

## Logical Statement

- Logical statement compound:  $\sim$  (neg): Not/Negation,  $\wedge$ : and,  $\vee$ : or
- Logical Equivalence:

Given any statement variables $p, q$ and $r$ , a tautology <b>true</b> and a contradiction <b>false</b> :			
1	Commutative laws	$p \wedge q \equiv q \wedge p$	$p \vee q \equiv q \vee p$
2	Associative laws	$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$	$(p \vee q) \vee r \equiv p \vee (q \vee r)$
3	Distributive laws	$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$	$p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
4	Identity laws	$p \wedge \text{true} \equiv p$	$p \vee \text{false} \equiv p$
5	Negation laws	$p \vee \sim p \equiv \text{true}$	$p \wedge \sim p \equiv \text{false}$
6	Double negative law	$\sim(\sim p) \equiv p$	

7	Idempotent laws	$p \wedge p \equiv p$	$p \vee p \equiv p$
8	Universal bound laws	$p \vee \text{true} \equiv \text{true}$	$p \wedge \text{false} \equiv \text{false}$
9	De Morgan's laws	$\sim(p \wedge q) \equiv \sim p \vee \sim q$	$\sim(p \vee q) \equiv \sim p \wedge \sim q$
10	Absorption laws	$p \vee (p \wedge q) \equiv p$	$p \wedge (p \vee q) \equiv p$
11	Negation of true and false	$\sim \text{true} \equiv \text{false}$	$\sim \text{false} \equiv \text{true}$

## Conditional Statement

- Only if**: If  $p$ , then  $q$  can be written as:  $p \rightarrow q$ 
  - $p$ 为前提premise,  $q$ 为结论conclusion
  - 只有当 $P$ 为真且 $Q$ 为假时, 整个语句为假。其他情况下都为真
- Implication law**:  $p \rightarrow q \equiv \sim p \vee q$ , with **negation**:  $\sim(p \rightarrow q) \equiv p \wedge \sim q$
- Contrapositive(逆否命题)**: 将原始命题两侧都取反并位置互换。逆否命题在逻辑上**等价于**原始命题
- Converse(逆命题)**: 将原始命题假设和结论的位置互换得到的命题。逆命题在逻辑上**并不一定等价于**原始命题。
- Inverse(反命题)**: 将原始命题的假设和结论都否定所得到的命题。反命题在逻辑上**并不一定等价于**原始命题。
- 反命题和逆命题**在逻辑上等价, **原始命题和逆否命题**在逻辑上等价

$p \rightarrow q$	$\equiv$	$\sim q \rightarrow \sim p$
conditional statement		contrapositive
$q \rightarrow p$	$\equiv$	$\sim p \rightarrow \sim q$
converse		inverse
Note that:		
$p \rightarrow q$	$\neq$	$q \rightarrow p$

- Biconditional (if and only if)(iff)( $\iff$ )**: 两个命题 $P$ 和 $Q$ 是等价的, 即 $P$ 成立当且仅当 $Q$ 成立。这意味着 $P$ 是 $Q$ 的必要条件, 同时 $Q$ 也是 $P$ 的必要条件。这个表达式仅在 $P$ 和 $Q$ 都为真或都为假时, 才为真。  
 $p \iff q \equiv (p \rightarrow q) \wedge (q \rightarrow p)$
- Argument(论证)**: 由两条前提(premises)构成, 第一个前提一般是包含了 *If  $p$ , then  $q$*  的条件语句, 第二个前提一般是对 $p$ 或 $q$ 的一个事实陈述。请注意, 如果第二个前提所陈述的是 $p$ , 那么结论应该是讨论 $q$ , 则归类为modus ponens。反之第二个前提所陈述的是 $q$ (一般为 $\sim q$ ), 则结论讨论的是 $p$ , 归为modus tollens
- 验证条件语句(Conditional Statement)是有效的(valid)**:
  - Modus ponens**: 给定一个条件语句(if  $p$ , then  $q$ ), 如果我们知道 $p$ 为真, 则可以得出结论 $q$ 也为真

- **Modus tollens**: 给定一个条件语句(if p, then q), 如果我们知道**非q为真**, 则可以得出结论**非p也为真**

- **Generalization**: 从特定示例推导出**一般性**的规则。

例如: Anton是少年。可以推导出一般性规则: Anton是少年 (包含幼年) 或成年 (包含老年)

