Mathese

Carl Pollard

Department of Linguistics
Ohio State University

September 29, 2011

And

- The standard abbreviation for *and* is the symbol $\wedge$, called **conjunction**.
- *And* is used for combining sentences to form a new sentence:

  S_1 and S_2. (Abbreviated form: $S_1 \wedge S_2$)
- A sentence formed this way is called a **conjunctive** sentence.
- Here S_1 is called the **first conjunct** and S_2 is called the **second conjunct**.
- A conjunctive sentence is considered to be true if both conjuncts are true. Otherwise it is false.

Or

- The standard abbreviation for *or* is the symbol $\vee$, called **disjunction**.
- *Or* is used for combining sentences to form a new sentence: S_1 or S_2. (Abbreviated form: $S_1 \vee S_2$)
- A sentence formed this way is called a **disjunctive** sentence.
- Here S_1 is called the **first disjunct** and S_2 is called the **second disjunct**.
- A disjunctive sentence is considered to be true if at least one disjunct is true. Otherwise it is false.

Implies (1/2)

- The standard abbreviation for *implies* is the symbol $\rightarrow$, called **implication**.
- Some authors write $\supset$ instead of $\rightarrow$ for implication.
- *Implies* is used for combining sentences to form a new sentence:

  S_1 implies S_2. (Abbreviated form: $S_1 \rightarrow S_2$)
- A synonym for 'implies' is 'if ..., then ...', as in:
  If S_1, then S_2.
- A sentence formed this way is called an **implicative** sentence, or alternatively, a **conditional** sentence.
- S_1 is called the **antecedent** and S_2 the **consequent**.

Implies (2/2)

- A conditional sentence is considered to be true if either the antecedent is false or the consequent is true (or both), *even if the antecedent and the consequent seem to have nothing to do with each other*. Otherwise it is false.
- For example:

  If there does not exist a set with no members, then $0 = 0$.

  is true!
- Another example:

  If $0 \neq 0$ then $1 \neq 1$.

  is true!

If and only if

- The standard abbreviation for *if and only if* is the symbol $\leftrightarrow$, called **biimplication**.
- A synonym for *if and only if* is the invented word *iff*.
- *If and only if (iff)* is used for combining sentences to form a new sentence:

  S_1 iff S_2. (Abbreviated form: $S_1 \leftrightarrow S_2$)
- A sentence formed this way is called an **biimplicative** sentence, or alternatively, a **biconditional** sentence.
- A biconditional sentence is considered to be true if either (1) both S_1 and S_2 are true, or (2) both S_1 and S_2 are false. Otherwise, it is false.
- S_1 iff S_2 can be thought of as shorthand for:

  S_1 implies S_2, and S_2 implies S_1.

It is not the case that (1/3)

- The standard abbreviation for *it is not the case that* is the symbol ¬, called **negation**.
- Some authors write ∼ instead of ¬ for negation.
- Negation is written before the sentence it negates:

  It is not the case that S. (Abbreviated form: ¬S)
- The sentence *it is not the case that S* is called the **negation** of S, or, equivalently, the **denial** of S, and S is called the **scope** of the negation.
- A sentence formed this way is called a **negative** sentence.
- More colloquial synonyms of *it is not the case that S* are *S not!* and *no way S*.
- Unsurprisingly, a negative sentence is considered to be true if the scope is false, and false if the scope is true.

It is not the case that (2/3)

- Often, the effect of negation with *it is not the case that* can be achieved by ordinary English **verb negation**, which involves:
  - replacing the finite verb (the one that agrees with the subject) V with 'does not V' if V is not an auxiliary verb (such as *has* or *is*), or
  - negating V with a following *not* or *-n't* if it *is* an auxiliary.
- for example, these pairs of sentences are equivalent (express the same thing):

  It is not the case that 2 belongs to 1.
  2 does not belong to 1.

  It is not the case that 1 is empty.
  1 isn't empty.

It is not the case that (3/3)

- But: negation by *it is not the case that* and verb negation cannot be counted on to produce equivalent effects if the verb is in the scope of a *quantifier* (see below).
- Example: these are not equivalent:

  (i) It is not the case that for every x, x belongs to x.
  (ii) For every x, x doesn't belong to x.
- For (i) is clearly true (for example, 0 doesn't belong to 0). But the truth or falsity of (ii) can't be determined on the basis of the assumptions about sets made in Chapter 1. (In fact, different ways of adding further set-theoretic assumptions resolve the issue in different ways.)

