Chapter udf

Syntax and Semantics

syn.1 Introduction

mvl:syn:int:sec In classical logic, we deal with formulas that are built from propositional variables using the propositional connectives $\neg$, $\wedge$, $\vee$, $\rightarrow$, and $\leftrightarrow$. When we define a semantics for classical logic, we do so using the two truth values $\mathbb{T}$ and $\mathbb{F}$. We interpret propositional variables in a valuation $\mathfrak{v}$, which assigns these truth values $\mathbb{T}$, $\mathbb{F}$ to the propositional variables. Any valuation then determines a truth value $\overline{\mathfrak{v}}(\varphi)$ for any formula φ, and A formula is satisfied in a valuation $\mathfrak{v}$, $\mathfrak{v} \vDash \varphi$, iff $\overline{\mathfrak{v}}(\varphi) = \mathbb{T}$.

Many-valued logics are generalizations of classical two-valued logic by allowing more truth values than just $\mathbb{T}$ and $\mathbb{F}$. So in many-valued logic, a valuation $\mathfrak{v}$ is a function assigning to every propositional variable p one of a range of possible truth values. We'll generally call the set of allowed truth values V. Classical logic is a many-valued logic where $V = \{\mathbb{T}, \mathbb{F}\}$, and the truth value $\overline{\mathfrak{v}}(\varphi)$ is computed using the familiar characteristic truth tables for the connectives.

Once we add additional truth values, we have more than one natural option for how to compute $\overline{\mathfrak{v}}(\varphi)$ for the connectives we read as "and," "or," "not," and "if—then." So a many-valued logic is determined not just by the set of truth values, but also by the *truth functions* we decide to use for each connective. Once these are selected for a many-valued logic $\mathbf{L}$, however, the truth value $\overline{\mathfrak{v}}_{\mathbf{L}}(\varphi)$ is uniquely determined by the valuation, just like in classical logic. Many-valued logics, like classical logic, are *truth functional*.

With this semantic building blocks in hand, we can go on to define the analogs of the semantic concepts of tautology, entailment, and satisfiability. In classical logic, a formula is a tautology if its truth value $\overline{\mathfrak{v}}(\varphi) = \mathbb{T}$ for any $\mathfrak{v}$. In many-valued logic, we have to generalize this a bit as well. First of all, there is no requirement that the set of truth values V contains $\mathbb{T}$. For instance, some many-valued logics use numbers, such as all rational numbers between 0 and 1 as their set of truth values. In such a case, 1 usually plays the rule of $\mathbb{T}$. In other logics, not just one but several truth values do. So, we require that every many-valued logic have a set V^+ of *designated values*. We can then say that a formula is satisfied in a valuation $\mathfrak{v}$, $\mathfrak{v} \vDash_{\mathbf{L}} \varphi$, iff $\overline{\mathfrak{v}}_{\mathbf{L}}(\varphi) \in V^+$. A formula φ is a tautology of the logic, $\vDash_{\mathbf{L}} \varphi$, iff $\overline{\mathfrak{v}}(\varphi) \in V^+$ for any $\mathfrak{v}$. And, finally, we say that φ is entailed by a set of formulas, $\Gamma \vDash_{\mathbf{L}} \varphi$, if every valuation that satisfies all the formulas in Γ also satisfies φ.

syn.2 Languages and Connectives

Classical propositional logic, and many other logics, use a set supply of *propositional constants* and *connectives*. For instance, we use the following as primitives: mvl:syn:con:sec

1. The propositional constant for falsity $\bot$.
2. The propositional constant for truth $\top$.
3. The logical connectives: $\neg$ (negation), $\wedge$ (conjunction), $\vee$ (disjunction), $\rightarrow$ (conditional), $\leftrightarrow$ (biconditional)

The same connectives are used in many-valued logics as well. However, it is often useful to include different versions of, say, conjunction, in the same logic, and that would require different symbols to keep the versions separate. Some many-valued logics also include connectives that have no equivalent in classical logic. So, we'll be a bit more general than usual.

