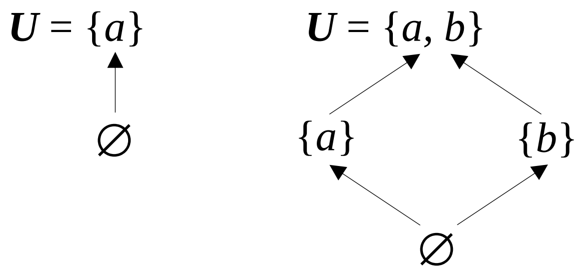
Propositional Logic

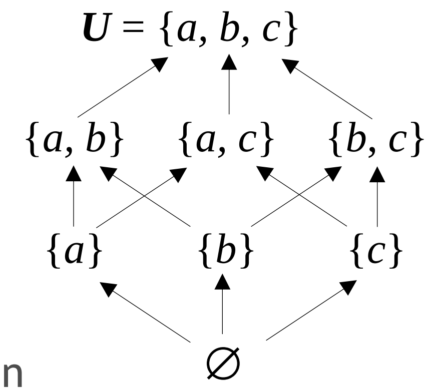
Marco Piastra

Boolean algebras by examples

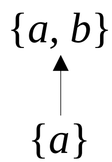
Start from a *finite* set of objects $\boldsymbol{U}$
and construct, in a *bottom-up fashion*, the collection $\boldsymbol{X}$ of all possible subsets of $\boldsymbol{U}$

Examples:





...



The arrows
represents proper inclusion
$\{a\} \subset \{a, b\}$

(Hasse diagrams)

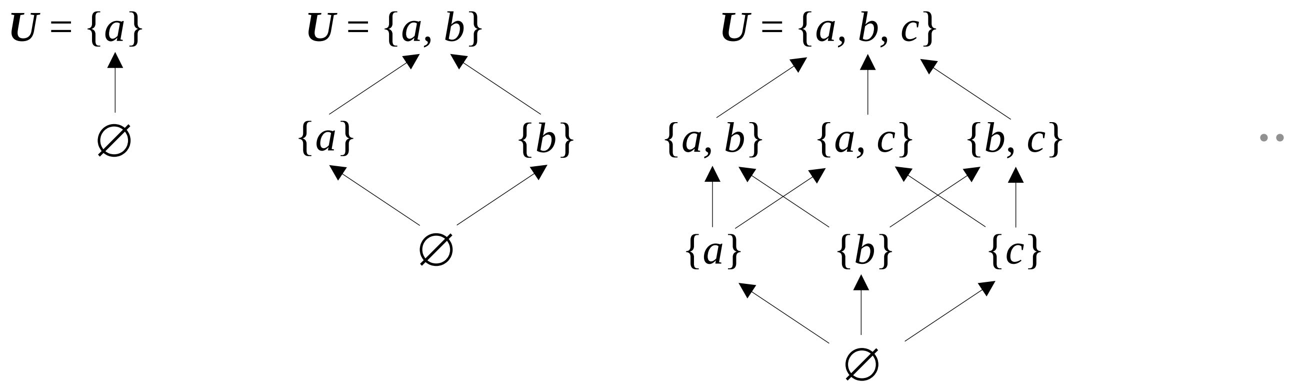
The collections $\boldsymbol{X}$ above are also examples of what is called the ***power set*** of $\boldsymbol{U}$
(i.e. the collection of all possible subsets of $\boldsymbol{U}$) which is denoted as $\boldsymbol{2^U}$ (i.e. $\boldsymbol{X} = \boldsymbol{2^U}$)

Consider the *power set* $\boldsymbol{X}$ of $\boldsymbol{U}$ together with the operations $\cup$, $\cap$, ^{c}
(i.e. *union*, *intersection* and *absolute complement*):
then the structure $< \boldsymbol{X}, \cup, \cap, ^c, \varnothing, \boldsymbol{U} >$ is a **Boolean algebra**

Boolean algebras by examples

Start from a *finite* set of objects $\boldsymbol{U}$
and construct, in a *bottom-up fashion*, the collection $\boldsymbol{X}$ of all possible subsets of $\boldsymbol{U}$

Examples:



Operations ∪, ∩, ^{c} (*union*, *intersection* and *absolute complement, i.e. to* $\boldsymbol{U}$)

