all assignments under which all formulas of Γ *are true, also* φ *is true, one says that* φ *is a logical consequence of* Γ*, which is denoted as*

$$\Gamma \models \varphi$$

When Γ *is the empty set one says that* φ *is valid, which is denoted as*

$$\models \varphi$$

Notice that the notion of validity of a propositional formula φ, corresponds to the nonexistence of assignments for which φ is false. Then by simple observations of the definitions, we have the following lemma.

Lemma 2 (Satisfiability versus validity)**.**

i. Any valid formula is satisfiable.

ii. The negation of a valid formula is unsatisfiable

Proof. i. Let φ be a propositional formula such that $\models \varphi$. Then given any assignment d, φ is true under d. Thus, φ is satisfiable.

ii. Let φ be a formula such that $\models \varphi$. Then for all assignments φ is true, which implies that for all assignments $(\neg\varphi)$ is false. Then there is no possible assignment for which $(\neg\varphi)$ is true. Thus, $(\neg\varphi)$ is unsatisfiable.

□

1.6 Soundness and Completeness of the Propositional Logic

The notions of *soundness* (or *correctness*) and *completeness* are not restricted to deductive systems being also applied in several areas of computer science. For instance, we can say that a sorting algorithm is sound or correct, whenever for any possible input, that is a list of keys, this algorithm computes as result a *sorted* list, according to some ordering which allows comparison of these keys. Unsoundness or incorrectness of the algorithm could happen, when for a specific input the algorithm cannot give as output a sorted version of the input; for instance, the algorithm can compute as output a unordered list containing all keys in the input, or it can omit some keys that appear in the input list, or it can include some keys that do not appear in the input list, etc. In the context of logical deduction, correctness means intuitively that all derived formulas are in fact semantically correct. Following our example, the sorting algorithm will be said to be complete, whenever it is capable to sort all possible input lists. An incomplete sorting algorithm may be unable to sort simple cases such as the cases of the empty or unitary lists, or may be unable to sort lists with repetitions. From the point of view of logical deduction, completeness can be intuitively interpreted as the capability of a deductive method of building proofs for all possible logical consequences.

1.6.1 Soundness of the Propositional Logic

The propositional calculus, as given by the rules of natural deduction presented in Table 1.3, allows derivation of semantically *sound* (or *correct*) conclusions. For instance, rule $(\wedge_i)$, allows a derivation for the sequent $\varphi, \psi \vdash \varphi \wedge \psi$, which is semantically correct because whenever φ and ψ are true, $\varphi \wedge \psi$ is true; that is denoted as $\varphi, \psi \models \varphi \wedge \psi$. The *correctness* of the propositional logic is formalized in the following theorem.

Theorem 3 (Soundness of the propositional logic)**.** *If $\Gamma \vdash \varphi$, for a finite set of propositional formulas $\Gamma = \{\gamma_1, \dots, \gamma_n\}$, then $\Gamma \models \varphi$. This can be summarized as*

$$\Gamma \vdash \varphi \textit{ implies } \Gamma \models \varphi$$

And for the case of Γ equal to the empty set, we have that provable theorems are valid formulas:

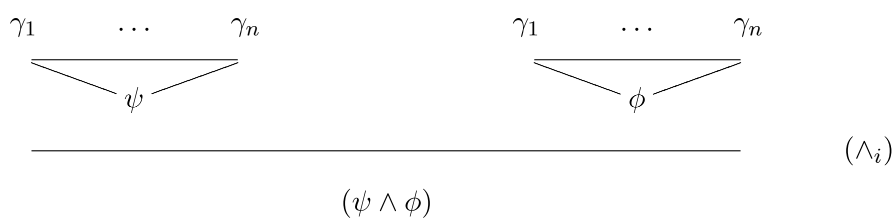
$$\vdash \varphi \textit{ implies } \models \varphi$$

Proof. The proof is by induction on the structure of derivations. We will consider the last step of a derivation having as consequence the formula φ and as assumptions only formulas of Γ.

