Logic for Computer Science. Knowledge Representation and Reasoning.

Lecture Notes

for

Computer Science Students
Faculty EAIiIB-IEiT AGH

Antoni Ligęza

Other support material:

`http://home.agh.edu.pl/~ligeza`

`https://ai.ia.agh.edu.pl/pl:dydaktyka:logic:start#logic_for_computer_science2020`

Propositional Calculus

- Alphabet,
- Syntax,
- Semantics,
- Logical implication ($\models$) and logical equivalence,
- Logical derivation ($\vdash$),
- Truth Tables,
- Functional completeness,
- Properties: satisfiability, unsatisfiability, tautologies,...
- Tautology verification,
- Minterms and implicents,
- Maxterms and implicants,
- CNF – Conjunctive Normal Form,
- DNF – Disjunctive Normal Form,
- Transformations preserving logical equivalence,
- minimal and maximal normal forms (CNF, DNF),
- transformation to CNF/DNF,
- Some observations on maxCNF and maxDNF; the Π and Σ notation shorthand,
- Normal forms of 0 and 1,
- Important tasks of logic...

The Alphabet of Propositional Calculus

A propositional variable can be assigned some meaning, e.g.:

$$p \stackrel{\text{def}}{=} \text{'Everybody is excited with this logic lecture'.}$$

Definition 1 *Propositional Calculus Alphabet:*

- P — *the set of propositional symbols (propositional variables),*

$$P = \{p, q, r, \ldots, p_1, q_1, r_1, \ldots, p_2, q_2, r_2, \ldots\},$$

- $\neg$ – *negation,*
- $\wedge$ – *conjunction,*
- $\vee$ – *disjunction (rather than alternative),*
- $\Rightarrow$ – *implication (also: $\Leftarrow$),*
- $\Leftrightarrow$ – *equivalence (two-side implication),*
- *two special symbols:*
  - $\top$ – *a formula always true (note that it is not the* **True** *value),*
  - $\bot$ – *a formula always false (note that it is not the* **False** *value),*
- *parentheses.*

There are many notations for logical connectives!

See: `https://en.wikipedia.org/wiki/Logical_connective`

By use of these logical connectives and propositional symbols one builds more complex logical formulas (formulae) of Propositional Calculus.

Not all expressions built with use of these symbols are Well-Formed Formulae (WFF).

Well-Formed Formula must satisfy the syntax rules; see next page.

Syntax

Definition 2 *Definition of legal formulas:*

- $\top$ *i* $\bot$ *are formulas,*
- *every* $p \in P$ *is a formula,*
- *if* ϕ, ψ *are formulas, then:*
  - $\neg(\phi)$ *is a formula (also:* $\neg(\psi)$*),*
  - $(\phi \wedge \psi)$ *is a formula,*
  - $(\phi \vee \psi)$ *is a formula,*
  - $(\phi \Rightarrow \psi)$ *is a formula,*
  - $(\phi \Leftrightarrow \psi)$ *is a formula,*
  - *and nothing else.*

Set of formulas = $\mathbf{FOR}$.

Every formula has a parsing tree. There is a grammar defining WFF.

Atomic formulas; atoms — simple propositional symbols; more exactly:

$$\mathrm{ATOM} = P \cup \{\top, \bot\}$$

Literals: atoms or their negations;

Positive literals: atomic formulas (with no negation);

Negative literals: negated atomic formulas ($\neg p$).

Pair of complementary literals: $\{p, \neg p\}$.

Clauses: disjunctions of literals: $(p \vee q \vee \neg r \vee s)$

Horn clauses: clauses with at most one positive literal: $(h \vee \neg p \vee \neg q)$, i.e.:

$$p \wedge q \rightarrow h$$

Hierarchy of Logical Connectives – Parentheses Elimination

The hierarchy of logical connectives (from top to bottom):

- negation ($\neg$),
- conjunction ($\wedge$),
- disjunction ($\vee$),
- implication ($\Rightarrow$),
- equivalence ($\Leftrightarrow$).

It allows to eliminate parentheses... Look for examples.

Some philosophical questions:

- What in fact does a negation mean?
  `https://en.wikipedia.org/wiki/Negation`
- Finite or infinite worlds? Closed-World Assumption vs. Open World
- Negation-as-Failure vs. Strong Negation
  `https://en.wikipedia.org/wiki/Stable_model_semantics#Strong_negation`
- Logical negation versus material negation!
- Do we need negation?

Semantics

Interpretation I maps propositional symbols into $\mathcal{T} = \{\mathbf{T}, \mathbf{F}\}$.