For these structures properties can be checked directly

$$A \cup A^c = \boldsymbol{U}$$

$A = \{a\}$
$A^c = \{b, c\}$
$A \cup A^c = \{a, b, c\}$

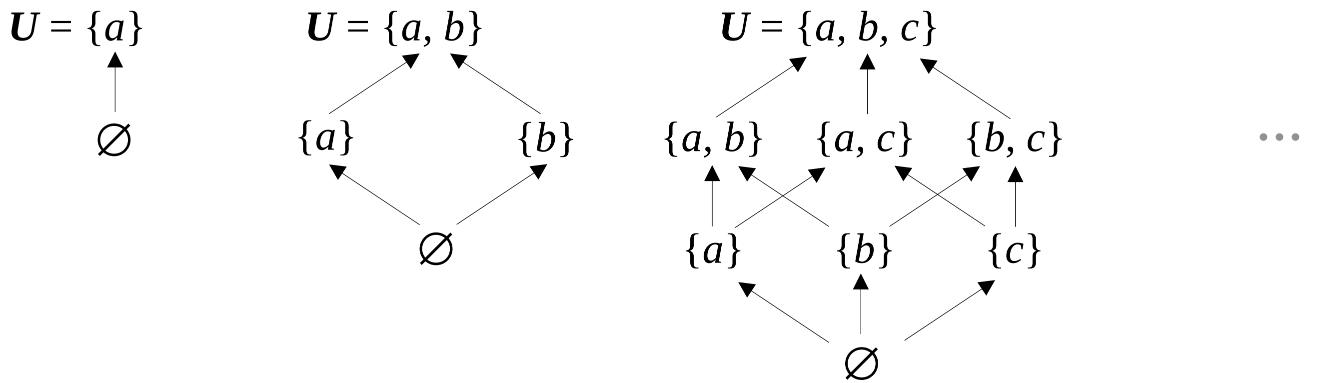
$$A \cap (A \cup B) = A$$

$A = \{b\}$
$B = \{c\}$
$A \cup B = \{b, c\}$
$A \cap (A \cup B) = \{b\}$

Boolean algebras by examples

Start from a *finite* set of objects $\boldsymbol{U}$
and construct, in a *bottom-up fashion*, the collection $\boldsymbol{X}$ of all possible subsets of $\boldsymbol{U}$

Examples:



Operations ∪, ∩, ^{c} (*union*, *intersection* and *absolute complement, i.e. to* $\boldsymbol{U}$)

For these structures properties can be checked directly

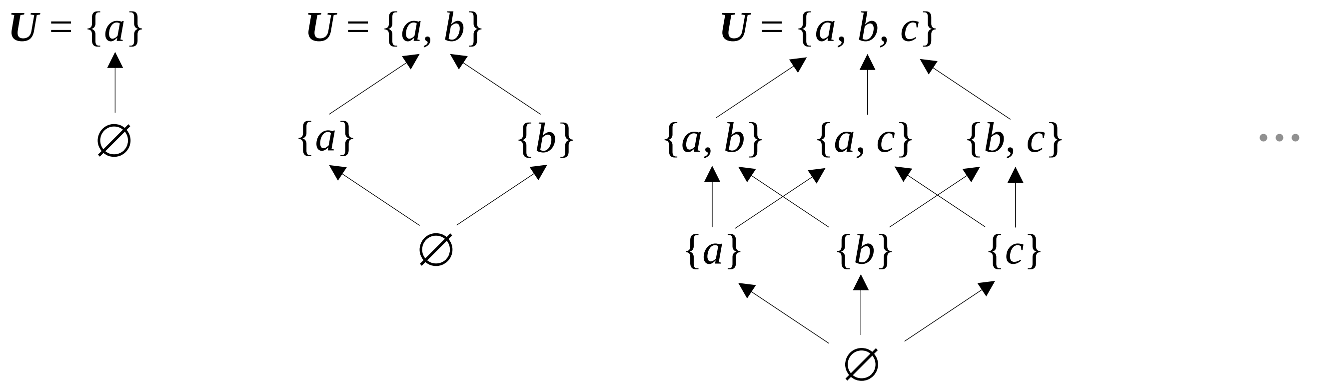
$(A \cup B)^c = A^c \cap B^c$	$(A \cap B)^c = A^c \cap B^c$
$A = \{b\}$	$A = \{b\}$
$A^c = \{a, c\}$	$A^c = \{a, c\}$
$B = \{b, c\}$	$B = \{b, c\}$
$B^c = \{a\}$	$B^c = \{a\}$
$A \cup B = \{b, c\}$	$A \cap B = \{b\}$
$(A \cup B)^c = \{a\}$	$(A \cap B)^c = \{a, c\}$
$A^c \cap B^c = \{a\}$	$A^c \cup B^c = \{a, c\}$

De Morgan's laws

Boolean algebras by examples

Start from a *finite* set of objects $\boldsymbol{U}$
and construct, in a *bottom-up fashion*, the collection $\boldsymbol{X}$ of all possible subsets of $\boldsymbol{U}$

Examples:



Operations $\cup$, $\cap$, ^{c} (*union*, *intersection* and *absolute complement, i.e. to* $\boldsymbol{U}$)

… but sometimes we fail

$A^c \cup B = \boldsymbol{U}$

$A = \{a\}$
$A^c = \{b, c\}$
$B = \{b\}$
$A^c \cup B = \{b, c\}$

** Ouch!*
This is NOT true in general

It is only valid when $A \subseteq B$

Abstract Boolean Algebras

"This type of algebraic structure captures essential properties of both set operations and logic operations." [Wikipedia]

Boolean algebra

Any structure $< \mathbf{X}, \cup, \cap, ^c, \varnothing, \mathbf{U} >$ *that has the following properties (for any* $A, B, C \in \mathbf{X}$*) :*

$A \cup A = A \cap A = A$	*idempotence*
$A \cup B = B \cup A\ ,\quad A \cap B = B \cap A$	*commutativity*
$A \cup (B \cup C) = (A \cup B) \cup C\ ,\quad A \cap (B \cap C) = (A \cap B) \cap C$	*associativity*
$A \cup (A \cap B) = A\ ,\quad A \cap (A \cup B) = A$	*absorption*
$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)\ ,\quad A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$	*distributivity*
$\varnothing \cup A = A\ ,\quad \varnothing \cap A = \varnothing\ ,\ \mathbf{U} \cup A = \mathbf{U}\ ,\ \mathbf{U} \cap A = A$	*special elements*
$A \cup (A^c) = \mathbf{U}\ ,\quad A \cap (A^c) = \varnothing$	*complement*

Which Boolean algebra for logic?

