2.25. Truth Trees:
Tautology, Contradiction, and Logical Equivalence

We've seen how truth trees preserve the better features of truth tables, with a lower cost in labor. But besides testing arguments for validity, we also used truth tables to test for tautologies, contradictions, and logical equivalence. Here we show that those tasks too fall within the truth tree's powers.

1. Logical Equivalence. Testing a pair of sentences for **logical equivalence** using truth tree is straightforward, since we noted previously that logical equivalence can be defined in terms of validity – something which truth trees already address.

> Two sentences are logically equivalent just when each **follows validly** from the other.

For instance, to show that "P" and "~~P" are logically equivalent we show that both of the following arguments are valid.

Valid	**Valid**
P	~~P
∴ ~~P	∴ P

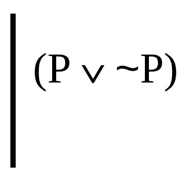
P		✓~~P	
	~~P ✓		P
✓~P			~P ✓
	P	P	
✖		✖	

And when two sentences aren't logically equivalent, truth trees show one or the other of the arguments to be **invalid**.

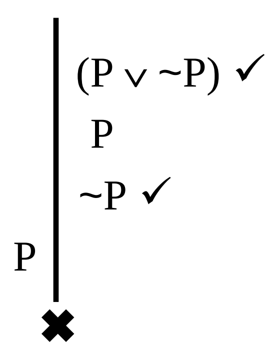
2. Tautology and Contradiction. A **tautology** was defined, in terms of truth tables, as a sentence true in every valuation; so there's no valuation – no 'possible situation' – where a tautology is false.

Our approach here mirrors the truth tree test for validity. To establish that no validity counterexample for an argument is possible, we assumed such a situation and showed that this assumption violates Bivalence at every turn. Likewise, to show there's no possible situation where the tautology is false, we assume such a situation and show how that violates Bivalence.

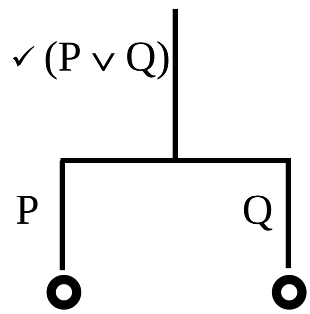
To demonstrate that "(P ∨ ~P)" is a tautology, we begin by picturing it as false – i.e., on the right of the tree.



The False Disjunction and False Negation Rules leave "P" on both left and right, and the tree closes.

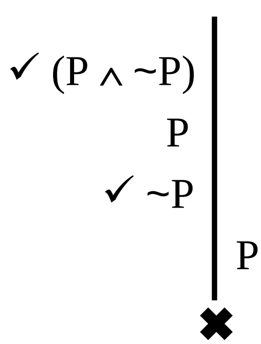


Note that to prove a sentence a tautology, it’s **not** an effective strategy to assume the sentence **true**, and show that every path stays open – taking that to mean that the sentence is true in every possible situation. For “(P ∨ Q)” is no tautology; but when we assume it true, every path stays open. That approach yields the *wrong* results.



As always with the tree method, we proceed indirectly: assuming the opposite, and tracing out the absurdity in that assumption.

To show that a sentence is a **contradiction** – true in **no** possible situation – we instead begin by assuming a situation where the sentence is true. “(P ∧ ~P),” for instance, is a contradiction. And assuming this sentence true closes the tree.



Furthermore, a sentence is **consistent** (**satisfiable**) just in case it’s not a contradiction – hence where it fails the contradiction tests. Hence to show a sentence consistent with a truth tree, we begin the contradiction test – putting the sentence on the left – and show that not every path closes. So the earlier truth tree with "(P ∨ Q)" on the left showed that sentence to be consistent (satisfiable), since at least one tree path remained open.

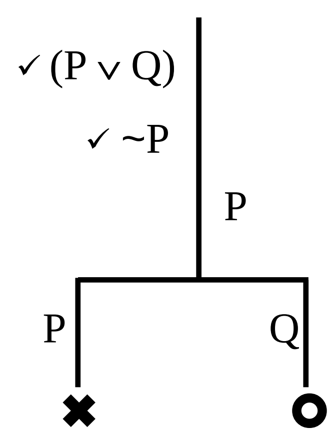
3. Consistency and Inconsistency. We extended the opposed concepts of **satisfiability** and **contradiction** beyond single sentences. In truth tables a set of sentences was **consistent** (or **simultaneously satisfiable**) if some valuation made every sentence in that set true; whereas it was **inconsistent** if no valuation made every sentence in the set true.

In each case the truth tree test applies just as with single sentences.

> Assume all the sentences **true** (on the left). If **every path closes**, the set is **inconsistent**; while if even **one path** remains **open**, the set is **consistent** (**simultaneously satisfiable)**.

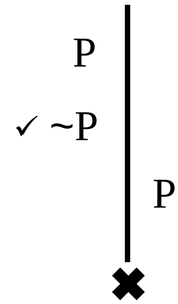
The set {(P ∨ Q), ~P} is consistent, as both truth tables and truth trees make clear.