Definition 3 *Let P be a set of propositional symbols. Interpretations is defined as:*

$$I\colon P \longrightarrow \{\mathbf{T}, \mathbf{F}\}, \tag{1}$$

Notation: $I(\phi) = \mathbf{T}$ is noted as $\models_I \phi$; $I(\phi) = \mathbf{F}$ is noted as $\not\models_I \phi$

Definition 4 *The Interpretation I is extended over all formulas ϕ, ψ, φ from* **FOR** *as follows:*

- $I(\top) = \mathbf{T}$ *(* $\models_I \top$ *),*
- $I(\bot) = \mathbf{F}$ *(* $\not\models_I \bot$ *),*
- $\models_I \neg\phi$ *iff* $\not\models_I \phi$*,*
- $\models_I (\phi \wedge \psi)$ *iff* $\models_I \phi$ *and* $\models_I \psi$*,*
- $\models_I (\phi \vee \psi)$ *iff* $\models_I \phi$ *or* $\models_I \psi$*,*
- $\models_I (\phi \Rightarrow \psi)$ *iff* $\models_I \psi$ *or* $\not\models_I \phi$*,*
- $\models_I (\phi \Leftrightarrow \psi)$ *iff* $\models_I (\phi \Rightarrow \psi)$ *and* $\models_I (\psi \Rightarrow \phi)$*.*

Definition 5 **Equivalence** *Formulas ϕ and ψ are logically equivalent iff for any I:*

$$\models_I \phi \qquad \textit{iff} \qquad \models_I \psi. \tag{2}$$

Definition 6 **Logical Implication** *Formula ψ is logical consequence of ϕ iff for any I:*

$$\textit{if} \qquad \models_I \phi \qquad \textit{then} \qquad \models_I \psi. \tag{3}$$

Truth Tables

ϕ	$\neg\phi$
F	**T**
T	**F**

ϕ	φ	$\phi \wedge \varphi$
F	**F**	**F**
F	**T**	**F**
T	**F**	**F**
T	**T**	**T**

ϕ	φ	$\phi \vee \varphi$
F	**F**	**F**
F	**T**	**T**
T	**F**	**T**
T	**T**	**T**

ϕ	φ	$\phi \Rightarrow \varphi$
F	**F**	**T**
F	**T**	**T**
T	**F**	**F**
T	**T**	**T**

ϕ	φ	$\phi \Leftrightarrow \varphi$
F	**F**	**T**
F	**T**	**F**
T	**F**	**F**
T	**T**	**T**

An Engineering Notation; Towards Boolean Algebra

Instead of the *True* and *False* we often use the 1 and 0 values; this simplifies notation in some cases. It is also applied in the Boolean Algebra.

The Truth Tables looks as follows:

p	$\neg p$
0	1
1	0

The case of conjunction:

p	q	$p \wedge q$
0	0	0
0	1	0
1	0	0
1	1	1

The case of disjunction:

p	q	$p \vee q$
0	0	0
0	1	1
1	0	1
1	1	1

The case of implication:

p	q	$p \Rightarrow q$
0	0	1
0	1	1
1	0	0
1	1	1

Tabular Definitions of Logical Connectives !?! !?!

ϕ	ψ	$\neg\phi$	$\phi \wedge \psi$	$\phi \vee \psi$	$\phi \Rightarrow \psi$	$\phi \Leftrightarrow \psi$
true	*true*	*false*	*true*	*true*	*true*	*true*
true	*false*	*false*	*false*	*true*	*false*	*false*
false	*true*	*true*	*false*	*true*	*true*	*false*
false	*false*	*true*	*false*	*false*	*true*	*true*

Semantics through equivalent transformation:

- $\phi \Rightarrow \psi \equiv \neg\phi \vee \psi$,
- $\phi \Leftrightarrow \psi \equiv (\phi \Rightarrow \psi) \wedge (\phi \Leftarrow \psi)$,
- $\phi|\psi \equiv \neg(\phi \wedge \psi)$ –Sheffer function or `NAND`; also noted as $\overline{\phi \wedge \psi}$,
- $\phi \downarrow \psi \equiv \neg(\phi \vee \psi)$ – Pierce function or `NOR`; other notation $\overline{\phi \vee \psi}$,
- $\phi \bigoplus \psi \equiv (\neg\phi \wedge \psi) \vee (\phi \wedge \neg\psi)$ — `EX-OR`,

But there are many other functions possible...
For n arguments there are as many as 2^{2^n} functions, so for $n = 2$ there is 16 different functions.
Try to justify this statement.
Try to find a systematic way to define all the functions of two arguments.

Functional Completeness !?! !?!

Definition 7 *A Set of Functions is functionally complete if it allows to express any logical function.*

Some examples:

AND, OR, NOT:

$$\{\neg, \wedge, \vee\}$$

AND, NOT:

$$\{\neg, \wedge\}$$

OR, NOT:

$$\{\neg, \vee\}$$

IMPLICATION, NOT:

$$\{\neg, \Rightarrow\}$$

NAND:

$$\{|\}$$

NOR:

$$\{\downarrow\}$$

Definition 8 *A functionally complete set of functions is minimal — if it cannot be further reduced without violating functional completeness.*

Is the implication itself a *functionally complete* set? But it can be: how to solve this problem?

For convenience, redundant systems are in use.

Properties of Formulas !?! !?!