IB The most simple derivations are those that correspond to a simple selection of the set of assumptions, that are derivations for sequents in which the conclusion is an assumption belonging to the set Γ; that is, $\gamma_1, \dots, \gamma_i (= \varphi), \dots, \gamma_n \vdash \varphi$. Notice that these derivations are correct since $\gamma_1, \dots, \gamma_i (= \varphi), \dots, \gamma_n \models \varphi$.

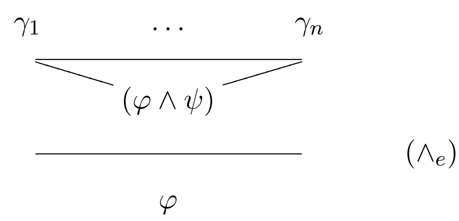
IS For the inductive step, we will consider the last rule (from the Table 1.3) applied in the derivation, supposing correctness of all previous fragments (or subtrees) of the proof.

Case $(\wedge_i)$. For a derivation finishing in an application of this rule, the last step of the proof gives as conclusion φ that should be of the form $(\psi \wedge \phi)$, for formulas ψ and ϕ, that are the premises of the last step of the proof. This is depicted in the following figure.



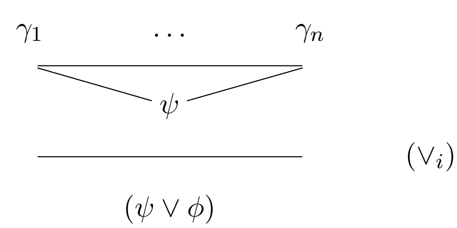
The left premise is the root of a derivation tree for the sequent $\Gamma \vdash \psi$ and the right one, for the sequent $\Gamma \vdash \phi$. In fact, not all assumptions in Γ need to be open leaves of these subtrees. By induction hypothesis, one has both $\Gamma \models \psi$ and $\Gamma \models \phi$. Thus, for all assignments that made the formulas in Γ true, the formulas ψ and ϕ are also true, which implies that $(\psi \wedge \phi)$ is true too. Consequently, $\Gamma \models \varphi$.

Case $(\wedge_e)$. For a derivation finishing in an application of this rule, one obtains as conclusion the formula φ from a premise of the form $(\varphi \wedge \psi)$. This is depicted in the figure below.



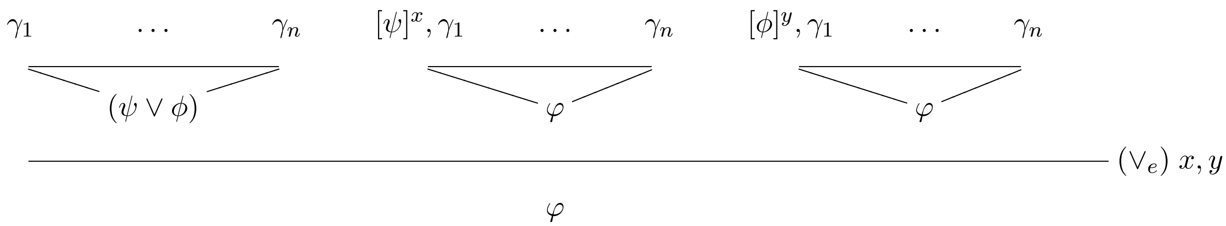
The subtree rooted by the formula $(\varphi \wedge \psi)$ has open leaves labelled with assumptions of the set Γ; not necessarily all these formulas. This subtree is a derivation for the sequent $\Gamma \vdash (\varphi \wedge \psi)$. By induction hypothesis, one has that $\Gamma \models (\varphi \wedge \psi)$, which means that all assignments which make true all formulas in Γ, make also true the formula $(\varphi \wedge \psi)$ and consequently both formulas φ and ψ. Thus, one can conclude that all assignments that make true all formulas in Γ, make also true φ; that is, $\Gamma \models \varphi$.

Case $(\vee_i)$. For a derivation finishing in an application of this rule, the conclusion, that is the formula φ, should be of the form $\varphi = (\psi \vee \phi)$, and the premise of the last rule is ψ as depicted in the following figure.



