Part 1: Formal Logic

Logical Connectives I: and or not

Martin Licht, Ph.D.

January 10, 2018

UC San Diego
Department of Mathematics

Math 109

Logical connectives allow us to combine statements into new statements.

We may **practically** think of them as arithmetic operations on statements.

For truth values we use the following notation.

true	T	1
false	F	0

Much of what follows can be seen as *Binary Circuits 101*.

Logical AND, also known as **conjunction**: Relates two statements and is true if and only if **both of them** are true.

P	Q	$P \wedge Q$
T	T	T
T	F	F
F	T	F
F	F	F

Example

- 4 is even and 4 is a square number: true
- 3 is odd and 7 is a prime number: true
- 5 is a rational and $\sqrt{2}$ is a rational: false

Logical OR, also known as **disjunction**: Relates two statements and is true if and only if at **least one of them** is true.

P	Q	$P \vee Q$
T	T	T
T	F	T
F	T	T
F	F	F

Example

- 4 is even or 4 is a square number: true
- The Riemann hypothesis is true or 2 is a prime number: true
- 1 is even or 9 is a prime number: false

Logical NEGATION: assumes the **opposite truth value** of a given statement.

P	$\neg P$
F	T
T	F

Example

- It is false that 4 is odd: true
- It is false that 7 is a prime number: false

Exclusive and inclusive OR.

The logical disjunction, i.e., the logical OR, can be represented more accurately by “and/or” in human language.

It is an **inclusive OR**: both statements may be true at the same time and are not mutually exclusive.

This is to be distinguished from the **exclusive OR**, which translates to “either ... or” in human language.

Logical Exclusive OR, also known as **XOR**: Relates two statements and is true if and only if exactly one of them is true.

P	Q	$P \oplus Q$
T	T	F
T	F	T
F	T	T
F	F	F

Example

- Either 4 is even or 4 is a square number: false
- Either 3 is odd or 7 is a prime number: false
- Either 5 is a rational or $\sqrt{2}$ is a rational: true
- Either 91 is prime or $1 = 3$: false

The logical operations may be composed. **Truth tables** can be a helpful device for that purpose.

Example

How does the truth value of

$$(\neg P \vee Q) \wedge (Q \vee P)$$

depend on the truth values of P and Q?

How to fill out the following truth table?

P	Q	$(\neg P \vee Q) \wedge (Q \vee P)$
T	T	?
T	F	?
F	T	?
F	F	?

We expand the subexpressions and work recursively:

P	Q	$(\neg P \vee Q) \wedge (Q \vee P)$	$\neg P$	$\neg P \vee Q$	$Q \vee P$
T	T				
T	F				
F	T				
F	F				

We expand the subexpressions and work recursively:

P	Q	$(\neg P \vee Q) \wedge (Q \vee P)$	$\neg P$	$\neg P \vee Q$	$Q \vee P$
T	T		F		
T	F		F		
F	T		T		
F	F		T		

We expand the subexpressions and work recursively:

P	Q	$(\neg P \lor Q) \land (Q \lor P)$	$\neg P$	$\neg P \lor Q$	$Q \lor P$
T	T		F	T	
T	F		F	F	
F	T		T	T	
F	F		T	T	

We expand the subexpressions and work recursively:

P	Q	$(\neg P \vee Q) \wedge (Q \vee P)$	$\neg P$	$\neg P \vee Q$	$Q \vee P$
T	T		F	T	T
T	F		F	F	T
F	T		T	T	T
F	F		T	T	F

We expand the subexpressions and work recursively:

P	Q	$(\neg P \vee Q) \wedge (Q \vee P)$	$\neg P$	$\neg P \vee Q$	$Q \vee P$
T	T	T	F	T	T
T	F	F	F	F	T
F	T	T	T	T	T
F	F	F	T	T	F

Could we have obtained the same result without *brute force*?

P	Q	$(\neg P \vee Q) \wedge (Q \vee P)$	Q
T	T	T	T
T	F	F	F
F	T	T	T
F	F	F	F

We can check the equivalence of two logical expressions that depend on several propositions $A, B, C, \dots$ by going over all possible combinations of truth values of the input propositions and then comparing the output truth values.

Questions?