* Given that all boolean algebras share the same properties (*see before*) we can adopt the simplest one as reference, namely the one based on $\boldsymbol{X} = \{\boldsymbol{U}, \varnothing\}$
  i.e. a *two-valued* algebra: $\{nothing, everything\}$ or $\{false, true\}$ or $\{\bot, \top\}$ or $\{0, 1\}$

- Algebraic structure

  $< \{0,1\}, OR, AND, NOT, 0, 1>$

- Boolean functions and *truth tables*

  Boolean functions: $f : \{0, 1\}^n \rightarrow \{0, 1\}$

  AND, *OR* and *NOT* are boolean functions, they are defined via *truth tables*

A	*B*	*OR*
0	0	0
0	1	1
1	0	1
1	1	1

A	*B*	*AND*
0	0	0
0	1	0
1	0	0
1	1	1

A	*NOT*
0	1
1	0

Composite functions

Truth tables can be defined also for composite functions

For example, to verify logical laws

These columns are identical

De Morgan's laws

A	*B*	*NOT A*	*NOT B*	*A OR B*	NOT(*A* OR *B*)	*NOT A AND NOT B*
0	0	1	1	0	1	1
0	1	1	0	1	0	0
1	0	0	1	1	0	0
1	1	0	0	1	0	0

Adequate basis

- How many *basic* boolean functions do we need to define *any* boolean function?

(2^n rows)

A_1	A_2	...	A_n	$f(A_1, A_2, ..., A_n)$
0	0	...	0	f_1
0	0	...	1	f_2
...	...	...	...	...
...	...	...	...	...
1	1	...	1	f_{2^n}

Just *OR*, *AND* and *NOT* : any other function can be expressed as composite function

In the generic *truth table* above:

- For each row where $f = 1$, we compose by *AND* the n input variables taking either A_i when the i-th value is 1, or $\neg A_i$ when i-th value is 0
- We compose by *OR* all the A_i expressions when the i-th value is 1

Other adequate basis

Also {*OR*, *NOT*} o {*AND*, *NOT*} sono basi adeguate

An adequate basis can be obtained by just one 'ad hoc' function: *NOR* or *NAND*

A	*B*	*A NOR B*
0	0	1
0	1	0
1	0	0
1	1	0

A	*B*	*A NAND B*
0	0	1
0	1	1
1	0	1
1	1	0

- Two remarkable functions: *implication* and *equivalence*

Logicians prefer the basis {*IMP*, *NOT*}

A	*B*	*A IMP B*
0	0	1
0	1	1
1	0	0
1	1	1

A	*B*	*A EQU B*
0	0	1
0	1	0
1	0	0
1	1	1

Identities: *A IMP B* = *NOT A OR B* *A EQU B* = (*A IMP B*) AND (*B IMP A*)

Propositional logic

i.e. the simplest of 'classical' logics

- Propositions

  We consider all *possible worlds* that can be described via atomic ***propositions***

  "Today is Friday"
  "Turkeys are birds with feathers"
  "Man is a featherless biped"

- Formal ***language***

  A precise and formal language in which *propositions* are the ***atoms***
  (i.e. no intention to represent the internal structure of *propositions*)

  Atoms can be composed in complex formulae via ***logical connectives***

- Formal ***semantics***

  A class of formal structures, each representing a *possible world*

  Fundamental: in each *possible world*, each formula of the language is either *true* or *false*

  - *Atoms* are given a *truth value* (*i.e. false, true*)
  - Logical connectives are associated to *boolean functions*: each *formula* corresponds to a functional composition in which *atoms* are the arguments (*truth-functionality*)

The class of propositional, semantic structures

They will define the meaning of the formal language (to be defined)

Each possible world is a structure $<\{0,1\}, \boldsymbol{P}, v>$

- $\{0,1\}$ are the *truth values*
- $\boldsymbol{P}$ is the ***signature*** of the formal language: a set of propositional symbols
- v is a *function* $: \boldsymbol{P} \rightarrow \{0,1\}$ assigning truth values to the symbols in $\boldsymbol{P}$

Propositional symbols (*signature*)

- Each symbol in $\boldsymbol{P}$ stands for an actual *proposition* (in natural language)
  - In the simple convention, we use the symbols $A, B, C, D, \ldots$
  - Caution: $\boldsymbol{P}$ is not necessarily *finite*

Possible worlds

- The class of structures contains all possible worlds:

  $<\{0,1\}, \boldsymbol{P}, v>$
  $<\{0,1\}, \boldsymbol{P}, v'>$
  $<\{0,1\}, \boldsymbol{P}, v''>$
  …

  - Each class of structure shares $\boldsymbol{P}$ and $\{0,1\}$
    - The functions v are different: the assignment of truth values varies, depending on the possible world
    - If $\boldsymbol{P}$ is finite, there are only *finitely* many distinct possible worlds (actually $2^{|\boldsymbol{P}|}$)

Propositional *language*

i.e. how we describe the world, by propositions

- In a propositional language L_P

  A set $\boldsymbol{P}$ of propositional symbols: $\boldsymbol{P} = \{A, B, C, \ldots\}$

  Two (primary) ***logical connectives***: $\neg, \rightarrow$

  Three (derived) ***logical connectives***: $\wedge, \vee, \leftrightarrow$

  Parenthesis: (,) (there are no *precedence rules* in this language)

- Well-formed formulae (**wff**)

  A set of syntactic rules

  The set of all the **wff** of L_P is denoted as $\mathrm{wff}(L_P)$

  $A \in \boldsymbol{P} \;\Rightarrow\; A \in \mathrm{wff}(L_P)$

  $\varphi \in \mathrm{wff}(L_P) \;\Rightarrow\; (\neg\varphi) \in \mathrm{wff}(L_P)$

  $\varphi, \psi \in \mathrm{wff}(L_P) \;\Rightarrow\; (\varphi \rightarrow \psi) \in \mathrm{wff}(L_P)$