A formula ϕ may be:

true/satisfied — for interpretation I, $\models_I \phi$,

false/unsatisfied — for interpretation I, $\not\models_I \phi$,

satisfiable — if there exists an interpretation I such that $\models_I \phi$,

falsifiable/may be false — if there exists an interpretation I, $\not\models_I \phi$,

tautology/valid — if for any interpretation I, $\models_I \phi$; we write:

$$\models \phi$$

always false — if for any interpretation I:

$$\not\models \phi$$

What are the mutual relationships – if any – between the formulas satisfying the following definitions?

- formula Ψ is a *logical consequence* of formula Φ, to be denoted as $\Phi \models \Psi$ iff for any interpretation I satisfying Φ, I satisfies also Ψ;
- formula Ψ is *derivable* from formula Φ, to be denoted as $\Phi \vdash \Psi$ iff there exists a sequence of (valid) inference rules transforming Φ into Φ;
- Such a derivation is called a linear derivation;
- a formula can be derived from a set Δ of formulas (the axioms; the Knowledge Base); in this case we often present the derivation in a form of inversed tree.

Most important equivalent transformations

- $\neg\neg\phi \equiv \phi$ — double negation elimination,
- $\phi \wedge \psi \equiv \psi \wedge \phi$ — conjunction alternation,
- $\phi \vee \psi \equiv \psi \vee \phi$ — disjunction alternation,
- $(\phi \wedge \varphi) \wedge \psi \equiv \phi \wedge (\varphi \wedge \psi)$ — commutativity,
- $(\phi \vee \varphi) \vee \psi \equiv \phi \vee (\varphi \vee \psi)$ — commutativity,
- $(\phi \vee \varphi) \wedge \psi \equiv (\phi \wedge \psi) \vee (\varphi \wedge \psi)$ — distributive law,
- $(\phi \wedge \varphi) \vee \psi \equiv (\phi \vee \psi) \wedge (\varphi \vee \psi)$ — distributive law,
- $\phi \wedge \phi \equiv \phi$ — idempotency,
- $\phi \vee \phi \equiv \phi$ — idempotency,
- $\phi \wedge \bot \equiv \bot$, $\phi \wedge \top \equiv \phi$ — identity,
- $\phi \vee \bot \equiv \phi$, $\phi \vee \top \equiv \top$ — identity,
- $\phi \vee \neg\phi \equiv \top$ — *tertium non datur*; excluded middle,
- $\phi \wedge \neg\phi \equiv \bot$ — falsification,
- $\neg(\phi \wedge \psi) \equiv \neg(\phi) \vee \neg(\psi)$ — De Morgan rule,
- $\neg(\phi \vee \psi) \equiv \neg(\phi) \wedge \neg(\psi)$ — De Morgan rule,
- $\phi \Rightarrow \psi \equiv \neg\psi \Rightarrow \neg\phi$ — contraposition,
- $\phi \Rightarrow \psi \equiv \neg\phi \vee \psi$ — implication elimination.

Some basic relationships between implications !?! !?!

Simple (direct) statement:

$$p \Rightarrow q$$

The Inverse Statement (causality analysis):

$$q \Rightarrow p$$

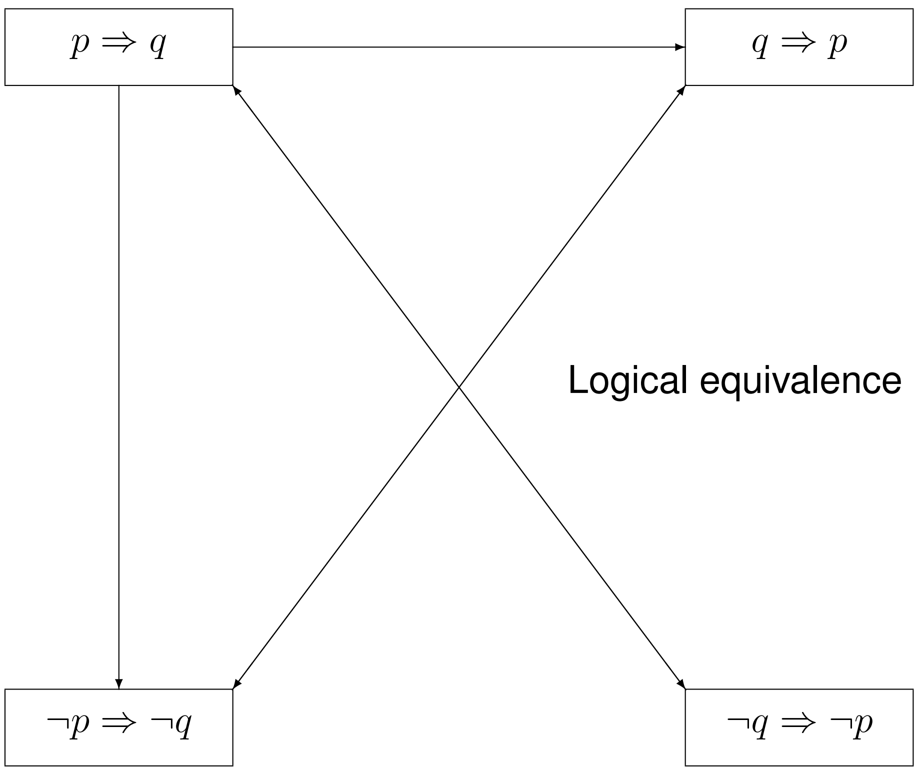
The Opposite/Contrary Statement(building exclusive rules):

$$\neg p \Rightarrow \neg q$$

The Contradictive Statement (proof by contradiction)

$$\neg q \Rightarrow \neg p$$

The Square of Logical Statements:



See also: https://en.wikipedia.org/wiki/Square_of_opposition

Example: Tautology Verification

$$\phi = ((p \Rightarrow r) \wedge (q \Rightarrow r)) \Leftrightarrow ((p \vee q) \Rightarrow r)$$

There are exactly 2^3 possible interpretations; we enumerate them in a consecutive way.

N	p	q	r	$p \Rightarrow r$	$q \Rightarrow r$	$(p \Rightarrow r) \wedge (q \Rightarrow r)$	$(p \vee q) \Rightarrow r$	Φ
0	0	0	0	1	1	**1**	**1**	**1**
1	0	0	1	1	1	**1**	**1**	**1**
2	0	1	0	1	0	**0**	**0**	**1**
3	0	1	1	1	1	**1**	**1**	**1**
4	1	0	0	0	1	**0**	**0**	**1**
5	1	0	1	1	1	**1**	**1**	**1**
6	1	1	0	0	0	**0**	**0**	**1**
7	1	1	1	1	1	**1**	**1**	**1**

Other possibility — through equivalence preserving transformations:

$$\phi \equiv ((\neg p \vee r) \wedge (\neg q \vee r)) \Leftrightarrow (\neg(p \vee q) \vee r).$$

$$\phi \equiv ((\neg p \wedge \neg q) \vee r) \Leftrightarrow (\neg(p \vee q) \vee r).$$

$$\phi \equiv (\neg(p \vee q) \vee r) \Leftrightarrow (\neg(p \vee q) \vee r).$$

Let us put: $\psi = (\neg(p \vee q) \vee r)$; so we see:

$$\phi \equiv \psi \Leftrightarrow \psi,$$

What about the following examples? Logical equivalence ($\equiv$)or logical implication ($\models$)? If so, which way? Try your intuitions first!

$$\phi = ((p \Rightarrow r) \wedge (q \Rightarrow r)) \Leftrightarrow ((p \wedge q) \Rightarrow r)$$

$$\phi = ((p \Rightarrow r) \vee (q \Rightarrow r)) \Leftrightarrow ((p \vee q) \Rightarrow r)$$

Example: Logical Consequence Verification (EX-LCV16)

$$\frac{(p \Rightarrow q) \wedge (r \Rightarrow s)}{(p \vee r) \Rightarrow (q \vee s)}$$

Put:

$$\phi = (p \Rightarrow q) \wedge (r \Rightarrow s)$$

and

$$\varphi = (p \vee r) \Rightarrow (q \vee s),$$

Now, check if:

$$\phi \models \varphi. \tag{4}$$

N	p	q	r	s	$p \Rightarrow q$	$r \Rightarrow s$	$(p \Rightarrow q) \wedge (r \Rightarrow s)$	$p \vee r$	$q \vee s$	$(p \vee r) \Rightarrow (q \vee s)$
0	0	0	0	0	1	1	**1**	0	0	**1**
1	0	0	0	1	1	1	**1**	0	1	**1**
2	0	0	1	0	1	0	**0**	1	0	**0**
3	0	0	1	1	1	1	**1**	1	1	**1**
4	0	1	0	0	1	1	**1**	0	1	**1**
5	0	1	0	1	1	1	**1**	0	1	**1**
6	0	1	1	0	1	0	**0**	1	1	**1**
7	0	1	1	1	1	1	**1**	1	1	**1**
8	1	0	0	0	0	1	**0**	1	0	**0**
9	1	0	0	1	0	1	**0**	1	1	**1**
10	1	0	1	0	0	0	**0**	1	0	**0**
11	1	0	1	1	0	1	**0**	1	1	**1**
12	1	1	0	0	1	1	**1**	1	1	**1**
13	1	1	0	1	1	1	**1**	1	1	**1**
14	1	1	1	0	1	0	**0**	1	1	**1**
15	1	1	1	1	1	1	**1**	1	1	**1**

From analysis of columns 8 (ϕ) and 11 (φ) the logical consequence is confirmed (but not equivalence; see row enumerated as: 6, 9, 11 and 14).

Minterms

Definition 9 *Literal A literal is an atomic formula p or its negation $\neg p$.*

Definition 10 *Let $q_1, q_2, \ldots q_n$ are literals:*

$$\phi = q_1 \wedge q_2 \wedge \ldots \wedge q_n$$

is a minterm, simple conjunction or product.