Definition syn.1. A *propositional language* consists of a set $\mathcal{L}$ of *connectives*. Each connective $\star$ has an *arity*; a connective of arity n is said to be *n-place*. Connectives of arity 0 are also called *constants*; connectives of arity 1 are called *unary*, and connectives of arity 2, *binary*.

Example syn.2. The standard language of propositional logic $\mathcal{L}_0$ consists of the following connectives (with associated arities): $\bot$ (0) $\neg$ (1), $\wedge$ (2), $\vee$ (2), $\rightarrow$ (2). Most logics we consider will use this language. Some logics by tradition an convention use different symbols for some connectives. For instance, in product logic, the conjunction symbol is often $\odot$ instead of $\wedge$. Sometimes it is convenient to add a new operator, e.g., the determinateness operator $\triangle$ (1-place).

syn.3 Formulas

mvl:syn:fml:sec

Definition syn.3 (Formula). The set $\mathrm{Frm}(\mathcal{L})$ of *formulas* of a propositional language $\mathcal{L}$ is defined inductively as follows: mvl:syn:fml:defn:formulas

1. Every propositional variable p_i is an atomic formula.
2. Every 0-place connective (propositional constant) of $\mathcal{L}$ is an atomic formula.

3. If $\star$ is an n-place connective of $\mathcal{L}$, and $\varphi_1, \dots, \varphi_n$ are formulas, then $\star(\varphi_1, \dots, \varphi_n)$ is a formula.

4. Nothing else is a formula.

If $\star$ is 1-place, then $\star(\varphi_1)$ will often be written simply as $\star\varphi_1$. If $\star$ is 2-place $\star(\varphi_1, \varphi_2)$ will often be written as $(\varphi_1 \star \varphi_2)$.

As usual, we will often silently leave out the outermost parentheses.

Example syn.4. In the standard language $\mathcal{L}_0$, $p_1 \to (p_1 \land \lnot p_2)$ is a formula. In the language of product logic, it would be written instead as $p_1 \to (p_1 \odot \lnot p_2)$. If we add the 1-place $\triangle$ to the language, we would also have formulas such as $\triangle(p_1 \land p_2) \to (\triangle p_1 \land \triangle p_2)$.

syn.4 Matrices

mvl:syn:mat:sec A many-valued logic is defined by its language, its set of truth values V, a subset of designated truth values, and truth functions for its connective. Together, these elements are called a *matrix*.

mvl:syn:mat:defn:matrix **Definition syn.5 (Matrix).** A *matrix* for the logic **L** consists of:

1. a set of connectives making up a language $\mathcal{L}$;

2. a set $V \neq \emptyset$ of truth values;

3. a set $V^+ \subseteq V$ of designated truth values;

4. for each n-place connective $\star$ in $\mathcal{L}$, a truth function $\widetilde{\star} : V^n \to V$. If $n = 0$, then $\widetilde{\star}$ is just an element of V.

Example syn.6. The matrix for classical logic **C** consists of:

1. The standard propositional language $\mathcal{L}_0$ with $\bot$, $\lnot$, $\land$, $\lor$, $\to$.

2. The set of truth values $V = \{\mathbb{T}, \mathbb{F}\}$.

3. $\mathbb{T}$ is the only designated value, i.e., $V^+ = \{\mathbb{T}\}$.

4. For $\bot$, we have $\widetilde{\bot} = \mathbb{F}$. The other truth functions are given by the usual truth tables (see Figure syn.1).