  $\varphi, \psi \in \mathrm{wff}(L_P) \;\Rightarrow\; (\varphi \vee \psi) \in \mathrm{wff}(L_P), \quad (\varphi \vee \psi) \;\Leftrightarrow\; ((\neg\varphi) \rightarrow \psi)$

  $\varphi, \psi \in \mathrm{wff}(L_P) \;\Rightarrow\; (\varphi \wedge \psi) \in \mathrm{wff}(L_P), \quad (\varphi \wedge \psi) \;\Leftrightarrow\; (\neg(\varphi \rightarrow (\neg\psi)))$

  $\varphi, \psi \in \mathrm{wff}(L_P) \;\Rightarrow\; (\varphi \leftrightarrow \psi) \in \mathrm{wff}(L_P), \quad (\varphi \leftrightarrow \psi) \;\Leftrightarrow\; ((\varphi \rightarrow \psi) \wedge (\psi \rightarrow \varphi))$

Semantics: interpretations

- Composite (i.e. *truth-functional*) semantics for wffs

  Given a possible world $<\{0,1\}, \boldsymbol{P}, v>$
  the function $v : \boldsymbol{P} \rightarrow \{0,1\}$ can be extended to assign a value to *every* wff

  Each logical connective is associated to a binary (i.e. *boolean*) function:

  $v(\neg\varphi) = NOT(v(\varphi))$

  $v(\varphi \wedge \psi) = AND(v(\varphi), v(\psi))$

  $v(\varphi \vee \psi) = OR(v(\varphi), v(\psi))$

  $v(\varphi \rightarrow \psi) = OR(NOT(v(\varphi)), v(\psi))$ (also $IMP(v(\varphi), v(\psi))$)

  $v(\varphi \leftrightarrow \psi) = AND(OR(NOT(v(\varphi)), v(\psi)), OR(NOT(v(\psi)), v(\varphi)))$

- Interpretations

  Function v (extended as above) assigns a truth value <u>*to each*</u> $\varphi \in \text{wff}(L_P)$

  $$v : \text{wff}(L_P) \rightarrow \{0,1\}$$

  Then v is said to be an *interpretation* of L_P

  Note that the truth value of any wff φ is univocally determined
  by the values assigned to each symbol in the *signature* $\boldsymbol{P}$

  Sometimes we will use just v instead of $<\{0,1\}, \boldsymbol{P}, v>$

Satisfaction, models

- Possible worlds and *truth tables*

  Examples: $\varphi = (A \vee B) \wedge C$

  Different rows
  different worlds

  Caution: in each possible world
  every $\varphi \in \mathrm{wff}(L_P)$ has a truth value

A	B	C	$A \vee B$	$(A \vee B) \wedge C$
0	0	0	0	0
0	0	1	0	0
0	1	0	1	0
0	1	1	1	1
1	0	0	1	0
1	0	1	1	1
1	1	0	1	0
1	1	1	1	1

A possible world **satisfies** a wff φ iff $v(\varphi) = 1$

We also write $<\{0,1\}, \boldsymbol{P}, v> \models \varphi$

In the truth table above, the rows that satisfy φ are in gray

Such possible world v is also said to be a **model** of φ

By extension, a possible world *satisfies* (i.e. is *model* of) a set of wff $\Gamma = \{\varphi_1, \varphi_2, \dots, \varphi_n\}$ iff v *satisfies* (i.e. is *model* of) each of its wff $\varphi_1, \varphi_2, \dots, \varphi_n$

Sometimes we will use $v \models \Gamma$ instead of $<\{0,1\}, \boldsymbol{P}, v> \models \Gamma$

Tautologies, contradictions

- A **tautology**

  Is a (propositional) wff that is always satisfied

  It is also said to be **valid**

  Any wff of the type $\varphi \vee \neg\varphi$ is a tautology

- A **contradiction**

  Is a (propositional) wff, that cannot be satisfied

  Any wff of the type $\varphi \wedge \neg\varphi$ is a contradiction

A	$A \wedge \neg A$	$A \vee \neg A$
0	0	1
1	0	1

A	B	$(\neg A \vee B) \vee (\neg B \vee A)$
0	0	1
0	1	1
1	0	1
1	1	1

A	B	$\neg((\neg A \vee B) \vee (\neg B \vee A))$
0	0	0
0	1	0
1	0	0
1	1	0

Note:

- Not all wffs are either tautologies or contradictions
- If φ is a tautology then $\neg\varphi$ is a contradiction and vice-versa

Formulae and subsets

- Consider the set W of all possible worlds

    Each wff of L_P corresponds to a **subset** of W

    i.e. the subset of possible worlds that *satisfy* it

    For example, φ corresponds to $\{v : v(\varphi) = 1\}$ (it can be written also as $\{v : v \models \varphi\}$)

    The corresponding subset may be empty (i.e. if φ is a contradiction)
    or it may coincide with W (i.e if φ is a tautology)