Lemma 1 *Minterm is satifiable iff it does not contain a pair of complementary literals.*

Lemma 2 *Minterm is unsatisfiable iff it contains a pair of complementary literals.*

Notation:

$$\phi = q_1 \wedge q_2 \wedge \ldots \wedge q_n$$

or

$$\phi = q_1 q_2 \ldots q_n$$

or a set of literals of a minterm ϕ

$$[\phi] = \{q_1, q_2, \ldots q_n\}$$

Definition 11 *Minterm ϕ subsumes minterm ψ iff $[\phi] \subseteq [\psi]$.*

Lemma 3 *Let ϕ and ψ are any minterms; then :*

$$\psi \models \phi \qquad \textit{iff} \qquad [\phi] \subseteq [\psi].$$

Think of conjunction as a constraint; a longer conjunction is a stronger constraint, since more literals must be satisfied!

Maxterms

Definition 12 *Let $q_1, q_2, \ldots q_n$ are literals; then:*

$$\phi = q_1 \vee q_2 \vee \ldots \vee q_n$$

is a maxterm, simple disjunction or a clause.

Lemma 4 *Maxterm is falsifiable iff it does not contain a pair of complementary literals.*

Lemma 5 *Maxterm is a tautology iff it contains a pair of complimentary literals.*

Definition 13 *Maxterm ψ subsumes maxterm ϕ iff*

$$[\psi] \subseteq [\phi]$$

Lemma 6 *Let ϕ and ψ are any maxterms; then:*

$$\psi \models \phi \qquad \textit{iff} \qquad [\psi] \subseteq [\phi].$$

Think of disjunction as a constraint; a longer disjunction is a weaker constraint since more literals are allowed to be satisfied! Let us consider a clause:

$$\psi = \neg p_1 \vee \neg p_2 \vee \ldots \vee \neg p_k \vee h_1 \vee h_2 \vee \ldots \vee h_m$$

After applying the de Morgan rule:

$$\neg(p_1 \wedge p_2 \wedge \ldots \wedge p_k) \vee (h_1 \vee h_2 \vee \ldots \vee h_m)$$

This can be put as:

$$p_1 \wedge p_2 \wedge \ldots \wedge p_k \Rightarrow h_1 \vee h_2 \vee \ldots \vee h_m$$

Definition 14 *A clause of the form:*

$$\psi = \neg p_1 \vee \neg p_2 \vee \ldots \vee \neg p_k \vee h$$

is called a Horn Clause.

Alternative notations:

$$p_1 \wedge p_2 \wedge \ldots \wedge p_k \Rightarrow h.$$

In PROLOG or in DATALOG:

$$h : -p_1, p_2, \ldots, p_k.$$

also:

```
h :- p_1, p_2,..., p_k.


h if p_1 and p_2 and ... and p_k.
```

Three forms of Horn clauses:

- facts,
- full clauses,
- constraints/calls.

Important intutions (EX-LCV16, the rightmost column):

- minterms define the 1-s of the table EX-LCV16; there are 13 of them,
- maxterms define the 0-s of the table EX-LCV16; there are only 3 of them.

But how do they look like? How to join them in order to define the formula?

CNF — Conjunctive Normal Form !?! Tips & tricks !?!

Definition 15 *Formula Ψ is in Conjunctive Normal Form (CNF; also called: Conjunction of Clauses, Product of Sums) iff*

$$\Psi = \psi_1 \wedge \psi_2 \wedge \ldots \wedge \psi_n$$

where $\psi_1, \psi_2, \ldots, \psi_n$ are clauses. Notation: $[\Psi] = \{\psi_1, \psi_2, \ldots, \psi_n\}$.

Examples:
Which of the following are in CNF:

1. $(p \vee g \vee \neg r) \wedge (p \vee r) \wedge \neg r$
2. $((p \wedge q) \vee \neg r) \wedge (p \vee r) \wedge \neg r$
3. $\neg(p \vee q) \wedge (p \vee r) \wedge \neg r$
4. $(M \longrightarrow I) \wedge (\neg M \longrightarrow (\neg I \wedge L)) \wedge ((I \vee L) \longrightarrow H) \wedge (H \longrightarrow G)$
5. $(\neg M \vee I) \wedge (M \vee \neg I) \wedge (M \vee L) \wedge (\neg I \vee H) \wedge (\neg L \vee H) \wedge (\neg H \vee G)$

Definition 16 *Implicent of a CNF formula — a clause, such that if it is false then the respective formula is also false.*

An implicent falisfies a CNF formula. Must it be equal to some of the clauses of the considered formula in CNF?

Definition 17 *A formula is in maximal CNF form (canonical CNF form) iff it is composed of all full/maximal clauses:*

$$maxCNF(\Psi) = \psi_1 \wedge \psi_2 \wedge \ldots \wedge \psi_n$$

all $\psi_1, \psi_2, \ldots, \psi_n$ contain all propositional symbols in use.

Definition 18 *Formula*

$$\Psi = \psi_1 \wedge \psi_2 \wedge \ldots \wedge \psi_n$$

in CNF is minimal iff it cannot be reduced without violating logical equivalence.

CNF — appropriate for inconsistency checking. But also basic for SAT!