$\widetilde{\neg}$	
$\mathbb{T}$	$\mathbb{F}$
$\mathbb{F}$	$\mathbb{T}$

$\widetilde{\land}$	$\mathbb{T}$	$\mathbb{F}$
$\mathbb{T}$	$\mathbb{T}$	$\mathbb{F}$
$\mathbb{F}$	$\mathbb{F}$	$\mathbb{F}$

$\widetilde{\lor}$	$\mathbb{T}$	$\mathbb{F}$
$\mathbb{T}$	$\mathbb{T}$	$\mathbb{T}$
$\mathbb{F}$	$\mathbb{T}$	$\mathbb{F}$

$\widetilde{\rightarrow}$	$\mathbb{T}$	$\mathbb{F}$
$\mathbb{T}$	$\mathbb{T}$	$\mathbb{F}$
$\mathbb{F}$	$\mathbb{T}$	$\mathbb{T}$

Figure syn.1: Truth functions for classical logic **C**.

mvl:syn:mat:fig:tf-CL

syn.5 Valuations and Satisfaction

mvl:syn:val:sec

Definition syn.7 (Valuations). Let V be a set of truth values. A *valuation* for $\mathcal{L}$ into V is a function $\mathfrak{v}$ assigning an element of V to the propositional variables of the language, i.e., $\mathfrak{v}\colon \mathrm{At}_0 \to V$.

Definition syn.8. Given a valuation $\mathfrak{v}$ into the set of truth values V of a many-valued logic **L**, define the evaluation function $\overline{\mathfrak{v}}\colon \mathrm{Frm}(\mathcal{L}) \to V$ inductively by: mvl:syn:val:defn:pValue

1. $\overline{\mathfrak{v}}(p_n) = \mathfrak{v}(p_n)$;
2. If $\star$ is a 0-place connective, then $\overline{\mathfrak{v}}(\star) = \widetilde{\star}_{\mathbf{L}}$;
3. If $\star$ is an n-place connective, then

$$\overline{\mathfrak{v}}(\star(\varphi_1, \dots, \varphi_n)) = \widetilde{\star}_{\mathbf{L}}(\overline{\mathfrak{v}}(\varphi_1), \dots, \overline{\mathfrak{v}}(\varphi_n)).$$

Definition syn.9 (Satisfaction). The formula φ is *satisfied* by a valuation $\mathfrak{v}$, $\mathfrak{v} \vDash_{\mathbf{L}} \varphi$, iff $\overline{\mathfrak{v}}_{\mathbf{L}}(\varphi) \in V^+$, where V^+ is the set of designated truth values of **L**. mvl:syn:val:defn:satisfaction

We write $\mathfrak{v} \nvDash_{\mathbf{L}} \varphi$ to mean "not $\mathfrak{v} \vDash_{\mathbf{L}} \varphi$." If Γ is a set of formulas, $\mathfrak{v} \vDash_{\mathbf{L}} \Gamma$ iff $\mathfrak{v} \vDash_{\mathbf{L}} \varphi$ for every $\varphi \in \Gamma$.

syn.6 Semantic Notions

Suppose a many-valued logic **L** is given by a matrix. Then we can define the usual semantic notions for **L**. mvl:syn:sem:sec

Definition syn.10.

1. A formula φ is *satisfiable* if for some $\mathfrak{v}$, $\mathfrak{v} \vDash \varphi$; it is *unsatisfiable* if for no $\mathfrak{v}$, $\mathfrak{v} \vDash \varphi$;
2. A formula φ is a *tautology* if $\mathfrak{v} \vDash \varphi$ for all valuations v;
3. If Γ is a set of formulas, $\Gamma \vDash \varphi$ ("Γ entails φ") if and only if $\mathfrak{v} \vDash \varphi$ for every valuation $\mathfrak{v}$ for which $\mathfrak{v} \vDash \Gamma$.
4. If Γ is a set of formulas, Γ is *satisfiable* if there is a valuation $\mathfrak{v}$ for which $\mathfrak{v} \vDash \Gamma$, and Γ is *unsatisfiable* otherwise.