The set of all possible worlds

Formulae and subsets

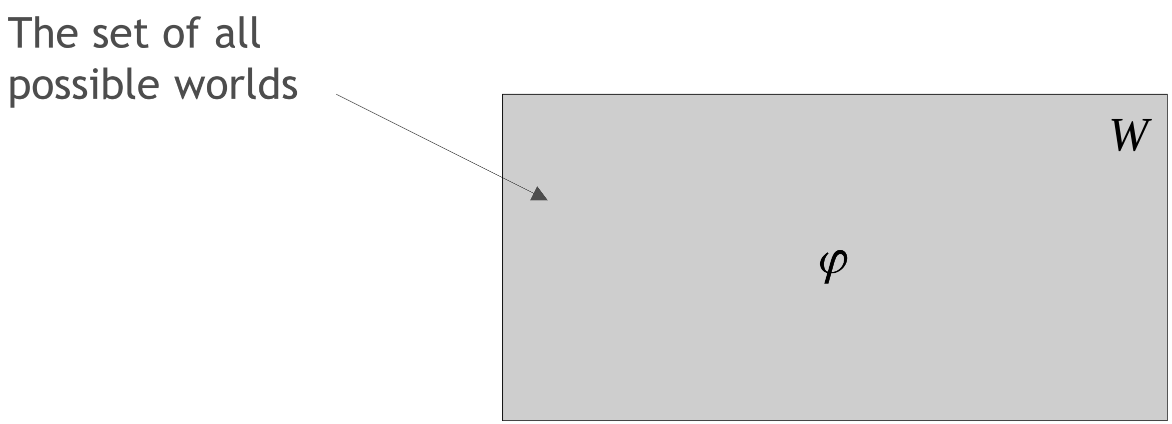
- Consider the set W of all possible worlds

  Each wff of L_P corresponds to a **subset** of W

  i.e. the subset of possible worlds that *satisfy* it

  For example, φ corresponds to $\{v : v(\varphi) = 1\}$ (it can be written also as $\{v : v \models \varphi\}$)

  The corresponding subset may be empty (i.e. if φ is a contradiction)
  or it may coincide with W (i.e if φ is a tautology)



"φ is a tautology"

"any possible world in W is a *model* of φ"

"φ is (logically) ***valid***"

Furthermore:
"φ is ***satisfiable***"
"φ is <u>not</u> ***falsifiable***"

Formulae and subsets

- Consider the set W of all possible worlds

  Each wff of L_P corresponds to a **subset** of W

  i.e. the subset of possible worlds that *satisfy* it

  For example, φ corresponds to $\{v : v(\varphi) = 1\}$ (it can be written also as $\{v : v \models \varphi\}$)

  The corresponding subset may be empty (i.e. if φ is a contradiction)
  or it may coincide with W (i.e if φ is a tautology)

The set of all possible worlds

"φ is a contradiction"

"none of the possible worlds in W is a *model* of φ"

"φ is not (logically) ***valid***"

Furthermore:
"φ is not ***satisfiable***"
"φ is ***falsifiable***"

Formulae and subsets

- Consider the set W of all possible worlds

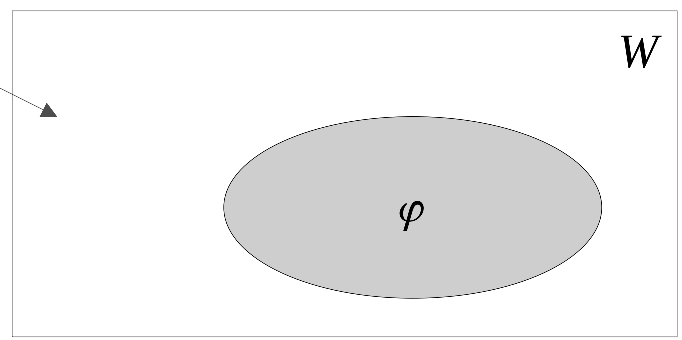
  Each wff of L_P corresponds to a **subset** of W

  i.e. the subset of possible worlds that *satisfy* it

  For example, φ corresponds to $\{v : v(\varphi) = 1\}$ (it can be written also as $\{v : v \models \varphi\}$)

  The corresponding subset may be empty (i.e. if φ is a contradiction) or it may coincide with W (i.e if φ is a tautology)

The set of all possible worlds



"φ is neither a contradiction nor a tautology"

"some possible worlds in W are *model* of φ, others are not"

"φ is <u>not</u> (logically) ***valid***"

Furthermore:
"φ is ***satisfiable***"
"φ is ***falsifiable***"

About formulae and their hidden relations

- Hypothesis:

$\varphi_1 = B \lor D \lor \neg(A \land C)$

"Sally likes Harry" OR "Harry is happy"
OR NOT ("Harry is human" AND "Harry is a featherless biped")

$\varphi_2 = B \lor C$

"Sally likes Harry" OR "Harry is a featherless biped"

$\varphi_3 = A \lor D$

"Harry is human" OR "Harry is happy"

$\varphi_4 = \neg B$

NOT "Sally likes Harry"

- Thesis:

$\psi = D$

"Harry is happy"

Is there any **logical relation** between hypothesis and thesis?

And among the propositions in the hypothesis?

Logical consequence

A	B	C	D	φ_1	φ_2	φ_3	φ_4	ψ
0	0	0	0	1	0	0	1	0
0	0	0	1	1	0	1	1	1
0	0	1	0	1	1	0	1	0
0	0	1	1	1	1	1	1	1
0	1	0	0	1	1	0	0	0
0	1	0	1	1	1	1	0	1
0	1	1	0	1	1	0	0	0
0	1	1	1	1	1	1	0	1
1	0	0	0	1	0	1	1	0
1	0	0	1	1	0	1	1	1
1	0	1	0	0	1	1	1	0
1	0	1	1	1	1	1	1	1
1	1	0	0	1	1	1	0	0
1	1	0	1	1	1	1	0	1
1	1	1	0	1	1	1	0	0
1	1	1	1	1	1	1	0	1

The overall truth table
for the wff in the example

$\varphi_1 = B \vee D \vee \neg(A \wedge C)$
$\varphi_2 = B \vee C$
$\varphi_3 = A \vee D$
$\varphi_4 = \neg B$

$\psi = D$

All the possible worlds that satisfy $\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\}$ satisfy ψ as well

