2. SL and truth tables

2. SL and truth tables

a. Characteristic truth tables

Sentence letters and connectives

- Symbolization involves **sentence letters** like H and **connectives** ($\sim$, $\vee$, $\&$, $\supset$, $\equiv$)

Sentence letters and connectives

- Symbolization involves **sentence letters** like H and **connectives** ($\sim$, $\vee$, $\&$, $\supset$, $\equiv$)
- Recall that a **case** makes atomic sentences **true** or **false** (and never both).

Sentence letters and connectives

- Symbolization involves **sentence letters** like H and **connectives** ($\sim$, $\vee$, $\&$, $\supset$, $\equiv$)
- Recall that a **case** makes atomic sentences **true** or **false** (and never both).
- So if we can determine **truth conditions** for sentences involving connectives, we can assign **true** and **false** also to results of symbolization.

Sentence letters and connectives

- Symbolization involves **sentence letters** like H and **connectives** ($\sim$, $\vee$, $\&$, $\supset$, $\equiv$)
- Recall that a **case** makes atomic sentences **true** or **false** (and never both).
- So if we can determine **truth conditions** for sentences involving connectives, we can assign **true** and **false** also to results of symbolization.

When is $(H \& S)$ true?

Sentence letters and connectives

- Symbolization involves **sentence letters** like H and **connectives** ($\sim$, $\vee$, $\&$, $\supset$, $\equiv$)
- Recall that a **case** makes atomic sentences **true** or **false** (and never both).
- So if we can determine **truth conditions** for sentences involving connectives, we can assign **true** and **false** also to results of symbolization.

When is $(H \& S)$ true?

$(H \& S)$ is true if and only if H is true and S is also true.

Sentence letters and connectives

- Symbolization involves **sentence letters** like H and **connectives** ($\sim$, $\vee$, $\&$, $\supset$, $\equiv$)
- Recall that a **case** makes atomic sentences **true** or **false** (and never both).
- So if we can determine **truth conditions** for sentences involving connectives, we can assign **true** and **false** also to results of symbolization.

When is $(H \& S)$ true?

$(H \& S)$ is true if and only if H is true and S is also true.
Suppose a case makes H true and S false.

Sentence letters and connectives

- Symbolization involves **sentence letters** like H and **connectives** ($\sim$, $\vee$, &, $\supset$, $\equiv$)
- Recall that a **case** makes atomic sentences **true** or **false** (and never both).
- So if we can determine **truth conditions** for sentences involving connectives, we can assign **true** and **false** also to results of symbolization.

When is $(H \,\&\, S)$ true?

$(H \,\&\, S)$ is true if and only if H is true and S is also true.
Suppose a case makes H true and S false.
In that case, $(H \,\&\, S)$ would be .

Sentence letters and connectives

- Symbolization involves **sentence letters** like H and **connectives** ($\sim$, $\vee$, &, $\supset$, $\equiv$)
- Recall that a **case** makes atomic sentences **true** or **false** (and never both).
- So if we can determine **truth conditions** for sentences involving connectives, we can assign **true** and **false** also to results of symbolization.

When is (H & S) true?

(H & S) is true if and only if H is true and S is also true.
Suppose a case makes H true and S false.
In that case, (H & S) would be ???.

Sentence letters and connectives

- Symbolization involves **sentence letters** like H and **connectives** ($\sim$, $\vee$, $\&$, $\supset$, $\equiv$)
- Recall that a **case** makes atomic sentences **true** or **false** (and never both).
- So if we can determine **truth conditions** for sentences involving connectives, we can assign **true** and **false** also to results of symbolization.

When is $(H \& S)$ true?

$(H \& S)$ is true if and only if H is true and S is also true.
Suppose a case makes H true and S false.
In that case, $(H \& S)$ would be **false**.

Negation $\sim$

Definition

$\sim\mathscr{A}$ is true iff $\mathscr{A}$ is false.

Characteristic truth table:

$\mathscr{A}$	$\sim\mathscr{A}$
T	F
F	T

Conjunction &

Definition

$(\mathscr{A} \& \mathscr{B})$ is true iff $\mathscr{A}$ is true and $\mathscr{B}$ is true, and false otherwise.

Characteristic truth table:

$\mathscr{A}$	$\mathscr{B}$	$(\mathscr{A} \& \mathscr{B})$
T	T	T
T	F	F
F	T	F
F	F	F

Disjunction $\vee$

Definition

$(\mathscr{A} \vee \mathscr{B})$ is true iff $\mathscr{A}$ is true or $\mathscr{B}$ is true (or both), and false otherwise.

Characteristic truth table:

$\mathscr{A}$	$\mathscr{B}$	$(\mathscr{A} \vee \mathscr{B})$
T	T	T
T	F	T
F	T	T
F	F	F

A logic puzzle

Which card(s) do you have to turn over to make sure that:

> *If a card has an even number on one side, then it has a vowel on the other.*

E	K	3	4
(1)	(2)	(3)	(4)

The material conditional $\supset$

Definition

$(\mathscr{A} \supset \mathscr{B})$ is true iff $\mathscr{A}$ is false or $\mathscr{B}$ is true (or both), and false otherwise.

$\mathscr{A}$	$\mathscr{B}$	$(\mathscr{A} \supset \mathscr{B})$
T	T	T
T	F	F
F	T	T
F	F	T

The material conditional $\supset$

Definition

$(\mathscr{A} \supset \mathscr{B})$ is true iff $\mathscr{A}$ is false or $\mathscr{B}$ is true (or both), and false otherwise.

$\mathscr{A}$	$\mathscr{B}$	$(\mathscr{A} \supset \mathscr{B})$
T	T	T
T	F	F
F	T	T
F	F	T

Memorize 2nd row!: Conditional is false ONLY in case of a counter–example, i.e. case where antecedent is true but consequent is false

The material biconditional $\equiv$

Definition

$(\mathscr{A} \equiv \mathscr{B})$ is true iff $\mathscr{A}$ and $\mathscr{B}$ have the same truth value, and false otherwise.

$\mathscr{A}$	$\mathscr{B}$	$(\mathscr{A} \equiv \mathscr{B})$
T	T	T
T	F	F
F	T	F
F	F	T

2. SL and truth tables

b. Sentences of SL

Sentences of SL (*Well-formed Formulae* (WFFs))

Definition

1. Every sentence letter is a sentence (wff).
2. If $\mathscr{A}$ is a sentence, then $\sim\mathscr{A}$ is a sentence.
3. If $\mathscr{A}$ and $\mathscr{B}$ are sentences, then
   - $(\mathscr{A} \,\&\, \mathscr{B})$ is a sentence.
   - $(\mathscr{A} \vee \mathscr{B})$ is a sentence.
   - $(\mathscr{A} \supset \mathscr{B})$ is a sentence.
   - $(\mathscr{A} \equiv \mathscr{B})$ is a sentence.
4. Nothing else is a sentence.

The indicated connective is called the **main connective**.

Construction of sentences

- H is a sentence.

Construction of sentences

- H is a sentence.
- S is a sentence.

Construction of sentences

- H is a sentence.
- S is a sentence.
- $(H \vee S)$ is a sentence.

Construction of sentences

- H is a sentence.
- S is a sentence.
- $(H \vee S)$ is a sentence.
- $(H \mathbin{\&} S)$ is a sentence.