We have some of the same facts for these notions as we do for the case of classical logic:

mvl:syn:sem: prop:semanticalfacts **Proposition syn.11.**

1. *φ is a tautology if and only if $\emptyset \vDash \varphi$;*
2. *If Γ is satisfiable then every finite subset of Γ is also satisfiable;*
3. mvl:syn:sem: def:Monotony *Monotony: if $\Gamma \subseteq \Delta$ and $\Gamma \vDash \varphi$ then also $\Delta \vDash \varphi$;*
4. mvl:syn:sem: def:Cut *Transitivity: if $\Gamma \vDash \varphi$ and $\Delta \cup \{\varphi\} \vDash \psi$ then $\Gamma \cup \Delta \vDash \psi$;*

Proof. Exercise. □

Problem syn.1. Prove Proposition syn.11

In classical logic we can connect entailment and the conditional. For instance, we have the validity of *modus ponens*: If $\Gamma \vDash \varphi$ and $\Gamma \vDash \varphi \to \psi$ then $\Gamma \vDash \psi$. Another important relationship between $\vDash$ and $\to$ in classical logic is the semantic deduction theorem: $\Gamma \vDash \varphi \to \psi$ if and only if $\Gamma \cup \{\varphi\} \vDash \psi$. These results *do not* always hold in many-valued logics. Whether they do depends on the truth function $\widetilde{\to}$.

syn.7 Many-valued logics as sublogics of C

mvl:syn:sub: sec The usual many-valued logics are all defined using matrices in which the value of a truth-function for arguments in $\{\mathbb{T}, \mathbb{F}\}$ agrees with the classical truth functions. Specifically, in these logics, if $x \in \{\mathbb{T}, \mathbb{F}\}$, then $\widetilde{\neg}_{\mathbf{L}}(x) = \widetilde{\neg}_{\mathbf{C}}(x)$, and for $\star$ any one of $\land$, $\lor$, $\to$, if $x, y \in \{\mathbb{T}, \mathbb{F}\}$, then $\widetilde{\star}_{\mathbf{L}}(x, y) = \widetilde{\star}_{\mathbf{C}}(x, y)$. In other words, the truth functions for $\neg$, $\land$, $\lor$, $\to$ restricted to $\{\mathbb{T}, \mathbb{F}\}$ are exactly the classical truth functions.

mvl:syn:sub: prop:mvl-cl **Proposition syn.12.** *Suppose that a many-valued logic $\mathbf{L}$ contains the connectives $\neg$, $\land$, $\lor$, $\to$ in its language, $\mathbb{T}, \mathbb{F} \in V$, and its truth functions satisfy:*

1. mvl:syn:sub: prop:not $\widetilde{\neg}_{\mathbf{L}}(x) = \widetilde{\neg}_{\mathbf{C}}(x)$ *if $x = \mathbb{T}$ or $x = \mathbb{F}$;*
2. mvl:syn:sub: prop:land $\widetilde{\land}_{\mathbf{L}}(x, y) = \widetilde{\land}_{\mathbf{C}}(x, y)$,
3. mvl:syn:sub: prop:lor $\widetilde{\lor}_{\mathbf{L}}(x, y) = \widetilde{\lor}_{\mathbf{C}}(x, y)$,
4. mvl:syn:sub: prop:lif $\widetilde{\to}_{\mathbf{L}}(x, y) = \widetilde{\to}_{\mathbf{C}}(x, y)$, *if $x, y \in \{\mathbb{T}, \mathbb{F}\}$.*

Then, for any valuation $\mathfrak{v}$ into V such that $\mathfrak{v}(p) \in \{\mathbb{T}, \mathbb{F}\}$, $\overline{\mathfrak{v}}_{\mathbf{L}}(\varphi) = \overline{\mathfrak{v}}_{\mathbf{C}}(\varphi)$.

Proof. By induction on φ.

1. If $\varphi \equiv p$ is atomic, we have $\overline{\mathfrak{v}}_{\mathbf{L}}(\varphi) = \mathfrak{v}(p) = \overline{\mathfrak{v}}_{\mathbf{C}}(\varphi)$.