- **Specialization**: 从一般性规则中推导出**更具体**的示例

例如: Ana理解代数, Ana理解图论。可以推导出具体的规则: Ana理解代数

- **Elimination**: 通过消除不可能或不适用的选项来逼近问题答案

例如:  $x-3=0$  or  $x+2=0$ ,  $x \geq 0$ 。如果知道x为正数, 则x不可能为-2, 可以得出结论 $x=3$

- **Transitivity**: 如果一个关系在一对对象间成立, 并且也在第二对对象间成立, 那么它同样在第一对和第三对对象间成立的逻辑

- **Proof by Division into Cases**: 将问题分解为几个更容易管理的情况或子问题, 并分别对每种情况进行证明

Rule of inference		
Modus Ponens	$p \rightarrow q$ $p$ • $q$	
Modus Tollens	$p \rightarrow q$ $\sim q$ • $\sim p$	
Generalization	$p$ • $p \vee q$	$q$ • $p \vee q$
Specialization	$p \wedge q$ • $p$	$p \wedge q$ • $q$
Conjunction	$p$ $q$ • $p \wedge q$	

Rule of inference		
Elimination	$p \vee q$ $\sim q$ • $p$	$p \vee q$ $\sim p$ • $q$
Transitivity	$p \rightarrow q$ $q \rightarrow r$ • $p \rightarrow r$	
Proof by Division Into Cases	$p \vee q$ $p \rightarrow r$ $q \rightarrow r$ • $r$	
Contradiction Rule	$\sim p \rightarrow \text{false}$ • $p$	

• **Fallacy**, 验证条件语句是无效的(*invalid*):

- **Ambiguous premises**: 使用含糊不清的前提, 但在推理过程中将它们当作明确无误的陈述。导致论证的结论站不住脚。
- **Circular reasoning**: 循环推理指论证在推理过程中直接或间接地预设了要证明的结论。例如“我之所以正确, 是因为我说的是对的”
- **Jumping to a conclusion**: 在论证的过程中**过于匆忙的得出结论**, 而没有充分的理由或证据支持。
- **Converse Error**: 错把逆命题等效于原始命题, 及将前提和结论互换。
- **Inverse Error**: 错把反命题等效于原始命题, 及将前提和结论都取反。

• **有效论证(Valid Argument)**: 如果所有前提都是真的, 那么结论必须是真的。有效性关注的是论证的形式和逻辑结构, 而**不涉及**前提的真实性。

• **真实论证(Sound Argument)**: 一个真实的论证不仅是**有效的(Valid)**, 同时其所有前提实际上都是**真的**。这意味着真实论证满足两个条件: 逻辑结构是正确的(有效性), 所有前提都是正确的(真实性)。

• **无效论证(Unsound Argument)**: 任何不满足有效论证(Valid Argument)定义的论证都是无效的。这意味着即使所有前提都是真的, 结论也不一定是真的。无效论证可能基于错误的逻辑结构(有效性), 或包含不真实的前提(真实性), 或两者都有。

## Quantified Statements

• **Predicate(谓词)**: 包含变量的语句。例如 $P(x)$ ="x is a student at school", 其中P为**predicate symbol**, x为谓词。如果将x替换成特定的值, 在此例子中替换为具体的名字, 那么 $P(x)$ 为**谓词语句(predicate statement)**。

• **Domain(域)**: 可以合法代入变量的所有可能值的集合。

- **Universal Quantifier**: 表达“对所有”的断言。符号是 $\forall$ 。
- **Universal Statement**: 例如 $\forall x \in D, Q(x)$ , 对于D域中的所有x, 命题Q(x)都是真的 (成立)
  - 当定义域D中的**所有元素**对谓词语句Q(x)都是真时, 该universal statement才能被定义为真
  - 反之, 只要在域中**存在至少一个元素**使得Q(x)为假, 该universal statement为假
    - 这被称为**反例(counterexample)**
- **Existential Quantifier(存在量词)**: 符号为 $\exists$ 。  $\exists x \in D$  such that  $Q(x)$ 表示在某个域中至少存在一个元素使得谓词语句Q(x)为真。 $\exists!$ 表示存在唯一, 即存在着有且仅有一个元素使得谓词语句为真。
- $\exists!$ : 此符号表示**存在唯一**, 指的是存在着有且仅有一个元素满足指定的条件或属性。
- **Skolemization**: - When a “ $\exists$ ” exists outside of a universal  $\forall$  quantifier, we can replace the variable with a new constant, known as a “Skolem constant”
  - For example:
    - $\exists x \text{ Female}(x)$   
Can be re-written as  $\text{Female}(c)$
    - Where c refers to one particular person in the universe of discourse
- **Universal Conditional Statement**:  $\forall x \in D, P(x) \rightarrow Q(x)$
- **Narrowing domain**: 通过缩小定义域的范围, 可以将谓词语句缩小范围。  
例如: 对于所有的x, 如果x是正方形, 那么x是长方形  
可以简化为: 对于所有的正方形x, x是长方形
- **Negation of Universal Conditional Statements**: 对于全称量词( $\forall$ )的否定可以表达成定义域内至少有一个元素使得谓词为假。即  
 $\sim (\forall x \in D, P(x)) \equiv \exists x \text{ such that } \sim (P(x) \rightarrow Q(x))$