Construction of sentences

- H is a sentence.
- S is a sentence.
- $(H \vee S)$ is a sentence.
- $(H \mathbin{\&} S)$ is a sentence.
- $\sim(H \mathbin{\&} S)$ is a sentence.

Construction of sentences

- H is a sentence.
- S is a sentence.
- $(H \vee S)$ is a sentence.
- $(H \ \& \ S)$ is a sentence.
- $\sim(H \& S)$ is a sentence.
- $((H \vee S) \ \& \ \sim(H \& S))$ is a sentence.

Construction of sentences

- H is a sentence.
- S is a sentence.
- $(H \vee S)$ is a sentence.
- $(H \mathbin{\&} S)$ is a sentence.
- $\sim(H \mathbin{\&} S)$ is a sentence.
- $((H \vee S) \mathbin{\&} \sim(H \mathbin{\&} S))$ is a sentence.

(Main connective is **highlighted**.)

Examples of non-sentences (i.e. NOT well-formed formulae)

- $HikesMandy$

  single sentence letters
- $(H \sim S)$

  $\sim$ can't go between sentences
- $(H \& S \& C)$

  $\&$ combines only two sentences
- $(\sim H)$

  no parentheses around $\sim H$
- $(H \supset (S \& C)$

  missing closing parenthesis
- $H \vee S$

  missing parentheses
- $[H \supset (S \& C)]$

  only one kind of parentheses allowed: no brackets!

2. SL and truth tables

c. Truth value assignments (a.k.a. ‘valuations’)

Truth value assignment (a.k.a. 'valuation')

Definition

A **truth value assignment** (TVA) (a.k.a. **valuation**) is an assignment of **T** or **F** to each atomic sentence letter in a sentence or sentences.

Definition

The **truth value of a sentence** $\mathscr{S}$ on a valuation is:

1. if $\mathscr{S}$ is a sentence letter: the truth value assigned to it
2. if $\mathscr{S}$ is $\sim\mathscr{A}$: opposite of the truth value of $\mathscr{A}$
3. if $\mathscr{S}$ is $(\mathscr{A} * \mathscr{B})$: result of characteristic truth table of $*$ for truth values of $\mathscr{A}$ and $\mathscr{B}$.

Computing truth values

Truth-value assignment (TVA): H is **T**, S is **F**.

On this valuation:

- H is **T**.
- S is **F**.
- $(H \vee S)$ is **T** (because '**T** $\vee$ **F**' gives **T**).
- $(H \mathbin{\&} S)$ is **F** (because '**T** & **F**' gives **F**).
- $\sim(H \mathbin{\&} S)$ is **T** (because $\sim$**F** is **T**).
- $((H \vee S) \mathbin{\&} \sim(H \mathbin{\&} S))$ is **T** (because '**T** & **T**' gives **T**).

Computing truth values

H	S	$((H$	$\vee$	$S)$	$\&$	$\sim$	$(H$	$\&$	$S))$
T	**F**								

Computing truth values

H	S	$((H$	$\vee$	$S)$	$\&$	$\sim$	$(H$	$\&$	$S))$
T	**F**	**T**		**F**			**T**		**F**

- Copy truth values under atomic sentence letters.

Computing truth values

H	S	$((H$	$\vee$	$S)$	$\&$	$\sim$	$(H$	$\&$	$S))$
T	**F**	**T**		**F**			**T**		**F**
			$\uparrow$						

- Copy truth values under atomic sentence letters.
- Compute values of parts that combine sentence letters (working from 'inside out' i.e. from minor connectives toward the main connective)

Computing truth values

H	S	$((H$	$\vee$	$S)$	$\&$	$\sim$	$(H$	$\&$	$S))$
T	**F**	**T**	**T**	**F**			T		F
			$\uparrow$						

- Copy truth values under atomic sentence letters.
- Compute values of parts that combine sentence letters (working from 'inside out' i.e. from minor connectives toward the main connective)

Computing truth values

H	S	$((H$	$\vee$	$S)$	$\&$	$\sim$	$(H$	$\&$	$S))$
T	**F**	T	T	F			**T**		**F**
								$\uparrow$	

- Copy truth values under atomic sentence letters.
- Compute values of parts that combine sentence letters (working from ‘inside out’ i.e. from minor connectives toward the main connective)

Computing truth values

H	S	$((H$	$\vee$	$S)$	$\&$	$\sim$	$(H$	$\&$	$S))$
T	**F**	T	T	F			**T**	**F**	**F**
								$\uparrow$	

- Copy truth values under atomic sentence letters.
- Compute values of parts that combine sentence letters (working from 'inside out' i.e. from minor connectives toward the main connective)

Computing truth values

H	S	$((H$	$\vee$	$S)$	$\&$	$\sim$	$(H$	$\&$	$S))$
T	**F**	T	T	F			T	**F**	F
						$\uparrow$			

- Copy truth values under atomic sentence letters.
- Compute values of parts that combine sentence letters (working from 'inside out' i.e. from minor connectives toward the main connective)
- Use computed values for larger parts.

Computing truth values

H	S	$((H$	$\vee$	$S)$	$\&$	$\sim$	$(H$	$\&$	$S))$
T	**F**	T	T	F		**T**	T	**F**	F
						$\uparrow$			

- Copy truth values under atomic sentence letters.
- Compute values of parts that combine sentence letters (working from ‘inside out’ i.e. from minor connectives toward the main connective)
- Use computed values for larger parts.

Computing truth values

H	S	$((H$	$\vee$	$S)$	$\&$	$\sim$	$(H$	$\&$	$S))$
T	F	T	T	F		T	T	F	F
					$\uparrow$				

- Copy truth values under atomic sentence letters.
- Compute values of parts that combine sentence letters (working from 'inside out' i.e. from minor connectives toward the main connective)
- Use computed values for larger parts.

Computing truth values

H	S	$((H$	$\vee$	$S)$	$\&$	$\sim$	$(H$	$\&$	$S))$
T	**F**	T	**T**	F	**T**	**T**	T	F	F
					$\uparrow$				

- Copy truth values under atomic sentence letters.
- Compute values of parts that combine sentence letters (working from ‘inside out’ i.e. from minor connectives toward the main connective)
- Use computed values for larger parts.
- Done row when you have the value under the main connective.

Computing truth values

H	S	$((H$	$\vee$	$S)$	$\&$	$\sim$	$(H$	$\&$	$S))$
T	F	T	T	F	T	T	T	F	F

- Copy truth values under atomic sentence letters.
- Compute values of parts that combine sentence letters (working from 'inside out' i.e. from minor connectives toward the main connective)
- Use computed values for larger parts.
- Done row when you have the value under the main connective.
- Complete this process for each row (each TVA)

2. SL and truth tables

d. Validity and truth tables

Validity

- Recall: an argument is (deductively) **valid** if there is no **case** where all premises are true and the conclusion is false.

Validity

- Recall: an argument is (deductively) **valid** if there is no **case** where all premises are true and the conclusion is false.
- In a **case**, every **atomic sentence** is either true or false (but not both!).

Validity

- Recall: an argument is (deductively) **valid** if there is no **case** where all premises are true and the conclusion is false.
- In a **case**, every **atomic sentence** is either true or false (but not both!).
- In SL, **valuations** make every **atomic sentence letter** true or false (and not both).