2. If $\varphi \equiv \neg B$, we have

$$\begin{aligned}
\overline{\mathfrak{v}}_{\mathbf{L}}(\varphi) &= \widetilde{\neg}_{\mathbf{L}}(\overline{\mathfrak{v}}_{\mathbf{L}}(\psi)) && \text{by Definition syn.8} \\
&= \widetilde{\neg}_{\mathbf{L}}(\overline{\mathfrak{v}}_{\mathbf{C}}(\psi)) && \text{by inductive hypothesis} \\
&= \widetilde{\neg}_{\mathbf{C}}(\overline{\mathfrak{v}}_{\mathbf{C}}(\psi)) && \text{by assumption (1),} \\
& && \text{since } \overline{\mathfrak{v}}_{\mathbf{C}}(\psi) \in \{\mathbb{T}, \mathbb{F}\}, \\
&= \overline{\mathfrak{v}}_{\mathbf{C}}(\varphi) && \text{by Definition syn.8.}
\end{aligned}$$

3. If $\varphi \equiv (\psi \wedge \chi)$, we have

$$\begin{aligned}
\overline{\mathfrak{v}}_{\mathbf{L}}(\varphi) &= \widetilde{\wedge}_{\mathbf{L}}(\overline{\mathfrak{v}}_{\mathbf{L}}(\psi), \overline{\mathfrak{v}}_{\mathbf{L}}(\chi)) && \text{by Definition syn.8} \\
&= \widetilde{\wedge}_{\mathbf{L}}(\overline{\mathfrak{v}}_{\mathbf{C}}(\psi), \overline{\mathfrak{v}}_{\mathbf{C}}(\chi)) && \text{by inductive hypothesis} \\
&= \widetilde{\wedge}_{\mathbf{C}}(\overline{\mathfrak{v}}_{\mathbf{C}}(\psi), \overline{\mathfrak{v}}_{\mathbf{C}}(\chi)) && \text{by assumption (2),} \\
& && \text{since } \overline{\mathfrak{v}}_{\mathbf{C}}(\psi), \overline{\mathfrak{v}}_{\mathbf{C}}(\chi) \in \{\mathbb{T}, \mathbb{F}\}, \\
&= \overline{\mathfrak{v}}_{\mathbf{C}}(\varphi) && \text{by Definition syn.8.}
\end{aligned}$$

The cases where $\varphi \equiv (\psi \vee \chi)$ and $\varphi \equiv (\psi \to \chi)$ are similar. □

Corollary syn.13. *If a many-valued logic satisfies the conditions of Proposition syn.12, $\mathbb{T} \in V^+$ and $\mathbb{F} \notin V^+$, then $\vDash_{\mathbf{L}} \subseteq \vDash_{\mathbf{C}}$, i.e., if $\Gamma \vDash_{\mathbf{L}} \psi$ then $\Gamma \vDash_{\mathbf{C}} \psi$. In particular, every tautology of* **L** *is also a classical tautology.*

Proof. We prove the contrapositive. Suppose $\Gamma \nvDash_{\mathbf{C}} \psi$. Then there is some valuation $\mathfrak{v}\colon \mathrm{At}_0 \to \{\mathbb{T}, \mathbb{F}\}$ such that $\overline{\mathfrak{v}}_{\mathbf{C}}(\varphi) = \mathbb{T}$ for all $\varphi \in \Gamma$ and $\overline{\mathfrak{v}}_{\mathbf{C}}(\psi) = \mathbb{F}$. Since $\mathbb{T}, \mathbb{F} \in V$, the valuation $\mathfrak{v}$ is also a valuation for **L**. By Proposition syn.12, $\overline{\mathfrak{v}}_{\mathbf{L}}(\varphi) = \mathbb{T}$ for all $\varphi \in \Gamma$ and $\overline{\mathfrak{v}}_{\mathbf{L}}(\psi) = \mathbb{F}$. Since $\mathbb{T} \in V^+$ and $\mathbb{F} \notin V^+$ that means $\mathfrak{v} \vDash_{\mathbf{L}} \Gamma$ and $\mathfrak{v} \nvDash_{\mathbf{L}} \psi$, i.e., $\Gamma \nvDash_{\mathbf{L}} \psi$. □

Photo Credits

Bibliography