Variables (1/2)

- *Very* roughly speaking, Mathese variables are the counterparts of ordinary English pronouns (but without such distinctions as case, number, and gender).
- Variables are "spelled" as upper- or lower-case roman letters (usually italicized except in handwriting), with or without numerical subscripts, e.g. x, y, x_0, x_1, X, Y, etc.
- In a context where the subject matter is set theory, we think of variables as ranging over arbitrary sets.

Variables (2/2)

Unlike pronouns, variables are not ambiguous with respect to what their 'antecedents' are. If ordinary English had variables instead of pronouns, we could disambiguate the sentence:

A donkey kicked a mule, and then it told its mother.

as follows:

There exists x such that there exists y such that ...

x told x's mother
x told y's mother.
y told x's mother.
y told y's mother.

For all

- Mathese ‘for all’, abbreviated by the **universal quantifier** symbol $\forall$, forms a sentence by combining first with a variable and then with a sentence, as in:

  For all x, S (abbreviated form: $\forall x$S).

- The variable x is said to be **bound** by the quantifier, and the sentence S is called the **scope** of the quantifier.
- Synonyms of ‘for all’ include ‘for each’, ‘for every’, and ‘for any’.
- Usually the bound variable also occurs in the scope; if it doesn’t, then the quantification is said to be **vacuous**.
- A sentence formed in this way is said to be **universally quantified**, or simply **universal**.

Restricted Universal Sentences (1/2)

- As long as we are using Mathese only to talk about set theory, we can assume that the bound variable in a universal sentence ranges over all sets, that is, 'for all x' is implicitly understood as 'for all sets x'.
- However, often we want to universally quantify not over *every* set, but just over the sets that satisfy some condition on x, $\mathrm{S}_1[x]$. Then we say:

  For every x with $\mathrm{S}_1[x]$, $\mathrm{S}_2[x]$.
- This is understood to be shorthand for

  For every x, $\mathrm{S}_1[x]$ implies $\mathrm{S}_2[x]$. (Abbreviated form: $\forall x(\mathrm{S}_1[x] \rightarrow \mathrm{S}_2[x])$)
- A sentence of this form is called a **restricted universal sentence**.

Restricted Universal Sentences (2/2)

- A restricted universal sentence $\forall x(\mathrm{S}_1[x] \rightarrow \mathrm{S}_2[x])$ is true provided, for every x, either $\mathrm{S}_1[x]$ is false or $\mathrm{S}_2[x]$ is true.
- In that case, we say that $\mathrm{S}_1[x]$ is a **sufficient condition** for $\mathrm{S}_2[x]$, or, equivalently, that $\mathrm{S}_2[x]$ is a **necessary condition** for $\mathrm{S}_1[x]$.
- A special case of this is that a restricted universal sentence is true provided, no matter what x is, $\mathrm{S}_1[x]$ is false. Such a sentence is said to be **vacuously true**.
- For example, the sentence

  For every x with $x \neq x$, $x = 2$.

  is (vacuously) true.
- If a universal sentence of the form

  For every x, $\mathrm{S}_1[x]$ iff $\mathrm{S}_2[x]$

  (i.e. whose scope is a biconditional) is true, then we say $\mathrm{S}_1[x]$ is a **necessary and sufficient condition** for $\mathrm{S}_2[x]$.

There exists . . . such that

- Mathese 'there exists . . . such that', abbreviated by the **existential quantifier** symbol $\exists$, forms a sentence by combining first with a variable and then with a sentence, as in:

  There exists x such that S (abbreviated form: $\exists x$S).

- The variable x is said to be **bound** by the quantifier, and the sentence S is called the **scope** of the quantifier.
- Synonyms of 'there exists . . . such that' include 'for some' and 'there is a(n) . . . such that'.
- Usually the bound variable also occurs in the scope; if it doesn't, then the quantification is said to be **vacuous**.
- A sentence formed in this way is said to be **existentially quantified**, or simply **existential**.