P	Q	**~P**	**(P ∨ Q)**
1	1	0	1
1	0	0	1
0	1	**1**	**1**
0	0	1	0



But the set {P, ~P} is inconsistent.

P	**~P**
1	**0**
0	1



4. Validity and Inconsistency (Again). Finally we recall the link noted earlier between validity and inconsistency.[1] An invalid argument is an argument for which there’s a validity counterexample – a possible situation where the premises are true but the conclusion false. And since the conclusion is false just when its negation is true, such a validity counterexamples is mirrored by a set of sentences – the **counterexample set**.

> **Counterexample set** for an argument: the set
> {Premises, Negation of Conclusion}

A validity counterexample is a possible situation where the counterexample set is simultaneously satisfied – showing that the counterexample set is **consistent**. So we could define “validity” and “invalidity” in terms of consistency .

> **Invalid argument:** an argument whose counterexample set is **consistent**.

> **Valid argument:** an argument whose counterexample set is **inconsistent**.

With a truth tree test of inconsistency now in hand, the way is clear to test arguments for validity **by way of inconsistency**. If an argument’s counterexample set passes the consistency test, the argument is invalid; whereas if it doesn’t, the argument is valid. However, the results are somewhat underwhelming.

[1] In 2.20.

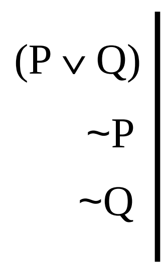
This familiar argument is certainly valid.

$(P \vee Q)$. $\sim P$ $\therefore$ Q

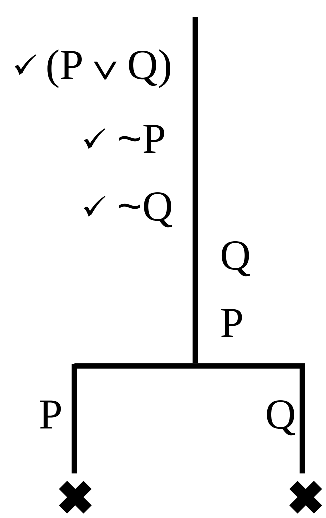
So its counterexample set must be inconsistent.

$\{(P \vee Q), \sim P, \sim Q\}$

To establish inconsistency we assume every sentence in the set is true.



We then show that such an assumption closes every tree path.



Since counterexample set $\{(P \vee Q), \sim P, \sim Q\}$ is **inconsistent**, the argument "$(P \vee Q)$. $\sim P$ $\therefore$ Q" is indeed valid.

Yet note the striking similarity between the inconsistency-based tree test and the original truth tree test of validity: but for the extra “~Q” in the new approach, they’re **identical**.

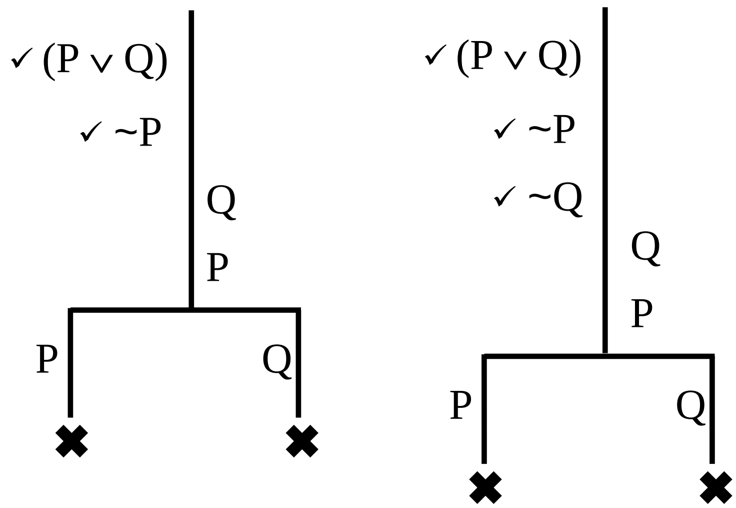
Argument

$(P \vee Q)$

$\sim P$

$\therefore Q$

Original Tree Test of Validity | **New Inconsistency-Based Test of Validity**



This highlights an important moral: though the link between validity and inconsistency might earlier have seemed like a theoretical curiosity not worth mentioning, we now recognize that link as the **foundation** of the truth tree test of validity. The truth tree test shows the argument valid precisely by showing that its counterexample set is inconsistent.[2]

[2] This point is explored further in 2.32 §2.

Summary

Logical Equivalence, Tautology, and Contradiction

- To show that two sentences are **logically equivalent**, show that each follows validly from the other.

- To show that a sentence is a **tautology**, put the sentence on the right (false) side, and show that every path closes.

- To show that a sentence is a **contradiction**, put the sentence on the left (true) side, and show that every path closes.

Consistency and Inconsistency

- To show that a set of sentences is **consistent** (**simultaneously satisfiable**), put all the sentences on the left (true) side, and show that at least one path stays open.

- To show that a set of sentences is **inconsistent**, put all the sentences on the left (true) side, and show that every path closes.