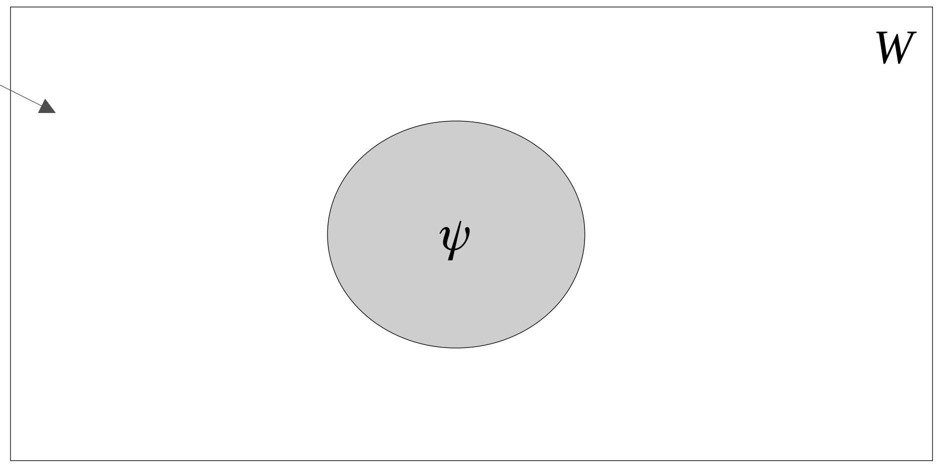
- This is the relation of ***logical consequence***: $\varphi_1, \varphi_2, \varphi_3, \varphi_4 \models \psi$
  (also *logical entailment* or *entailment*)

(Pay attention to notation!)

Formulae, subsets and entailment

- Consider the set of all possible worlds W

All possible worlds

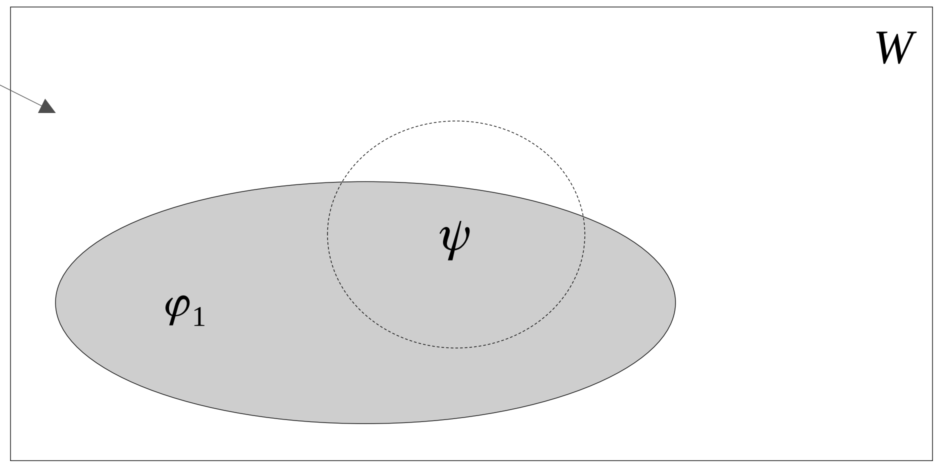


"All possible worlds that are *models* of ψ"

Formulae, subsets and entailment

- Consider the set of all possible worlds W

All possible worlds

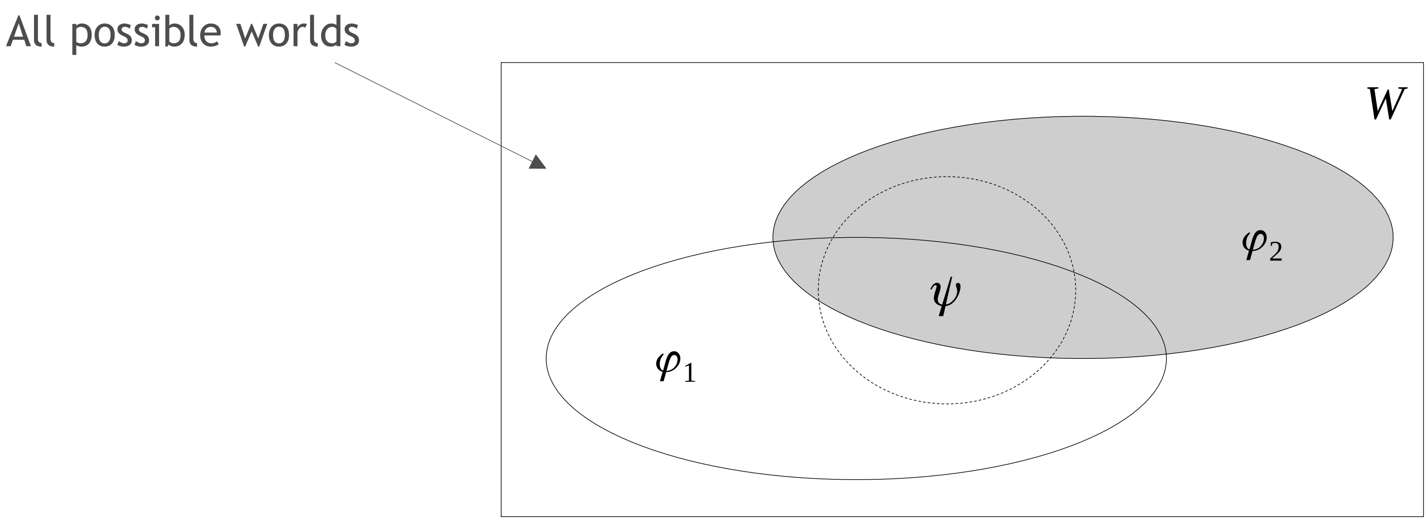


"All possible worlds that are models of φ_1"