The always false formula $\perp$ of n propositional variables can be represented in maximal CNF in a unique way and it consists of 2^n different clauses, each of n propositional symbols (negated or not), e.g.:

$$\perp = pqr \wedge pq\bar{r} \wedge p\bar{q}r \wedge p\bar{q}\bar{r} \wedge \bar{p}qr \wedge \bar{p}q\bar{r} \wedge \bar{p}\bar{q}r \wedge \bar{p}\bar{q}\bar{r} \quad \text{(CNF)}$$

Why the formula is always false?

This formula is also called the normal form of 0.

Some observations on CNF:

Important intuition: A $maxCNF$ covers 1:1 all the 0-s in the truth table (e.g. EX-LCV16). Write all the 3 maxterms/clauses just looking at the table – as an example...

A CNF can contain single literals as components; this can be explored as the single literal clause/unit preference strategy in SAT and Resolution Theorem Proving:

$$p \wedge (p \vee q) \wedge (\neg p \vee q \vee r)$$

Weaker components (clauses) in CNF can be absorbed — this leads to simplification of the CNF

$$p \quad \equiv \quad p \wedge (p \vee q)$$

Clauses different in one position — defined by complementary literals can be resolved (RR):

$$(p \vee q) \wedge (p \vee \neg q) \quad \equiv \quad p$$

A CNF containing complementary literals as unit clauses is immediately false:

$$p \wedge (q \vee \neg s) \wedge \neg p \quad \equiv \quad \bot$$

DNF — Disjunctive Normal Form !?! !?!

Definition 19 *Formula Φ is in Disjunctive Normal Form (DNF; also called: Disjunction of Minterms, Sum of Products) iff*

$$\Phi = \phi_1 \vee \phi_2 \vee \ldots \vee \phi_n$$

where $\phi_1, \phi_2, \ldots, \phi_n$ are any minterms. Notation: $[\Phi] = \{\phi_1, \phi_2, \ldots, \phi_n\}$.

Example:
Which of the following are in DNF:

1. $(p \wedge q) \vee ((p \vee \neg q) \wedge (\neg p \vee \neg q)))$
2. $(p \wedge q) \vee ((p \vee q) \vee \neg(p \wedge q)))$
3. $(p \wedge q) \vee ((p \wedge \neg q) \vee (\neg p \wedge \neg q)))$
4. $(p \wedge q) \vee (p \wedge \neg q) \vee (\neg p \wedge q) \vee (\neg p \wedge \neg q)$
5. Try to find the DNF of the CNF No. 5 – example from page 20...

Definition 20 *Implicant of a DNF formula — a minterm, such that if it is true, then the respective formula is also true.*

An implicant validates a DNF formula. Must it be equal to some of the components of the formula?

Definition 21 *A maximal DNF form (canonical DNF form) is any formula containing all the possible minterms:*

$$maxDNF(\Phi) = \phi_1 \vee \phi_2 \vee \ldots \vee \phi_n$$

where all the minterms $\phi_1 \vee \phi_2 \vee \ldots \vee \phi_n$ are composed of all the propositional symbols in use.

Definition 22 *Formula*

$$\Phi = \phi_1 \vee \phi_2 \vee \ldots \vee \phi_n$$

in DNF is minimal iff it cannot be reduced without violating logical equivalency.

DNF — is appropriate for checking satisfiability.

The formula always true $\top$ containing n propositional variables can be transformed to maximal DNF form in a unique way and it is composed of 2^n different products, each of them of n variables (negated or not), e.g.:

$$\top = pqr \vee pq\bar{r} \vee p\bar{q}r \vee p\bar{q}\bar{r} \vee \bar{p}qr \vee \bar{p}q\bar{r} \vee \bar{p}\bar{q}r \vee \bar{p}\bar{q}\bar{r} \qquad \text{(DNF)}$$

Why the formula is always true? This formula is also called the normal form of 1.

Some observations on DNF:

Important intuition: A $maxDNF$ covers 1:1 all the 1-s in the truth table (e.g. EX-LCV16). Write all the 13 full minterms just looking at the table – as en example...

A DNF can contain single literals as components; this can be explored in the single literal/unit preference strategy in Dual Resolution Theorem Proving and looking for falsifying interpretations:

$$p \vee (p \wedge q) \vee (\neg p \wedge q \wedge r)$$

Stronger components (minterms) in CNF can be absorbed — this leads to simplification of the DNF

$$p \quad \equiv \quad p \vee (p \wedge q)$$

Minterms different in one position — defined by complementary literals can be resolved (Dual RR):

$$(p \wedge q) \vee (p \wedge \neg q) \quad \equiv \quad p$$

A DNF containing complementary literals as unit minterms is immediately true:

$$p \vee (q \wedge \neg s) \vee \neg p \quad \equiv \quad \top$$

Transformation to CNF/DNF

1. $\Phi \Leftrightarrow \Psi \equiv (\Phi \Rightarrow \Psi) \wedge (\Psi \Rightarrow \Phi)$ – elimination of equivalence,
2. $\Phi \Rightarrow \Psi \equiv \neg\Phi \vee \Psi$ – elimination of implication,
3. $\neg(\neg\Phi) \equiv \Phi$ – elimination of double negations,
4. $\neg(\Phi \vee \Psi) \equiv \neg\Phi \wedge \neg\Psi$ – De Morgan's rule,
5. $\neg(\Phi \wedge \Psi) \equiv \neg\Phi \vee \neg\Psi$ – De Morgan's rule,
6. $\Phi \vee (\Psi \wedge \Upsilon) \equiv (\Phi \vee \Psi) \wedge (\Phi \vee \Upsilon)$ – distributivity rule; towards CNF,
7. $\Phi \wedge (\Psi \vee \Upsilon) \equiv (\Phi \wedge \Psi) \vee (\Phi \wedge \Upsilon)$ – distributivity rule; towards DNF.

Example:

$$(p \wedge (p \Rightarrow q)) \Rightarrow q \equiv \neg(p \wedge (p \Rightarrow q)) \vee q \equiv$$
$$\neg(p \wedge (\neg p \vee q)) \vee q \equiv (\neg p \vee \neg(\neg p \vee q)) \vee q \equiv$$
$$(\neg p \vee (p \wedge \neg q)) \vee q \equiv \neg\mathbf{p} \vee (\mathbf{p} \wedge \neg\mathbf{q}) \vee \mathbf{q} \equiv$$
$$(\neg p \vee p) \wedge (\neg p \vee \neg q) \vee q \equiv \neg p \vee \neg q \vee q \equiv \neg p \vee \top \equiv \top.$$

Example:
Transforming CNF to DNF:

- $\phi = ((p \vee q) \wedge (p \vee r) \wedge (q \vee s) \wedge (r \vee s)), \quad \psi = ((p \wedge s) \vee (q \wedge r))$
- $\phi = ((p \vee q) \wedge (q \vee r) \wedge (r \vee p)), \quad \psi = ((p \wedge q) \vee (q \wedge r) \vee (r \wedge p))$
- $\phi = ((p \vee q \vee r) \wedge (q \vee r \vee s) \wedge (r \vee s \vee p)) \quad \psi = ((p \wedge q) \vee (p \wedge s) \vee (q \wedge s) \vee r).$

Example EX-LCV16 continued: Comparing DNF

Let us reconsider:

$$\phi = (p \Rightarrow q) \wedge (r \Rightarrow s),$$

$$\varphi = (p \vee r) \Rightarrow (q \vee s).$$

We check for logical implication:

$$\phi \models \varphi.$$

Transform ϕ to DNF:

$$\begin{aligned} \phi &= (p \Rightarrow q) \wedge (r \Rightarrow s) = (\neg p \vee q) \wedge (\neg r \vee s) = \\ &= (\neg p \wedge \neg r) \vee (\neg p \wedge s) \vee (q \wedge \neg r) \vee (q \wedge s). \end{aligned}$$

and next to its maxDNF form:

$$\begin{aligned} maxDNF(\phi) = &(\neg p \wedge \neg q \wedge \neg r \wedge \neg s) \vee (\neg p \wedge \neg q \wedge \neg r \wedge s) \vee (\neg p \wedge \neg q \wedge r \wedge s) \vee \\ &(\neg p \wedge q \wedge \neg r \wedge \neg s) \vee (\neg p \wedge q \wedge \neg r \wedge s) \vee (\neg p \wedge q \wedge r \wedge s) \vee \\ &(p \wedge q \wedge \neg r \wedge \neg s) \vee (p \wedge q \wedge \neg r \wedge s) \vee (p \wedge q \wedge r \wedge s). \end{aligned}$$

Transform φ to DNF:

$$\begin{aligned} \varphi &= (p \vee r) \Rightarrow (q \vee s) = \neg(p \vee r) \vee q \vee s = (\neg p \wedge \neg r) \vee q \vee s = \\ &= (\neg p \wedge \neg r) \vee q \vee s. \end{aligned}$$

and next to its maxDNF form:

$$\begin{aligned} maxDNF(\varphi) = &(\neg p \wedge \neg q \wedge \neg r \wedge \neg s) \vee (\neg p \wedge \neg q \wedge \neg r \wedge s) \vee (\neg p \wedge \neg q \wedge r \wedge s) \vee \\ &(\neg p \wedge q \wedge \neg r \wedge \neg s) \vee (\neg p \wedge q \wedge \neg r \wedge s) \vee (\neg p \wedge q \wedge r \wedge s) \vee \\ &(\neg p \wedge q \wedge r \wedge \neg s) \vee (p \wedge q \wedge \neg r \wedge \neg s) \vee (p \wedge q \wedge \neg r \wedge s) \vee \\ &(p \wedge q \wedge r \wedge s) \vee (p \wedge q \wedge r \wedge \neg s) \vee (p \wedge \neg q \wedge \neg r \wedge s) \vee \\ &(p \wedge \neg q \wedge r \wedge s). \end{aligned}$$

and so we have all the 9 full minterms covering all the 1-s of column ϕ and all the 13 full minterms covering all the 1-s columnn φ ofthe EX-LCV16 table; check them!