Of special importance in mathematics.

$$\sim(\forall x, P(x) \rightarrow Q(x)) \equiv \exists x \text{ such that } \sim(P(x) \rightarrow Q(x)) \dots (A)$$

$$\sim(P(x) \rightarrow Q(x)) \equiv P(x) \wedge \sim Q(x) \dots (B)$$

Substituting (B) into (A):

$$\sim(\forall x, P(x) \rightarrow Q(x)) \equiv \exists x \text{ such that } P(x) \wedge \sim Q(x)$$

- Contrapositive, Converse, Inverse of Conditional Statements

#### Definition 3.2.1 (Contrapositive, converse, inverse)

Consider a statement of the form:  $\forall x \in D, P(x) \rightarrow Q(x)$ .

1. Its **contrapositive** is:  $\forall x \in D, \sim Q(x) \rightarrow \sim P(x)$ .
2. Its **converse** is:  $\forall x \in D, Q(x) \rightarrow P(x)$ .
3. Its **inverse** is:  $\forall x \in D, \sim P(x) \rightarrow \sim Q(x)$ .

If a real number is greater than 2, then its square is greater than 4.

The formal version:  $\forall x \in \mathbb{R}, x > 2 \rightarrow x^2 > 4$ .

**Contrapositive:**  $\forall x \in \mathbb{R}, x^2 \leq 4 \rightarrow x \leq 2$ .

If the square of a real number is less than or equal to 4, then the number is less than or equal to 2.

**Converse:**  $\forall x \in \mathbb{R}, x^2 > 4 \rightarrow x > 2$ .

If the square of a real number is greater than 4, then the number is greater than 2.

**Inverse:**  $\forall x \in \mathbb{R}, x \leq 2 \rightarrow x^2 \leq 4$ .

If a real number is less than or equal to 2, then the square of the number is less than or equal to 4.

- The universal conditional statement is:
  - **equivalent** to its contrapositive (逆否命题)
  - **not logically equivalent** to its converse (转置命题)
  - **not logically equivalent** to its inverse (逆命题)
- **Sufficient Condition (充分条件)**: 在语句“如果P(x), 那么Q(x)”中, P(x)是Q(x)的充分条件。这意味着P(x)的真实性足以保证Q(x)的真实性。换句话说, 每当P(x)发生时, Q(x)必然发生。但请注意, Q(x)的真实性不一定依赖于P(x); Q(x)也可能因为其他原因而为真。
- **Necessary Condition (必要条件)**: 在同一个语句中, Q(x)是P(x)的必要条件。这意味着Q(x)的真实性是P(x)真实性的必要条件。如果Q(x)不为真, 那么P(x)也不可能为真。但Q(x)的真实性并不保证P(x)的真实性, 因为即使Q(x)为真, P(x)也可能为假。

- **Only if:** “Only if”用于指出必要条件。当我们说“ $P(x)$  only if  $Q(x)$ ”时，意思是 $Q(x)$ 是 $P(x)$ 成立的必要条件。用另一种方式表达，就是“如果 $P(x)$ ，那么 $Q(x)$ ”。这强调了无论何时 $P(x)$ 为真， $Q(x)$ 也必须为真，但它并不意味着 $Q(x)$ 为真就足以使 $P(x)$ 为真。
- “如果 $P$ ，那么 $Q$ ” ( $P \rightarrow Q$ ) :  $P$ 是 $Q$ 的充分条件， $Q$ 是 $P$ 的必要条件。
- “ $P$  only if  $Q$ ”: 强调了 $Q$ 是 $P$ 成立的必要条件，与 $P \rightarrow Q$ 逻辑上等价。

#### Definition 3.2.2 (Necessary and Sufficient conditions, Only if)

- “ $\forall x, r(x)$  is a **sufficient condition** for  $s(x)$ ” means “ $\forall x, r(x) \rightarrow s(x)$ ”.
- “ $\forall x, r(x)$  is a **necessary condition** for  $s(x)$ ” means “ $\forall x, \sim r(x) \rightarrow \sim s(x)$ ” or, equivalently, “ $\forall x, s(x) \rightarrow r(x)$ ”.
- “ $\forall x, r(x)$  **only if**  $s(x)$ ” means “ $\forall x, \sim s(x) \rightarrow \sim r(x)$ ” or, equivalently, “ $\forall x, r(x) \rightarrow s(x)$ ”.