Validity

- Recall: an argument is (deductively) **valid** if there is no **case** where all premises are true and the conclusion is false.
- In a **case**, every **atomic sentence** is either true or false (but not both!).
- In SL, **valuations** make every **atomic sentence letter** true or false (and not both).
- Also: every valuation makes every **wff** (i.e. ‘sentence’) true or false (but not both!), and we can compute the truth value of any well–formed formula (wff).

Validity in SL

Definition

An argument is **valid in SL** if there is **no** valuation in which all premises are **T** and the conclusion is **F**.

An argument is **invalid in SL** if there is **at least one** valuation in which all premises are **T** and the conclusion is **F**.

Validity in SL

Definition

An argument is **valid in SL** if there is **no** valuation in which all premises are **T** and the conclusion is **F**.

An argument is **invalid in SL** if there is **at least one** valuation in which all premises are **T** and the conclusion is **F**.

Don't forget two *vacuous* cases of valid arguments:
(i) inconsistent premises (never all true on a valuation)
(ii) conclusion is a tautology (true on every valuation)

Disjunctive syllogism

$$H \vee S$$
$$\sim S$$
$$\therefore\ H$$

H	S	$(H \vee S)$	$\sim S$	H

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$$H \vee S$$
$$\sim S$$
$$\therefore H$$

H	S	$(H \vee S)$	$\sim S$	H
T	**T**			
T	**F**			
F	**T**			
F	**F**			

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$$\begin{array}{l} H \vee S \\ \sim S \\ \therefore\ H \end{array}$$

H	S	$(H \vee S)$		$\sim S$		H
T	T	T	T			
T	F	T	F			
F	T	F	T			
F	F	F	F			

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$$\begin{array}{rl} & H \vee S \\ & \sim S \\ \therefore & H \end{array}$$

H	S	$(H$	$\vee$	$S)$	$\sim$	S	H
T	**T**	**T**	**T**	**T**			
T	**F**	**T**	**T**	**F**			
F	**T**	**F**	**T**	**T**			
F	**F**	**F**	**F**	**F**			

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$H \vee S$
$\sim S$
$\therefore H$

H	S	$(H$	$\vee$	$S)$	$\sim$	S	H
T	T	T	T	T		T	
T	F	T	T	F		F	
F	T	F	T	T		T	
F	F	F	F	F		F	

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$$H \vee S$$
$$\sim S$$
$$\therefore\ H$$

H	S	$(H$	$\vee$	$S)$	$\sim$	S	H
T	**T**	T	T	T	**F**	**T**	
T	**F**	T	T	F	**T**	**F**	
F	**T**	F	T	T	**F**	**T**	
F	**F**	F	F	F	**T**	**F**	

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$$H \vee S$$
$$\sim S$$
$$\therefore H$$

H	S	$(H$	$\vee$	$S)$	$\sim$	S	H
T	**T**	T	T	T	F	T	**T**
T	**F**	T	T	F	T	F	**T**
F	**T**	F	T	T	F	T	**F**
F	**F**	F	F	F	T	F	**F**

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$$H \vee S$$
$$\sim S$$
$$\therefore\ H$$

H	S	$(H$	$\vee$	$S)$	$\sim$	S	H
T	**T**	T	**T**	T	**F**	T	**T**
T	**F**	T	**T**	F	**T**	F	**T**
F	**T**	F	**T**	T	**F**	T	**F**
F	**F**	F	**F**	F	**T**	F	**F**

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$H \vee S$
$\sim S$
$\therefore\ H$

H	S	$(H$	$\vee$	$S)$	$\sim$	S	H	
T	**T**	T	**T**	T	**F**	T	**T**	$\leftarrow$
T	**F**	T	**T**	F	**T**	F	**T**	
F	**T**	F	**T**	T	**F**	T	**F**	
F	**F**	F	**F**	F	**T**	F	**F**	

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$$H \vee S$$
$$\sim S$$
$$\therefore\ H$$

H	S	$(H$	$\vee$	$S)$	$\sim$	S	H	
T	**T**	T	**T**	T	**F**	T	**T**	✓
T	**F**	T	**T**	F	**T**	F	**T**	←
F	**T**	F	**T**	T	**F**	T	**F**	
F	**F**	F	**F**	F	**T**	F	**F**	

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$$H \vee S$$
$$\sim S$$
$$\therefore\ H$$

H	S	$(H$	$\vee$	$S)$	$\sim$	S	H	
T	**T**	T	**T**	T	**F**	T	**T**	✓
T	**F**	T	**T**	F	**T**	F	**T**	✓
F	**T**	F	**T**	T	**F**	T	**F**	←
F	**F**	F	**F**	F	**T**	F	**F**	

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$H \vee S$
$\sim S$
$\therefore\ H$

H	S	$(H$	$\vee$	$S)$	$\sim$	S	H	
T	**T**	T	**T**	T	**F**	T	**T**	✓
T	**F**	T	**T**	F	**T**	F	**T**	✓
F	**T**	F	**T**	T	**F**	T	**F**	✓
F	**F**	F	**F**	F	**T**	F	**F**	←

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

Disjunctive syllogism

$H \vee S$
$\sim S$
$\therefore\ H$

H	S	$(H$	$\vee$	$S)$	$\sim$	S	H	
T	**T**	T	**T**	T	**F**	T	**T**	✓
T	**F**	T	**T**	F	**T**	F	**T**	✓
F	**T**	F	**T**	T	**F**	T	**F**	✓
F	**F**	F	**F**	F	**T**	F	**F**	✓

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- All valuations check out: valid.

An invalid argument

$$\begin{array}{rl} & H \lor S \\ & H \\ \therefore & \sim S \end{array}$$

H	S	$(H \lor S)$	H	$\sim S$

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- Find a valuation with all premises **T** and conclusion **F**: invalid.

An invalid argument

$H \vee S$
H
$\therefore \sim S$

H	S	$(H \vee S)$	H	$\sim S$
T	**T**			
T	**F**			
F	**T**			
F	**F**			

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- Find a valuation with all premises **T** and conclusion **F**: invalid.

An invalid argument

$H \vee S$
H
$\therefore \sim S$

H	S	$(H$	$\vee$	$S)$	H	$\sim$	S
T	**T**	**T**	**T**	**T**	**T**	**F**	**T**
T	**F**	**T**	**T**	**F**	**T**	**T**	**F**
F	**T**	**F**	**T**	**T**	**F**	**F**	**T**
F	**F**	**F**	**F**	**F**	**F**	**T**	**F**

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- Find a valuation with all premises **T** and conclusion **F**: invalid.

An invalid argument

$H \vee S$
H
$\therefore \sim S$

H	S	$(H$	$\vee$	$S)$	H	$\sim$	S
T	**T**	T	**T**	T	**T**	**F**	T
T	**F**	T	**T**	F	**T**	**T**	F
F	**T**	F	**T**	T	**F**	**F**	T
F	**F**	F	**F**	F	**F**	**T**	F

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- Find a valuation with all premises **T** and conclusion **F**: invalid.

An invalid argument

$H \vee S$

H

$\therefore \sim S$

H	S	$(H \vee S)$	H	$\sim S$	
T	T	T T T	T	F T	$\leftarrow$
T	F	T T F	T	T F	
F	T	F T T	F	F T	
F	F	F F F	F	T F	

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- Find a valuation with all premises **T** and conclusion **F**: invalid.