Restricted Existential Sentences

- As long as we are using Mathese only to talk about set theory, we can assume that the bound variable in an existential sentence ranges over all sets, that is, 'there exists x' is implicitly understood as 'there exists a set x'.
- However, often we want to existentially quantify not over *every* set, but just over the sets that satisfy some condition $\mathrm{S}_1[x]$. Then we say:

  There exists x with $\mathrm{S}_1[x]$, such that $\mathrm{S}_2[x]$.
- This is understood to be shorthand for

  There exists x such that $\mathrm{S}_1[x]$ and $\mathrm{S}_2[x]$.
  (Abbreviated form: $\exists x(\mathrm{S}_1[x] \wedge \mathrm{S}_2[x])$)

Using Parentheses for Disambiguation

- Note the use of parentheses in restricted universal or existential formulas:

$$\forall x(\mathrm{S}_1[x] \rightarrow \mathrm{S}_2[x])$$
$$\exists x(\mathrm{S}_1[x] \wedge \mathrm{S}_2[x]))$$

- Without the parentheses, it would be hard to be sure whether the scope of the quantifier in the first (second) example was whole the conditional (conjunctive) formula or just its antecdent (first conjunct).
- This is a common notational device in FOL and other symbolic logical languages.
- Both round and square parentheses can be used.
- Multiple sets of parentheses can be used in the same formula.

Free Variables

- A variable in a sentence (or formula) is called **free** if it is not bound by any (universal or existential) quantifier.
- A sentence (or formula) is called **closed** if it has no free variables, and **open** otherwise.
- A sentence (or formula) whose free variables are $x_0, \ldots, x_n$ is often called a **condition** on $x_0, \ldots, x_n$.
- The number of free variables in a condition is called its **arity**. Thus conditions might be **nullary** (no free variables, i.e. a closed sentence), **unary** (one free variable), **binary** (two free variables), **ternary** (three free variables), etc.

There exists unique ... such that

- In Mathese, 'there exists unique ... such that' (abbreviated form: $\exists! x$) combines first with a variable, then with a sentence, as in:

  There exists unique x such that S. (Abbreviated form: $\exists! x$ S)

- This is understood to be shorthand for:

$$\exists x(\mathrm{S}[x] \land \forall y(\mathrm{S}[y] \rightarrow (y = x)))$$

Defining Predicates

- At the outset, the only predicates in Mathese are *equals* (abbreviated $=$) or synonyms such as *is the same as* or *is identical to*, and *is a member of* (abbreviated $\in$) or synonyms such as *belongs to* or *is an element of*.
- But we can *define* new predicates in terms of these and other predicates which have already been defined.
- The **arity** of a defined predicate is the arity of the condition that is used to define it.
- *Examples:* we defined "x is **empty**" to mean $\forall y(y \notin x)$, and "x is a **singleton**" to mean $\exists! y(y \in x)$.

  So *is empty* and *is a singleton* are unary predicates.
- *Example:* We defined "x **is a subset** of y" (abbreviation: $x \subseteq y$) to mean $\forall z(z \in x \rightarrow z \in y)$.

  So $\subseteq$ is a binary predicate.

Defining Names

- If we can prove, for some unary condition $\mathrm{S}[s]$, that

$$\exists! x \mathrm{S}[x]$$

  then we permit ourselves to bestow a name on the unique set that satisfies that condition.
- *Example:* We already gave the name '$\emptyset$' to the unique set x satisfsying the condition 'x is empty'.

Defining Functional Names (1/3)

- Often we can prove that for any set y, there exists a unique set x satisfying some condition $\mathrm{S}[x, y]$.
- In such cases, we permit ourselves to introduce a **functional name**, a scheme which, for each y, provides a name for the unique set x such that $\mathrm{S}[x, y]$.
- To make an analogy with real life: obviously everybody has a mother, so we can use the functional name y**'s mom** to refer to the unique individual x such that x is a mother of y, no matter who y is.

Defining Functional Names (2/3)

- *Example:* It is easy to prove that for any set y, there is a unique set x such that y is the only member of x.

  This justifies introducing the functional name **singleton**(y), abbreviated $\{y\}$.
- *Example:* Likewise, we introduce the functional name **successor**(y), abbreviated $s(y)$ which, for each set y, names the unique set x that satisfies the binary condition $x = y \cup \{y\}$.

Defining Functional Names (3/3)

- This practice extends to names that depend on more than one variable. To take anotherreal-life example, we might introduce the functional name x**'s seniority over** y.

  For any two individuals x and y this is defined to be the number of days (rounded off) from x's birthdate to y's birthdate (this is a negative integer if y's birthdate precedes x's).

- In general: if, for some positive natural number n and some $(n+1)$-ary condition $\mathrm{S}[x_0, \ldots x_n]$ we can prove

  $$\forall x_1 \ldots \forall x_n \exists! x_0 \mathrm{S}[x_0, \ldots x_n]$$

  then we can introduce a functional name $\mathbf{name}(x_1, \ldots, x_n)$ which, for each choice of values for the n variables $x_1, \ldots, x_n$ provides a name for the unique set which satisfies the condition for that choice of values.