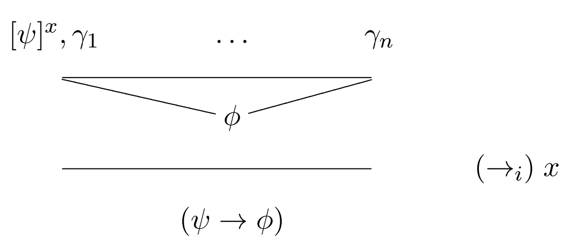
The subtree rooted by the formula ψ and with open leaves labelled by formulas of Γ, corresponds to a derivation for the sequent $\Gamma \vdash \psi$, that by induction hypothesis implies $\Gamma \models \psi$. This implies that all assignments that make the formulas in Γ true, make also ψ true and consequently, the formula $(\psi \vee \phi)$ is true too, under these assignments. Thus, $\Gamma \models \varphi$.

Case $(\vee_e)$. For a derivation of the sequent $\Gamma \vdash \varphi$ that finishes in an application of this rule, one has as premises formulas $(\psi \vee \phi)$, and two repetitions of φ. The former premise labels a root of a subtree with open leaves labelled by assumptions in Γ, that corresponds to a derivation for the sequent $\Gamma \vdash (\psi \vee \phi)$, for some formulas ψ and ϕ. The latter two repetitions of φ, are labeling subtrees with open leaves in Γ and $[\psi]^x$, the first one, and $[\phi]^y$, the second one, as depicted in the figure below.



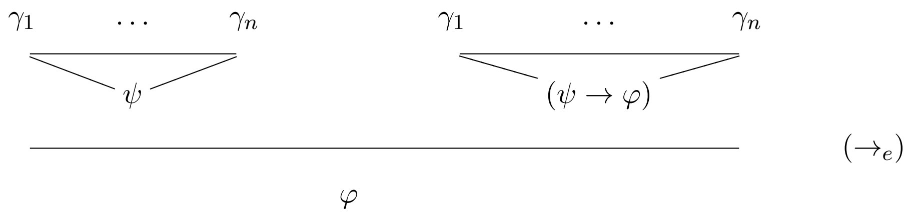
The left subtree whose root is labelled with formula φ, corresponds to a derivation for the sequent $\Gamma, \psi \vdash \varphi$, and the right subtree with φ as root, to a derivation for the sequent $\Gamma, \phi \vdash \varphi$. By induction hypothesis, one has $\Gamma \models (\psi \vee \phi)$, $\Gamma, \psi \models \varphi$ and $\Gamma, \phi \models \varphi$. The first means, that for all assignments that make the formulas in Γ true, $(\psi \vee \phi)$ is also true. And by the semantics of the logical connective $\vee$, $(\psi \vee \phi)$ is true if at least one of the formulas ψ or ϕ is true. In the case that ψ is true, since $\Gamma, \psi \models \varphi$, φ should be true too; in the case in which ϕ is true, since $\Gamma, \phi \models \varphi$, φ should be true as well. Then whenever all formulas in Γ are true, φ is true as well, which implies that $\Gamma \models \varphi$.

Case $(\rightarrow_i)$. For a derivation that finishes in an application of this rule, φ should be of the form $(\psi \rightarrow \phi)$, for some formulas ψ and ϕ. The premise of the last step in this derivation should be the formula ϕ. This formula labels the root of subtree that is a derivation for the sequent $\Gamma, \psi \vdash \phi$. See the next figure.



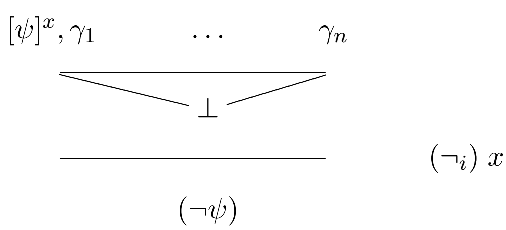
By induction hypothesis, one has that $\Gamma, \psi \models \phi$, which means that for all assignments that make all formulas in Γ and ψ true, ϕ is also true. Suppose, one has an assignment d, that makes all formulas in Γ true. If ψ is true under this assignment, ϕ is also true. If ψ is false under this assignment, by the semantical interpretation of the connective $\to$, $\psi \to \phi$ is also true under this assignment. Thus, one can conclude that for any assignment that makes all formulas in Γ true, the formula φ, that is $\psi \to \phi$ is true too. Consequently, $\Gamma \models \varphi$.