#### 1. $\forall x \in D, \exists y \in E$ such that $P(x, y)$

- 这个语句的意思是“对于 $D$ 中的每一个 $x$ ，存在至少一个 $E$ 中的 $y$ ，使得 $P(x, y)$ 成立。”
- 这里的关键在于，对于 $D$ 中的每一个元素，我们都可以找到至少一个在 $E$ 中的元素，使得某个条件 $P(x, y)$ 成立。这个“存在”的 $y$ 可能对于不同的 $x$ 而不同。

#### 1. $\exists x \in D$ such that $\forall y \in E, P(x, y)$

- 这个语句的意思是“存在至少一个 $D$ 中的 $x$ ，对于 $E$ 中的所有 $y$ ， $P(x, y)$ 都成立。”
- 这里的关键在于，我们能找到至少一个特定的 $x$ ，在这种情况下，对于 $E$ 中的每一个 $y$ ，条件 $P(x, y)$ 都是成立的。这意味着这个特定的 $x$ 与 $E$ 中所有的 $y$ 都满足条件 $P$ 。

## Inference Mechanisms

- We say that statement **a entails (蕴含)** statement **b**, if, in every model in which **a** is true, then **b** must be true. This is written as  $a \models b$ .  
Formally:  $a \models b \iff M(a) \subseteq M(b)$
- **逻辑蕴含(Logical Entailment):** 在一组命题（前提集合）的逻辑基础上必然导出另一个命题（结论）的关系。假设前提集合为 **a**，结论为 **b**，我们说 **a** 逻辑蕴含 **b**，则如果在所有 **a** 为真的情形下，**b** 也必定为真。
- In **model checking**, we try all possible combinations of truth values for each variable, and see if **a** is true in all of these.
- **Model Checking Steps:** 假设要证明KB逻辑蕴含 **a**
  1. 写出KB和 **a** 的Truth Table
  2. 找到KB为真(1)的所有行(Critical Rows)，查看 **a** 是否为真
  3. 如果有KB为真但 **a** 为假的行，则证明KB不能逻辑蕴含 **a**
  4. 反之，如果KB为真的所有行 **a** 都为真，则证明KB逻辑蕴含 **a**
- **归结(Resolution):** 是一种证明KB是否逻辑蕴含某个命题 **a** 的方法。归结特别适用于合取范式(CNF)。合取范式由多个字句构成，每个字句中的运算符都为或( $\vee$ )，字句之间的链接都为与( $\wedge$ )
- **归结(Resolution)的工作原理:**
  1. **规范CNF:** 在应用归结前，所有逻辑表达式必须转换成合取范式(CNF)。即命题a和KB都以合取范式的形式表示。
  2. **归结规则:** 归结法基于一个简单的规则，即归结规则，它允许我们从两个包含互补文字的子句中推导出新的子句。如果一个子句包含某个文字  $P$ ，另一个子句包含该文字的否定  $\neg P$ ，那么通过归结可以得到一个新的子句，该子句包含这两个子句中的所有其他文字，但不包括  $P$  和  $\neg P$ 。
- **确定性字句(definite clause):** 恰好包含一个正文字的字句。正文字指为被否定的变量，例如 $P$ 。否定的文字，例如 $\sim Q$ ，被视为负文字。
- **霍恩字句(Horn clause):** 包含最多一个正文字的字句。这意味着霍恩字句可以只有负文字，或者一个正文字加上任意数量的负文字。
- **Forward Chaining 正向链接算法的工作原理:**
  1. **初始化:** 从已知的事实（知识库）开始。
  2. **规则匹配:** 扫描所有的规则，找出其前提（如果部分）与当前已知事实匹配的规则。
  3. **规则应用:** 对于匹配的规则，将其结论（那么部分）加入到已知事实中。这可能会引入一些新的事实。
  4. **重复过程:** 重复步骤2和3，直到无法找到新的匹配规则，或者达到了特定的目标。
  5. **终止:** 当没有更多的规则可以应用时，算法终止。此时，所有可以从初始事实集和规则集推导出的结论都被推导出。
- **统一unification:** 统一用于确定是否可以通过变量替换使得两个或多个逻辑表达式在形式上相等。这种变量替换的集合称为**替换** (Substitution)  $\theta$ 。统一过程的目的是找到这样的替换 $\theta$ ，使得经过替换后，某些表达式在结构和内容上完全匹配