An invalid argument

$$H \lor S$$
$$H$$
$$\therefore \sim S$$

H	S	$(H$	$\lor$	$S)$	H	$\sim$	S	
T	**T**	T	**T**	T	**T**	**F**	T	✗
T	**F**	T	**T**	F	**T**	**T**	F	
F	**T**	F	**T**	T	**F**	**F**	T	
F	**F**	F	**F**	F	**F**	**T**	F	

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- Find a valuation with all premises **T** and conclusion **F**: invalid.

An invalid argument

$$\begin{array}{l} H \lor S \\ H \\ \therefore \sim S \end{array}$$

H	S	$(H$	$\lor$	$S)$	H	$\sim$	S	
T	**T**	T	**T**	T	**T**	**F**	T	✗
T	**F**	T	**T**	F	**T**	**T**	F	←
F	**T**	F	**T**	T	**F**	**F**	T	
F	**F**	F	**F**	F	**F**	**T**	F	

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- Find a valuation with all premises **T** and conclusion **F**: invalid.

An invalid argument

$$H \vee S$$
$$H$$
$$\therefore \sim S$$

H	S	$(H$	$\vee$	$S)$	H	$\sim$	S	
T	**T**	T	**T**	T	**T**	**F**	T	✗
T	**F**	T	**T**	F	**T**	**T**	F	✓
F	**T**	F	**T**	T	**F**	**F**	T	←
F	**F**	F	**F**	F	**F**	**T**	F	

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- Find a valuation with all premises **T** and conclusion **F**: invalid.

An invalid argument

$H \vee S$
H
$\therefore \sim S$

H	S	$(H$	$\vee$	$S)$	H	$\sim$	S	
T	**T**	T	**T**	T	**T**	**F**	T	✗
T	**F**	T	**T**	F	**T**	**T**	F	✓
F	**T**	F	**T**	T	**F**	**F**	T	✓
F	**F**	F	**F**	F	**F**	**T**	F	$\leftarrow$

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- Find a valuation with all premises **T** and conclusion **F**: invalid.

An invalid argument

$H \vee S$
H
$\therefore\ \sim S$

H	S	$(H$	$\vee$	$S)$	H	$\sim$	S	
T	T	T	T	T	T	F	T	✗
T	F	T	T	F	T	T	F	✓
F	T	F	T	T	F	F	T	✓
F	F	F	F	F	F	T	F	✓

- List all valuations for H, S.
- Compute truth values of premises, conclusion.
- Check each valuation: one premise **F**, or conclusion **T**?
- Find a valuation with all premises **T** and conclusion **F**: invalid.

2. SL and truth tables

e. Large truth tables

Large truth tables

- For arguments with n sentence letters, there are 2^n possible valuations

Large truth tables

- For arguments with n sentence letters, there are 2^n possible valuations
  - A single letter A can be be **T** or **F**: $2^1 = 2$ valuations.

Large truth tables

- For arguments with n sentence letters, there are 2^n possible valuations
  - A single letter A can be **T** or **F**: $2^1 = 2$ valuations.
  - For two letters A, B: B can be **T** or **F** for every possible valuation (2) of A: $2 \times 2 = 2^2 = 4$ valuations

Large truth tables

- For arguments with n sentence letters, there are 2^n possible valuations
  - A single letter A can be **T** or **F**: $2^1 = 2$ valuations.
  - For two letters A, B: B can be **T** or **F** for every possible valuation (2) of A: $2 \times 2 = 2^2 = 4$ valuations
  - For three letters A, B, C: C can be **T** or **F** for every possible valuation (4) of A and B: $2 \times 4 = 2^3 = 8$ valuations

Large truth tables

- For arguments with n sentence letters, there are 2^n possible valuations
  - A single letter A can be **T** or **F**: $2^1 = 2$ valuations.
  - For two letters A, B: B can be **T** or **F** for every possible valuation (2) of A: $2 \times 2 = 2^2 = 4$ valuations
  - For three letters A, B, C: C can be **T** or **F** for every possible valuation (4) of A and B: $2 \times 4 = 2^3 = 8$ valuations
  - Etc.

Large truth tables

- For arguments with n sentence letters, there are 2^n possible valuations
  - A single letter A can be **T** or **F**: $2^1 = 2$ valuations.
  - For two letters A, B: B can be **T** or **F** for every possible valuation (2) of A: $2 \times 2 = 2^2 = 4$ valuations
  - For three letters A, B, C: C can be **T** or **F** for every possible valuation (4) of A and B: $2 \times 4 = 2^3 = 8$ valuations
  - Etc.
- In the ith reference column, alternate **T** and **F** every 2^{n-i} lines

A complex truth table

3 sentence letters A, C, E: $2^3 = 8$ lines

	A	C	E	$\ldots$
1				$\ldots$
2				$\ldots$
3				$\ldots$
4				$\ldots$
5				$\ldots$
6				$\ldots$
7				$\ldots$
8				$\ldots$

A complex truth table

3 sentence letters A, C, E: $2^3 = 8$ lines

	A	C	E	...
1				...
2				...
3				...
4				...
5				...
6				...
7				...
8				...

$\uparrow$

alternate every ...

4

rows

A complex truth table

3 sentence letters A, C, E: $2^3 = 8$ lines

	A	C	E	...
1	**T**			...
2	**T**			...
3	**T**			...
4	**T**			...
5	**F**			...
6	**F**			...
7	**F**			...
8	**F**			...

↑

alternate every ...

2

rows

A complex truth table

3 sentence letters A, C, E: $2^3 = 8$ lines

	A	C	E	...
1	T	T		...
2	T	T		...
3	T	F		...
4	T	F		...
5	F	T		...
6	F	T		...
7	F	F		...
8	F	F		...

$\uparrow$

alternate every ...

1

rows

A complex truth table

3 sentence letters A, C, E: $2^3 = 8$ lines

	A	C	E	...
1	**T**	**T**	**T**	...
2	**T**	**T**	**F**	...
3	**T**	**F**	**T**	...
4	**T**	**F**	**F**	...
5	**F**	**T**	**T**	...
6	**F**	**T**	**F**	...
7	**F**	**F**	**T**	...
8	**F**	**F**	**F**	...

alternate every ...