$\{ \varphi_1 \} \not\models \psi$
because the set of models of $\{ \varphi_1 \}$
is <u>not</u> contained in the set of models of ψ

Formulae, subsets and entailment

- Consider the set of all possible worlds W



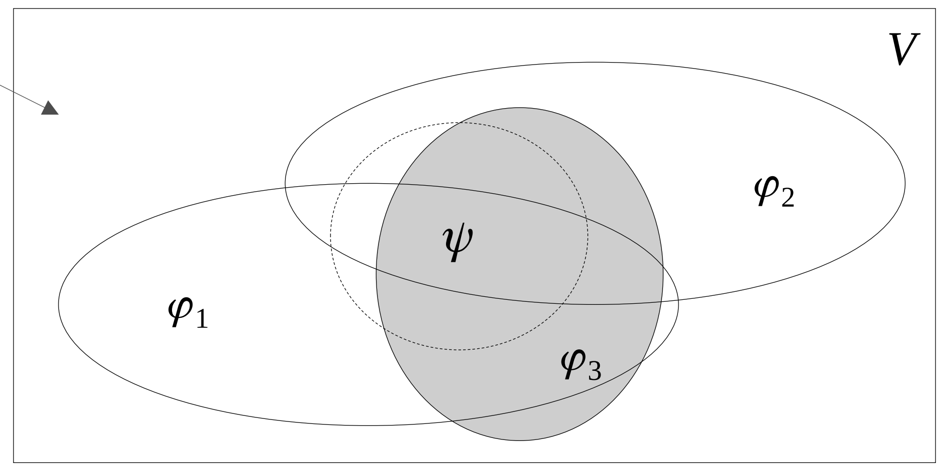
"All possible worlds that are models of φ_2"

$\{ \varphi_1, \varphi_2 \} \not\models \psi$
because the set of models of $\{ \varphi_1, \varphi_2 \}$ (i.e. the *intersection* of the two subsets) is not contained in the set of models of ψ

Formulae, subsets and entailment

- Consider the set of all possible worlds W

All possible worlds



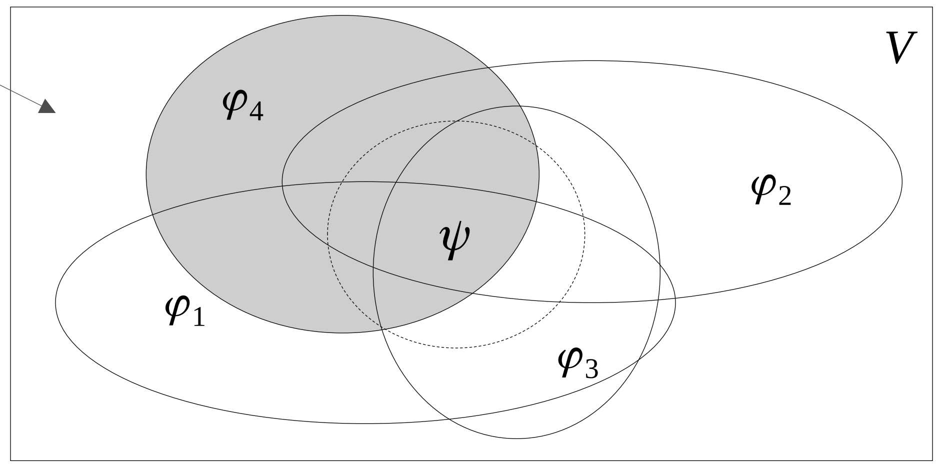
"All possible worlds that are models of φ_3"

$\{ \varphi_1, \varphi_2, \varphi_3\} \not\models \psi$
because the set of models of $\{ \varphi_1, \varphi_2, \varphi_3\}$
is <u>not</u> contained in the set of models of ψ

Formulae, subsets and entailment

- Consider the set of all possible worlds W

All possible worlds



"All possible worlds that are models of φ_4"

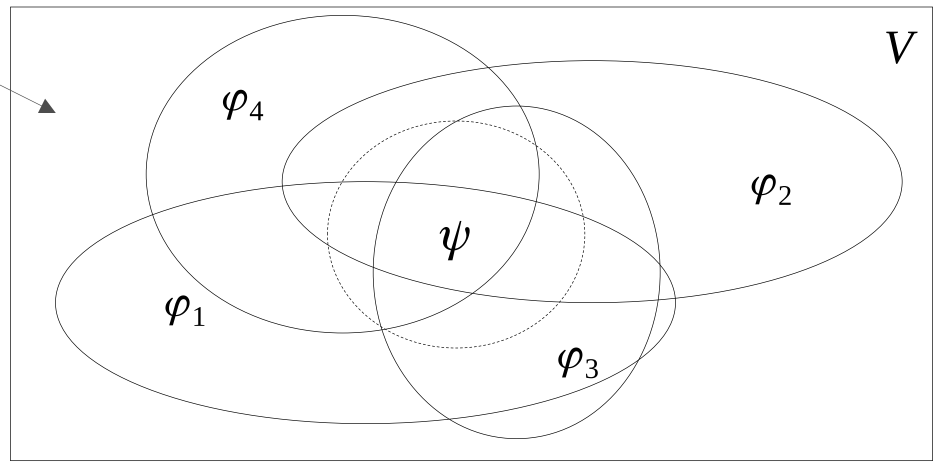
$\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\} \models \psi$

Because the set of models of $\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\}$
is contained in the set of models of ψ