Further on, it can be seen that:

$$[maxDNF(\phi)] \subseteq [maxDNF(\varphi)],$$

Could it be checked earlier - without generating the maxDNF forms?

Important: short Σ notation (i.e the sum of products): Note that taking into account the enumeration of the minterms in the leftmost column of the EX-LCV16, the ϕ and φ formulas can be represented as *the sums of products* in the following form:

$$maxDNF(\phi) = \Sigma(0, 1, 3, 4, 5, 7, 12, 13, 15)$$

and

$$maxDNF(\varphi) = \Sigma(0, 1, 3, 4, 5, 6, 7, 9, 11, 12, 13, 14, 15)$$

In this shorthand notation it is well-visible that in fact $maxDNF(\varphi)$ *covers* $maxDNF(\phi)$.

Example EX-LCV16 continued: Comparing CNF

Let us reconsider:

$$\phi = (p \Rightarrow q) \wedge (r \Rightarrow s),$$

$$\varphi = (p \vee r) \Rightarrow (q \vee s).$$

We check for logical implication:

$$\phi \models \varphi.$$

Transform ϕ to CNF:

$$\phi = (p \Rightarrow q) \wedge (r \Rightarrow s) = (\neg p \vee q) \wedge (\neg r \vee s).$$

and next to its maxCNF form:

$$\begin{aligned} maxCNF(\phi) = &(p \vee q \vee \neg r \vee s) \wedge (p \vee \neg q \vee \neg r \vee s) \wedge \\ &(\neg p \vee q \vee r \vee s) \wedge (\neg p \vee q \vee r \vee \neg s) \wedge \\ &(\neg p \vee q \vee \neg r \vee s) \wedge (\neg p \vee q \vee \neg r \vee \neg s) \wedge \\ &(\neg p \vee \neg q \vee \neg r \vee s). \end{aligned}$$

Transform φ to CNF:

$$\begin{aligned} \varphi &= (p \vee r) \Rightarrow (q \vee s) = \neg(p \vee r) \vee q \vee s = (\neg p \wedge \neg r) \vee q \vee s = \\ &= (\neg p \vee q \vee s) \wedge (\neg r \vee q \vee s). \end{aligned}$$

and next to its maxCNF form:

$$\begin{aligned} maxCNF(\varphi) = &(p \vee q \vee \neg r \vee s) \wedge \\ &(\neg p \vee q \vee r \vee s) \wedge \\ &(\neg p \vee q \vee \neg r \vee s) \wedge . \end{aligned}$$

and so we have 7 full maxterms covering all the 0-s of the ϕ column and 3 full maxterms covering all the 0-s of the φ column of the EX-LCV16 table; check them!

Further on, it can be seen that:

$$[maxCNF(\varphi)] \subseteq [maxCNF(\phi)],$$

Could it be checked earlier - without generating the maxCNF forms?

Important: short Π notation: Note that taking into account the enumeration of the maxterms in the leftmost column the ϕ and φ formulas can be represented as *the product of sums* in the following form:

$$maxCNF(\phi) = \Pi(2, 6, 8, 9, 10, 11, 14)$$

and

$$maxCNF(\varphi) = \Pi(2, 8, 10)$$

Can you see the relationship between the Σ and the Π representation of the respective formulas?

Maximal CNF and DNF Forms – Two Observations

Let ϕ and ψ be two propositional formulas.

We have:

Lemma 7 [1]

$$\phi \models \psi \quad \textit{iff} \quad [maxDNF(\phi)] \subseteq [maxDNF(\psi)]$$

For intuition, all the 1-s of ϕ are covered by the 1-s of ψ.

Also:

Lemma 8 [2]

$$\phi \models \psi \quad \textit{iff} \quad [maxCNF(\psi)] \subseteq [maxDNF(\phi)]$$

For intuition, all the 0-s of ψ are covered by the 0-s of ϕ.

Conclusions:

- Two propositional formulas ϕ and ψ are logically equivalent iff their maximal CNF forms are identical (up to the order of components).
- Two propositional formulas ϕ and ψ are logically equivalent iff their maximal DNF forms are identical (up to the order of components).

[1] Corrected w.r.t former edition

[2] Corrected w.r.t former edition

Logic for KRR – Tasks and Tools

- Theorem Proving – Verification of Logical Consequence:

$$\Delta \models H;$$

- Automated Inference – Derivation:

$$\Delta \vdash H;$$

- SAT (checking for models) – satisfiability:

$$\models_I H;$$

- un-SAT verification – unsatisfiability:

$$\not\models_I H \quad \text{for any interpretation I;}$$

- Tautology verification (completeness):

$$\models H$$

- valid inference rules – checking:

$$(\Delta \vdash H) \quad \longrightarrow \quad (\Delta \models H)$$

- complete inference rules – checking:

$$(\Delta \models H) \quad \longrightarrow \quad (\Delta \vdash H)$$

- finding minimal forms: CNF and DNF.

Question: what are the areas of application of CNF vs. DNF? Why and when CNF vs. DNF?