1

rows

Example (simplified)

Sarah lives in Chicago or Erie.
Amir lives in Chicago unless he enjoys hiking.
If Amir lives in Chicago, Sarah doesn't.
Amir doesn't enjoy hiking.
∴ Sarah lives in Erie.

$$C \vee E$$
$$A \vee M$$
$$A \supset {\sim}C$$
$${\sim}M$$
$$\therefore E$$

A	C	E	M	$C \vee E$	$A \vee M$	$A \supset \sim C$	$\sim M$	E
T	T	T	T					
T	T	T	F					
T	T	F	T					
T	T	F	F					
T	F	T	T					
T	F	T	F					
T	F	F	T					
T	F	F	F					
F	T	T	T					
F	T	T	F					
F	T	F	T					
F	T	F	F					
F	F	T	T					
F	F	T	F					
F	F	F	T					
F	F	F	F					

A	C	E	M	$C \vee E$		$A \vee M$		$A \supset \sim C$		$\sim M$	E
T	T	T	T	T	T	T	T	T	T	T	T
T	T	T	F	T	T	T	F	T	T	F	T
T	T	F	T	T	F	T	T	T	T	T	F
T	T	F	F	T	F	T	F	T	T	F	F
T	F	T	T	F	T	T	T	T	F	T	T
T	F	T	F	F	T	T	F	T	F	F	T
T	F	F	T	F	F	T	T	T	F	T	F
T	F	F	F	F	F	T	F	T	F	F	F
F	T	T	T	T	T	F	T	F	T	T	T
F	T	T	F	T	T	F	F	F	T	F	T
F	T	F	T	T	F	F	T	F	T	T	F
F	T	F	F	T	F	F	F	F	T	F	F
F	F	T	T	F	T	F	T	F	F	T	T
F	F	T	F	F	T	F	F	F	F	F	T
F	F	F	T	F	F	F	T	F	F	T	F
F	F	F	F	F	F	F	F	F	F	F	F

A	C	E	M	$C \vee E$			$A \vee M$			$A \supset \sim C$				$\sim M$		E
T	T	T	T	T	T	T	T	T	T	T	F	F	T	F	T	T
T	T	T	F	T	T	T	T	T	F	T	F	F	T	T	F	T
T	T	F	T	T	T	F	T	T	T	T	F	F	T	F	T	F
T	T	F	F	T	T	F	T	T	F	T	F	F	T	T	F	F
T	F	T	T	F	T	T	T	T	T	T	T	T	F	F	T	T
T	F	T	F	F	T	T	T	T	F	T	T	T	F	T	F	T
T	F	F	T	F	F	F	T	T	T	T	T	T	F	F	T	F
T	F	F	F	F	F	F	T	T	F	T	T	T	F	T	F	F
F	T	T	T	T	T	T	F	T	T	F	T	F	T	F	T	T
F	T	T	F	T	T	T	F	F	F	F	T	F	T	T	F	T
F	T	F	T	T	T	F	F	T	T	F	T	F	T	F	T	F
F	T	F	F	T	T	F	F	F	F	F	T	F	T	T	F	F
F	F	T	T	F	T	T	F	T	T	F	T	T	F	F	T	T
F	F	T	F	F	T	T	F	F	F	F	T	T	F	T	F	T
F	F	F	T	F	F	F	F	T	T	F	T	T	F	F	T	F
F	F	F	F	F	F	F	F	F	F	F	T	T	F	T	F	F

A	C	E	M	C	$\vee$	E	A	$\vee$	M	A	$\supset$	$\sim$	C	$\sim$	M	E
T	T	T	T	T	T	T	T	T	T	T	F	F	T	F	T	T
T	T	T	F	T	T	T	T	T	F	T	F	F	T	T	F	T
T	T	F	T	T	T	F	T	T	T	T	F	F	T	F	T	F
T	T	F	F	T	T	F	T	T	F	T	F	F	T	T	F	F
T	F	T	T	F	T	T	T	T	T	T	T	T	F	F	T	T
T	F	T	F	F	T	T	T	T	F	T	T	T	F	T	F	T
T	F	F	T	F	F	F	T	T	T	T	T	T	F	F	T	F
T	F	F	F	F	F	F	T	T	F	T	T	T	F	T	F	F
F	T	T	T	T	T	T	F	T	T	F	T	F	T	F	T	T
F	T	T	F	T	T	T	F	F	F	F	T	F	T	T	F	T
F	T	F	T	T	T	F	F	T	T	F	T	F	T	F	T	F
F	T	F	F	T	T	F	F	F	F	F	T	F	T	T	F	F
F	F	T	T	F	T	T	F	T	T	F	T	T	F	F	T	T
F	F	T	F	F	T	T	F	F	F	F	T	T	F	T	F	T
F	F	F	T	F	F	F	F	T	T	F	T	T	F	F	T	F
F	F	F	F	F	F	F	F	F	F	F	T	T	F	T	F	F

A	C	E	M	$C \vee E$	$A \vee M$	$A \supset \sim C$	$\sim M$	E
T	T	T	T	T T T	T T T	T F F T	F T	T
T	T	T	F	T T T	T T F	T F F T	T F	T
T	T	F	T	T T F	T T T	T F F T	F T	F
T	T	F	F	T T F	T T F	T F F T	T F	F
T	F	T	T	F T T	T T T	T T T F	F T	T
T	F	T	F	F T T	T T F	T T T F	T F	T
T	F	F	T	F F F	T T T	T T T F	F T	F
T	F	F	F	F F F	T T F	T T T F	T F	F
F	T	T	T	T T T	F T T	F T F T	F T	T
F	T	T	F	T T T	F F F	F T F T	T F	T
F	T	F	T	T T F	F T T	F T F T	F T	F
F	T	F	F	T T F	F F F	F T F T	T F	F
F	F	T	T	F T T	F T T	F T T F	F T	T
F	F	T	F	F T T	F F F	F T T F	T F	T
F	F	F	T	F F F	F T T	F T T F	F T	F
F	F	F	F	F F F	F F F	F T T F	T F	F

Every valuation makes at least one premise false, or makes the conclusion true: the argument is valid.

2. SL and truth tables

f. Entailment, equivalence, tautologies

Validity of arguments

Definition

An argument is **valid in SL** iff every truth-value assignment either makes one or more of the premises false or it makes the conclusion true.

Validity of arguments

Definition

An argument is **valid in SL** iff every truth-value assignment either makes one or more of the premises false or it makes the conclusion true.
(i.e. there is no TVA where the premises are true but the conclusion is false).

Validity of arguments

Definition

An argument is **valid in SL** iff every truth-value assignment either makes one or more of the premises false or it makes the conclusion true.
(i.e. there is no TVA where the premises are true but the conclusion is false).

An argument is **invalid in SL** iff at least one TVA makes all the premises true and it makes the conclusion false.

Entailment

Definition

Sentences $\mathscr{A}_1, \ldots, \mathscr{A}_n$ **entail** a sentence $\mathscr{B}$ iff every TVA either makes at least one of $\mathscr{A}_1, \ldots, \mathscr{A}_n$ false or makes $\mathscr{B}$ true.

(i.e. for any valuation where the premises are all true, the conclusion is true)

Entailment

Definition

Sentences $\mathscr{A}_1, \ldots, \mathscr{A}_n$ **entail** a sentence $\mathscr{B}$ iff every TVA either makes at least one of $\mathscr{A}_1, \ldots, \mathscr{A}_n$ false or makes $\mathscr{B}$ true.

(i.e. for any valuation where the premises are all true, the conclusion is true)

In that case we write $\mathscr{A}_1, \ldots, \mathscr{A}_n \vDash \mathscr{B}$.

Entailment

Definition

Sentences $\mathscr{A}_1, \ldots, \mathscr{A}_n$ **entail** a sentence $\mathscr{B}$ iff every TVA either makes at least one of $\mathscr{A}_1, \ldots, \mathscr{A}_n$ false or makes $\mathscr{B}$ true.

(i.e. for any valuation where the premises are all true, the conclusion is true)

In that case we write $\mathscr{A}_1, \ldots, \mathscr{A}_n \vDash \mathscr{B}$.