Case $(\to_e)$. If the last step of the derivation is $(\to_e)$, then its premises are formulas of the form ψ and $(\psi \to \varphi)$, for some formula ψ, as illustrated in the figure below.



The left subtree corresponds to a derivation for the sequent $\Gamma \vdash \psi$ and the right one to a derivation for the sequent $\Gamma \vdash (\psi \to \varphi)$. By induction hypothesis, one has both $\Gamma \models \psi$ and $\Gamma \models (\psi \to \varphi)$. This means that any assignment that makes all formulas in Γ true also makes ψ and $(\psi \to \varphi)$ true. By the semantical interpretation of implication, whenever both ψ and $\psi \to \varphi$ are true, φ should be also true, which implies that $\Gamma \models \varphi$.

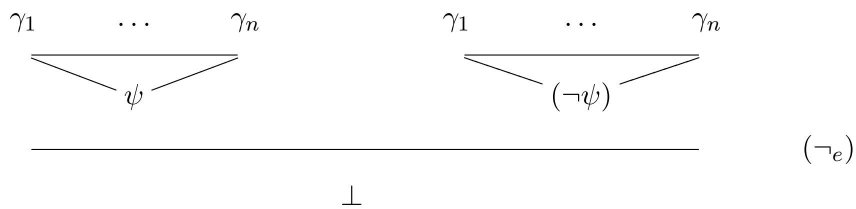
Case $(\neg_i)$. When this is the last applied rule in the derivation, φ is of the form $(\neg\psi)$, and the premise of the last step is $\perp$ as depicted in the next figure.



The subtree rooted by $\perp$, has open leaves labelled by formulas in Γ and ψ and corresponds to a proof of the sequent $\Gamma, \psi \vdash \perp$. By induction hypothesis, one has that $\Gamma, \psi \models \perp$, which means that for all assignments that make all formulas in Γ and ψ true, $\perp$ should also be true. But, by the semantical interpretation of $\perp$, this is always false. Then, there is no possible assignment, that makes all formulas in Γ and ψ true. Consequently, any assignment that makes all formulas in Γ true should make ψ false and, by the interpretation of the connective

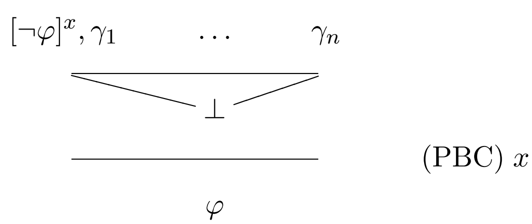
$\neg$, it makes $(\neg\psi)$ true. Thus, one can conclude that $\Gamma \models \varphi$.

Case $(\neg_e)$. For a derivation with last applied rule $(\neg_e)$, the conclusion, that is φ, is equal to the atomic formula $\bot$ and the premises of the last applied rule are formulas ψ and $(\neg\psi)$, for some formula ψ as illustrated in the next figure.



The derivation has open leaves labelled by formulas in Γ. The left and right subtrees respectively rooted by ψ and $\neg\psi$. correspond to derivations for the sequents $\Gamma \vdash \psi$ and $\Gamma \vdash \neg\psi$. By induction hypothesis, one has both $\Gamma \models \psi$ and $\Gamma \models \neg\psi$, which means that any assignment that makes all formulas in Γ true makes also both ψ and $\neg\psi$ true. Consequently, there is no assignment that makes all formulas in Γ true. Thus, one concludes that $\Gamma \models \bot$.

Case (PBC). For a derivation with last applied rule (PBC), the situation is illustrated in the next figure.



One has $\neg\varphi, \Gamma \vdash \bot$ and by induction hypothesis, $\neg\varphi, \Gamma \models \bot$. The latter implies that no assignment makes $\neg\varphi$ and all formulas in Γ true. Consequently, for any assignment that makes all formulas in Γ true, $\neg\varphi$ should be false and consequently φ true. Thus, one concludes $\Gamma \models \varphi$.

□