

Theorem

Let L be a language with duals, $k > 0$, and u any k-ary symbol. Let $\Rightarrow$ be s.t. $u'(\psi_1, \ldots, \psi_k) \Leftrightarrow \neg u(\neg\psi_1, \ldots, \neg\psi_k)$ (in particular, this holds for the maximal truth-preserving $\Rightarrow$ on L). If u is not self-dual, then it is a constant.

Proof.

Let ρ be a replacement s.t. $\rho(u) = u'$ which is identity on $Symb \setminus \{u\}$. Then $u'(\psi_1, \ldots, \psi_k) \not\Rightarrow \neg u'(\neg\psi_1, \ldots, \neg\psi_k)$ or $\neg u'(\neg\psi_1, \ldots, \neg\psi_k) \not\Rightarrow u'(\psi_1, \ldots, \psi_k)$ (otherwise u is self-dual). □

Questions:

- What about self-duals? Idea: replace by a symbol of the same type which is not self-dual to prove it is a constant.
- What about 0-ary symbols? Example: $\top \models_{PL} \neg\bot$, but $\bot \not\models_{PL} \neg\bot$ (we can say $\top$ and $\bot$ are dual if we allow $k = 0$).