The symbol '$\vDash$' is called a 'double turnstile'

Entailment

Definition

Sentences $\mathscr{A}_1, \ldots, \mathscr{A}_n$ **entail** a sentence $\mathscr{B}$ iff every TVA either makes at least one of $\mathscr{A}_1, \ldots, \mathscr{A}_n$ false or makes $\mathscr{B}$ true.

(i.e. for any valuation where the premises are all true, the conclusion is true)

In that case we write $\mathscr{A}_1, \ldots, \mathscr{A}_n \vDash \mathscr{B}$.

The symbol '$\vDash$' is called a 'double turnstile'

Note the following relationship between entailment and validity:

$\mathscr{A}_1, \ldots, \mathscr{A}_n \vDash \mathscr{B}$ iff the argument '$\mathscr{A}_1, \ldots, \mathscr{A}_n \therefore \mathscr{B}$' is valid.

Entailment

Does $\sim(\sim A \vee \sim B), A \supset \sim C \vDash A \supset (B \supset C)$?

Entailment

Does $\sim(\sim A \vee \sim B), A \supset \sim C \vDash A \supset (B \supset C)$?

Note that we are asking whether two sentences of SL entail a third (i.e. do the first two provide a deductively valid argument for the conclusion $A \supset (B \supset C)$?)

Entailment

A	B	C	$\sim$	$(\sim$	A	$\vee$	$\sim$	$B)$	A	$\supset$	$\sim$	C	A	$\supset$	$(B$	$\supset$	$C)$
T	T	T	T	F	T	F	F	T	T	F	F	T	T	T	T	T	T
T	T	F	T	F	T	F	F	T	T	T	T	F	T	F	T	F	F
T	F	T	F	F	T	T	T	F	T	F	F	T	T	T	F	T	T
T	F	F	F	F	T	T	T	F	T	T	T	F	T	T	F	T	F
F	T	T	F	T	F	T	F	T	F	T	F	T	F	T	T	T	T
F	T	F	F	T	F	T	F	T	F	T	T	F	F	T	T	F	F
F	F	T	F	T	F	T	T	F	F	T	F	T	F	T	F	T	T
F	F	F	F	T	F	T	T	F	F	T	T	F	F	T	F	T	F

Entailment

A	B	C	$\sim$	(	$\sim$	A	$\vee$	$\sim$	B	)	A	$\supset$	$\sim$	C	A	$\supset$	(	B	$\supset$	C	)	
T	T	T	T		F	T	F	F	T		T	F	F	T	T	T		T	T	T		
T	T	F	T		F	T	F	F	T		T	T	T	F	T	F		T	F	F		$\leftarrow$
T	F	T	F		F	T	T	T	F		T	F	F	T	T	T		F	T	T		
T	F	F	F		F	T	T	T	F		T	T	T	F	T	T		F	T	F		
F	T	T	F		T	F	T	F	T		F	T	F	T	F	T		T	T	T		
F	T	F	F		T	F	T	F	T		F	T	T	F	F	T		T	F	F		
F	F	T	F		T	F	T	T	F		F	T	F	T	F	T		F	T	T		
F	F	F	F		T	F	T	T	F		F	T	T	F	F	T		F	T	F		

Tautologies

Definition

A sentence $\mathscr{A}$ is a **tautology** iff it is true on every valuation.

P	P	$\supset$	P
T	T	T	T
F	F	T	F

Contradictions

Definition

A sentence $\mathscr{A}$ is a **contradiction** iff it is false on every valuation.

P	P	$\&$	$\sim$	P
T	T	F	F	T
F	F	F	T	F

Logically equivalent sentences

Definition

Two sentences $\mathscr{A}$ and $\mathscr{B}$ are **equivalent in SL** iff every TVA either makes both $\mathscr{A}$ and $\mathscr{B}$ true or it makes both $\mathscr{A}$ and $\mathscr{B}$ false.

In other words: $\mathscr{A}$ and $\mathscr{B}$ agree in truth value, for every valuation.

Logically equivalent sentences

Definition

Two sentences $\mathscr{A}$ and $\mathscr{B}$ are **equivalent in SL** iff every TVA either makes both $\mathscr{A}$ and $\mathscr{B}$ true or it makes both $\mathscr{A}$ and $\mathscr{B}$ false.

In other words: $\mathscr{A}$ and $\mathscr{B}$ agree in truth value, for every valuation.

Interesting case of equivalence: $\mathscr{A}$ and $\mathscr{B}$ comprise the same atomic sentence letters

Logically equivalent sentences

Definition

Two sentences $\mathscr{A}$ and $\mathscr{B}$ are **equivalent in SL** iff every TVA either makes both $\mathscr{A}$ and $\mathscr{B}$ true or it makes both $\mathscr{A}$ and $\mathscr{B}$ false.

In other words: $\mathscr{A}$ and $\mathscr{B}$ agree in truth value, for every valuation.

Interesting case of equivalence: $\mathscr{A}$ and $\mathscr{B}$ comprise the same atomic sentence letters

Uninteresting cases: (1) all tautologies are equivalent;
(2) all contradictions are equivalent

Logically equivalent sentences

Definition

Two sentences $\mathscr{A}$ and $\mathscr{B}$ are **equivalent in SL** iff every TVA either makes both $\mathscr{A}$ and $\mathscr{B}$ true or it makes both $\mathscr{A}$ and $\mathscr{B}$ false.

In other words: $\mathscr{A}$ and $\mathscr{B}$ agree in truth value, for every valuation.

Interesting case of equivalence: $\mathscr{A}$ and $\mathscr{B}$ comprise the same atomic sentence letters

Uninteresting cases: (1) all tautologies are equivalent;
(2) all contradictions are equivalent

Philosophy question: What might we take this to indicate about *the meaning* of tautologies and contradictions?

Equivalent sentences

A	B	$\sim$	A	$\vee$	B	A	$\supset$	B
T	**T**		**T**		**T**	**T**		**T**
T	**F**		**T**		**F**	**T**		**F**
F	**T**		**F**		**T**	**F**		**T**
F	**F**		**F**		**F**	**F**		**F**

Equivalent sentences

A	B	$\sim$	A	$\vee$	B	A	$\supset$	B
T	T	F	T	T	T	T	T	T
T	F	F	T	F	F	T	F	F
F	T	T	F	T	T	F	T	T
F	F	T	F	T	F	F	T	F

Equivalent sentences

A	B	$\sim$	A	$\vee$	B	A	$\supset$	B
T	T	F	T	T	T	T	T	T
T	F	F	T	F	F	T	F	F
F	T	T	F	T	T	F	T	T
F	F	T	F	T	F	F	T	F

Equivalent sentences

A	B	$\sim$	A	$\vee$	B	A	$\supset$	B
T	**T**	F	T	**T**	T	T	T	T
T	**F**	F	T	**F**	F	T	F	F
F	**T**	T	F	**T**	T	F	T	T
F	**F**	T	F	**T**	F	F	T	F

Note how $(\sim A \vee B)$ wears the truth-conditions of $A \supset B$ "on its sleeves"

Equivalence and entailment

Fact

If $\mathscr{A}$ and $\mathscr{B}$ are equivalent, then $\mathscr{A} \vDash \mathscr{B}$ (likewise, $\mathscr{B} \vDash \mathscr{A}$).