Formulae, subsets and entailment

- Consider the set of all possible worlds W

All possible worlds



"All possible worlds that are models of φ_4"

$\{ \varphi_1, \varphi_2, \varphi_3, \varphi_4\} \models \psi$

Because the set of models of $\{ \varphi_1, \varphi_2, \varphi_3, \varphi_4\}$
<u>is</u> contained in the set of models of ψ

In this case,
all the wffs $\varphi 1, \varphi 2, \varphi 3, \varphi 4$
are needed for the relation
of *entailment* to hold

Symmetric entailment = logical equivalence

- Equivalence

  Let φ and ψ be wffs such that:

  $\varphi \models \psi$ e $\psi \models \varphi$

  The two wffs are also said to be ***logically equivalent***

  In symbols: $\varphi \equiv \psi$

- Substitutability

  Two equivalent wffs have exactly the same ***models***

  In terms of entailment, equivalent wffs are substitutable

  (even as sub-formulae)

  In the example: $\{\varphi_1, \varphi_2, \varphi_3, \varphi_4\} \models \psi$

$$\begin{aligned} \varphi_1 &= B \lor D \lor \neg(A \land C) \\ \varphi_2 &= B \lor C \\ \varphi_3 &= A \lor D \\ \varphi_4 &= \neg B \\ \psi &= D \end{aligned}$$

$$\begin{aligned} \varphi_1 &= B \lor D \lor (A \to \neg C) \\ \varphi_2 &= B \lor C \\ \varphi_3 &= \neg A \to D \\ \varphi_4 &= \neg B \\ \psi &= D \end{aligned}$$

Implication

The wffs of the problem can be re-written using equivalent expressions:

(using the basis $\{\rightarrow, \neg\}$)

$\varphi_1 = C \rightarrow (\neg B \rightarrow (A \rightarrow D))$
$\varphi_2 = \neg B \rightarrow C$
$\varphi_3 = \neg A \rightarrow D$
$\varphi_4 = \neg B$
$\psi = D$

$\varphi_1 = B \vee D \vee \neg(A \wedge C)$
$\varphi_2 = B \vee C$
$\varphi_3 = A \vee D$
$\varphi_4 = \neg B$
$\psi = D$

- Some schemes are *valid* in terms of *entailment*:

$$\begin{array}{l} \varphi \rightarrow \psi \\ \varphi \\ \hline \psi \end{array}$$

It can be verified that:

$\varphi \rightarrow \psi, \varphi \models \psi$

Analogously:

$\varphi \rightarrow \psi, \neg\psi \models \neg\varphi$

Modern formal logic: fundamentals

- **Formal language (*symbolic*)**

  A set of symbols, not necessarily *finite*

  Syntactic rules for composite formulae (wff)

- **Formal semantics**

  For <u>each</u> formal language, a *class* of structures (i.e. a class of *possible worlds*)

  In each possible world, <u>every</u> wff in the language is assigned a *value*

  In classical propositional logic, the set of values is the simplest: $\{1, 0\}$

- **Satisfaction, entailment**

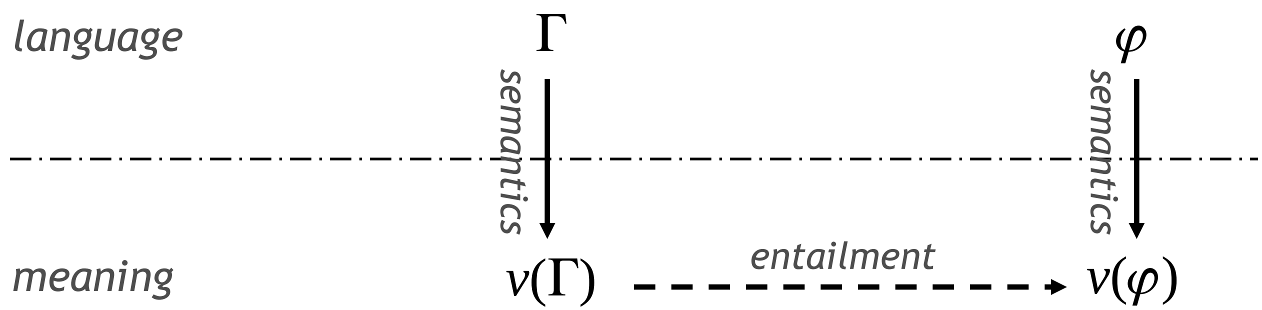
  A wff is *satisfied* in a possible world if it is <u>true</u> in that possible world

  In classical propositional logic, iff the wff has value 1 in that world
  (Caution: the definition of *satisfaction* will become definitely more complex with *first order logic*)

  Entailment is a <u>relation</u> between a set of wffs and a wff

  This relation holds when all possible worlds satisfying the set also satisfy the wff

What we have seen so far



Subtleties: object *language* and *metalanguage*

- The ***object language*** is L_P

  It is the tool that we plan to use

  It only contains the items just defined:

  $\boldsymbol{P}$, $\neg$, $\rightarrow$, $\wedge$, $\vee$, $\leftrightarrow$, (,), plus *syntactic rules* (wff)

- ***Metalanguage***

  Everything else we use to define the properties of the object language

  Small greek letters ($\alpha, \beta, \chi, \varphi, \psi$) will be used to denote a generic formula (wff)

  Capital greek letters (Γ, Δ, Σ) will be used to denote a set of formulae

  Satisfaction, logical consequence (see after): $\models$

  Derivability (see after): $\vdash$

  Symbols for "iff" and "if and only if" (also "iff"): $\Rightarrow$, $\Leftrightarrow$