Equivalence and entailment

Fact

If $\mathscr{A}$ and $\mathscr{B}$ are equivalent, then $\mathscr{A} \vDash \mathscr{B}$ (likewise, $\mathscr{B} \vDash \mathscr{A}$).

Equivalence and entailment

Fact

If $\mathscr{A}$ and $\mathscr{B}$ are equivalent, then $\mathscr{A} \vDash \mathscr{B}$ (likewise, $\mathscr{B} \vDash \mathscr{A}$).

Proof

Equivalence and entailment

Fact

If $\mathscr{A}$ and $\mathscr{B}$ are equivalent, then $\mathscr{A} \vDash \mathscr{B}$ (likewise, $\mathscr{B} \vDash \mathscr{A}$).

Proof

- Look at any valuation: it makes $\mathscr{A}$ true or false.

Equivalence and entailment

Fact

If $\mathscr{A}$ and $\mathscr{B}$ are equivalent, then $\mathscr{A} \vDash \mathscr{B}$ (likewise, $\mathscr{B} \vDash \mathscr{A}$).

Proof

- Look at any valuation: it makes $\mathscr{A}$ true or false.
- If $\mathscr{A}$ is false, the valuation is not a counterexample.

Equivalence and entailment

Fact

If $\mathscr{A}$ and $\mathscr{B}$ are equivalent, then $\mathscr{A} \vDash \mathscr{B}$ (likewise, $\mathscr{B} \vDash \mathscr{A}$).

Proof

- Look at any valuation: it makes $\mathscr{A}$ true or false.
- If $\mathscr{A}$ is false, the valuation is not a counterexample.
- If $\mathscr{A}$ is true, $\mathscr{B}$ is also true (since $\mathscr{A}$ and $\mathscr{B}$ agree in truth value on every valuation).

Equivalence and entailment

Fact

If $\mathscr{A}$ and $\mathscr{B}$ are equivalent, then $\mathscr{A} \vDash \mathscr{B}$ (likewise, $\mathscr{B} \vDash \mathscr{A}$).

Proof

- Look at any valuation: it makes $\mathscr{A}$ true or false.
- If $\mathscr{A}$ is false, the valuation is not a counterexample.
- If $\mathscr{A}$ is true, $\mathscr{B}$ is also true (since $\mathscr{A}$ and $\mathscr{B}$ agree in truth value on every valuation).
- So if $\mathscr{A}$ is true, the valuation is also not a counterexample.

Equivalence and entailment

Fact

If $\mathscr{A}$ and $\mathscr{B}$ are equivalent, then $\mathscr{A} \vDash \mathscr{B}$ (likewise, $\mathscr{B} \vDash \mathscr{A}$).

Proof

- Look at any valuation: it makes $\mathscr{A}$ true or false.
- If $\mathscr{A}$ is false, the valuation is not a counterexample.
- If $\mathscr{A}$ is true, $\mathscr{B}$ is also true (since $\mathscr{A}$ and $\mathscr{B}$ agree in truth value on every valuation).
- So if $\mathscr{A}$ is true, the valuation is also not a counterexample.
- So, no valuation can be a counterexample to $\mathscr{A} \vDash \mathscr{B}$.

Two Questions about Entailment

Let '$\Gamma \not\models \Delta$' mean that the (set of) sentence(s) Γ does not semantically entail Δ, i.e. an argument from Γ to Δ is invalid.

1. True or False? If $\Phi \models \Psi$, then $\sim\Phi \not\models \Psi$

Two Questions about Entailment

Let ‘$\Gamma \not\models \Delta$’ mean that the (set of) sentence(s) Γ does not semantically entail Δ, i.e. an argument from Γ to Δ is invalid.

1. True or False? If $\Phi \models \Psi$, then $\sim\Phi \not\models \Psi$

2. True or False? If $\Gamma \models \Phi$ and $\Delta, \Phi \models \Psi$, then $\Gamma, \Delta \models \Psi$?

2. SL and truth tables

g. Consistency

Consistency (a.k.a joint satisfiability)

Definition

Sentences $\mathscr{A}_1, \ldots, \mathscr{A}_n$ are **consistent** (i.e. 'satisfiable') in SL if there is at least one TVA that makes all of them true.

If they are not satisfiable, we say that they are **inconsistent** (a.k.a 'jointly unsatisfiable').

Consistency (a.k.a joint satisfiability)

Definition

Sentences $\mathscr{A}_1, \ldots, \mathscr{A}_n$ are **consistent** (i.e. 'satisfiable') in SL if there is at least one TVA that makes all of them true.

If they are not satisfiable, we say that they are **inconsistent** (a.k.a 'jointly unsatisfiable').

$A \vee B$, $\sim A$, B are consistent (a.k.a 'satisfiable').

Consistency (a.k.a joint satisfiability)

Definition

Sentences $\mathscr{A}_1, \ldots, \mathscr{A}_n$ are **consistent** (i.e. 'satisfiable') in SL if there is at least one TVA that makes all of them true.

If they are not satisfiable, we say that they are **inconsistent** (a.k.a 'jointly unsatisfiable').

$A \vee B$, $\sim A$, B are consistent (a.k.a 'satisfiable').

$A \vee B$, $\sim A$, $\sim B$ are inconsistent (a.k.a 'unsatisfiable').

Inconsistency and validity

- Any argument with inconsistent premises is valid.

Inconsistency and validity

- Any argument with inconsistent premises is valid.
  - If premises are inconsistent, no valuation makes them all true.

Inconsistency and validity

- Any argument with inconsistent premises is valid.
  - If premises are inconsistent, no valuation makes them all true.
    $\Rightarrow$ No valuation makes them all true and the conclusion false.

Inconsistency and validity

- Any argument with inconsistent premises is valid.
  - If premises are inconsistent, no valuation makes them all true.

    $\Rightarrow$ No valuation makes them all true and the conclusion false.

    $\Rightarrow$ No valuation can be a counterexample.

Inconsistency and validity

- Any argument with inconsistent premises is valid.
  - If premises are inconsistent, no valuation makes them all true.
    $\Rightarrow$ No valuation makes them all true and the conclusion false.
    $\Rightarrow$ No valuation can be a counterexample.
- An argument is valid if, and only if, the premises together with the **negation** of the conclusion are inconsistent.

Inconsistency and validity

- Any argument with inconsistent premises is valid.
  - If premises are inconsistent, no valuation makes them all true.

    $\Rightarrow$ No valuation makes them all true and the conclusion false.

    $\Rightarrow$ No valuation can be a counterexample.
- An argument is valid if, and only if, the premises together with the **negation** of the conclusion are inconsistent.
- This fact is the basis our tree method for validity ('STD')!

LSAT puzzle: Can you send Amir in the boat?

A, B, C, D: Amir, Betty, Chad, Dana are in the boat.

Amir won't go without Chad.

Chad only goes if at least one of Betty and Dana goes too.

Amir and Dana can't be in the boat together.

LSAT puzzle: Can you send Amir in the boat?

A, B, C, D: Amir, Betty, Chad, Dana are in the boat.

Amir won't go without Chad. (If no Chad, then no Amir)

Chad only goes if at least one of Betty and Dana goes too.

Amir and Dana can't be in the boat together.

LSAT puzzle: Can you send Amir in the boat?

A, B, C, D: Amir, Betty, Chad, Dana are in the boat.

Amir won't go without Chad. (If no Chad, then no Amir)

$$A \supset C$$

Chad only goes if at least one of Betty and Dana goes too.

Amir and Dana can't be in the boat together.

LSAT puzzle: Can you send Amir in the boat?

A, B, C, D: Amir, Betty, Chad, Dana are in the boat.

Amir won't go without Chad. (If no Chad, then no Amir)

$$A \supset C$$

Chad only goes if at least one of Betty and Dana goes too.

$$C \supset (B \vee D)$$

Amir and Dana can't be in the boat together.

LSAT puzzle: Can you send Amir in the boat?

A, B, C, D: Amir, Betty, Chad, Dana are in the boat.

Amir won't go without Chad. (If no Chad, then no Amir)

$$A \supset C$$

Chad only goes if at least one of Betty and Dana goes too.

$$C \supset (B \vee D)$$

Amir and Dana can't be in the boat together.

$$\sim(A \,\&\, D)$$

$$A \supset \sim D$$

$$\sim A \vee \sim D$$

Dependency resolution by SAT checking

A, B, C, D: package A, B, C, D is installed.

Package A depends on package C.

Package C requires either package B or D.

Package A is incompatible with package D.

Dependency resolution by SAT checking

A, B, C, D: package A, B, C, D is installed.

Package A depends on package C.

$$A \supset C$$

Package C requires either package B or D.

$$C \supset (B \vee D)$$

Package A is incompatible with package D.

$$\sim(A \,\&\, D)$$

$$A \supset \sim D$$

$$\sim A \vee \sim D$$

Solution as satisfiability question

Can you send **Amir** in the boat?

Can package **A** be installed?

Same as: Are these sentences consistent?

$$\mathbf{A}$$

$$A \supset C$$

$$C \supset (B \vee D)$$

$$\sim(A \,\&\, D)$$

More complex satisfiability questions

Can you send **Amir without Betty** in the boat?

Can package **A** be installed **without installing B**?

Same as: Are these sentences consistent?

$$A \,\&\, {\sim}B$$

$$A \supset C$$

$$C \supset (B \vee D)$$

$$\sim(A \,\&\, D)$$

(Exercise: construct a complete truth table. Which valuations, if any, satisfy all four sentences?)

Complexity of logical testing

- In general, testing for validity, satisfiability, tautology, etc., requires making a complete truth table

Complexity of logical testing

- In general, testing for validity, satisfiability, tautology, etc., requires making a complete truth table
  - Testing for validity requires checking **every** valuation (although can halt if find a counterexample to validity).

Complexity of logical testing

- In general, testing for validity, satisfiability, tautology, etc., requires making a complete truth table
  - Testing for validity requires checking **every** valuation (although can halt if find a counterexample to validity).
  - Testing for satisfiability requires finding **at least one** valuation.

Complexity of logical testing

- In general, testing for validity, satisfiability, tautology, etc., requires making a complete truth table
  - Testing for validity requires checking **every** valuation (although can halt if find a counterexample to validity).
  - Testing for satisfiability requires finding **at least one** valuation.
- If there are n sentence letters, there are 2^n valuations to check.

Complexity of logical testing

- In general, testing for validity, satisfiability, tautology, etc., requires making a complete truth table
  - Testing for validity requires checking **every** valuation (although can halt if find a counterexample to validity).
  - Testing for satisfiability requires finding **at least one** valuation.
- If there are n sentence letters, there are 2^n valuations to check.
- Computer scientists have yet to find a method that can (always) do this faster than truth tables (connected to "P vs NP problem").

Complexity of logical testing

- In general, testing for validity, satisfiability, tautology, etc., requires making a complete truth table
  - Testing for validity requires checking **every** valuation (although can halt if find a counterexample to validity).
  - Testing for satisfiability requires finding **at least one** valuation.
- If there are n sentence letters, there are 2^n valuations to check.
- Computer scientists have yet to find a method that can (always) do this faster than truth tables (connected to "P vs NP problem").
- See Cook-Levin Theorem

More on Complexity Classes

- **Class P**: solvable in polynomial time by a deterministic Turing machine (DTM)

More on Complexity Classes

- **Class P**: solvable in polynomial time by a deterministic Turing machine (DTM)
- **Class NP**: solvable in polynomial time by a non–deterministic Turing machine (NTM). Equivalently: can *verify/check* solutions by DTM in polynomial time

More on Complexity Classes

- **Class P**: solvable in polynomial time by a deterministic Turing machine (DTM)
- **Class NP**: solvable in polynomial time by a non-deterministic Turing machine (NTM). Equivalently: can *verify/check* solutions by DTM in polynomial time
- Boolean satisfiability and validity problems are *NP-complete*:

More on Complexity Classes

- **Class P**: solvable in polynomial time by a deterministic Turing machine (DTM)
- **Class NP**: solvable in polynomial time by a non-deterministic Turing machine (NTM). Equivalently: can *verify/check* solutions by DTM in polynomial time
- Boolean satisfiability and validity problems are *NP-complete*:
  - Problems that are easy to check, but difficult to solve

More on Complexity Classes

- **Class P**: solvable in polynomial time by a deterministic Turing machine (DTM)
- **Class NP**: solvable in polynomial time by a non-deterministic Turing machine (NTM). Equivalently: can *verify/check* solutions by DTM in polynomial time
- Boolean satisfiability and validity problems are *NP-complete*:
  - Problems that are easy to check, but difficult to solve
  - If we could solve these in polynomial time with DTM, then we could solve *ANY* NP problem in polynomial time

More on Complexity Classes

- **Class P**: solvable in polynomial time by a deterministic Turing machine (DTM)
- **Class NP**: solvable in polynomial time by a non-deterministic Turing machine (NTM). Equivalently: can *verify/check* solutions by DTM in polynomial time
- Boolean satisfiability and validity problems are *NP-complete*:
  - Problems that are easy to check, but difficult to solve
  - If we could solve these in polynomial time with DTM, then we could solve *ANY* NP problem in polynomial time
  - You might stand to make a lot of money (you would potentially have an efficient algorithm for many problems)!!!

More on Complexity Classes

- **Class P**: solvable in polynomial time by a deterministic Turing machine (DTM)
- **Class NP**: solvable in polynomial time by a non-deterministic Turing machine (NTM). Equivalently: can *verify/check* solutions by DTM in polynomial time
- Boolean satisfiability and validity problems are *NP-complete*:
  - Problems that are easy to check, but difficult to solve
  - If we could solve these in polynomial time with DTM, then we could solve *ANY* NP problem in polynomial time
  - You might stand to make a lot of money (you would potentially have an efficient algorithm for many problems)!!!
  - (At the very least, you could cash in for a million-dollar prize)

More on Complexity Classes

- **Class P**: solvable in polynomial time by a deterministic Turing machine (DTM)
- **Class NP**: solvable in polynomial time by a non-deterministic Turing machine (NTM). Equivalently: can *verify/check* solutions by DTM in polynomial time
- Boolean satisfiability and validity problems are *NP-complete*:
  - Problems that are easy to check, but difficult to solve
  - If we could solve these in polynomial time with DTM, then we could solve *ANY* NP problem in polynomial time
  - You might stand to make a lot of money (you would potentially have an efficient algorithm for many problems)!!!
  - (At the very least, you could cash in for a million-dollar prize)
  - But most experts think that class P $\neq$ class NP