Crash Introduction to Quantifiers

Margaret M. Fleck

29 January 2010

This lecture finishes the quick introduction to quantifiers, which follows part of section 1.3 of Rosen. We'll also see some examples of direct proof, from section 1.6 of Rosen. Don't worry about reading either section right now. We'll do that when we're closer to covering their full content.

1 Quick recap of quantifiers

Last class, we saw examples of statements with quantifiers. We saw three quantifiers: universal "for all" ($\forall$), existential "there exists" ($\exists$), and unique existence "there is exactly one" ($\exists!$). A sample universally quantified statement might be

$$\forall x \in \mathbb{R}, x^2 + 3 \geq 0$$

This contains the quantifier itself $\forall$. It "binds" a variable x, which is restricted to come from the replacement set $\mathbb{R}$. Or, in other words, x has to be a real number. And then we have a statement $x^2 + 3 \geq 0$ that uses the variable x. So we are making a claim that this statement holds no matter what real value you substitute for x.

The "bound" variable in a quantification is only defined until the end of the quantified statement. That's usually the end of the sentence, or the end of the line, but you have to use common sense about what the author intended. The text for which the definition is in use is called the "scope" of this variable. This is just like the dummy variables ued in summations and integrals, e.g. the i in $\sum_{i=0}^{n} \frac{1}{i}$ is only defined while you are still inside the summation.

Each variable in computer languages have a "scope" over which its definition remains valid. This is sometimes the whole program (global variables) or an entire code file, but more often just within an individual function/procedure/method. If you try to use a variable outside the scope of its definition, you'll get a compiler error. Similarly, the "scope" of a quantifier in mathematics is the equations or text during which the binding is supposed to be valid. Bindings normally persist from the quantifier until the end of the sentence or the end of the line. Normally, the intended scope is obvious. When it isn't, writers sometimes use parentheses to make it clear.

When writing mathematics, variables have to be defined, just like variables in computer programs. Some variables are bound by quantifiers. Others are set to specific values (e.g. suppose that x is 3.1415). But you shouldn't just start using a variable without making sure the reader knows why it has been introduce and what type of value it contains.

If a variable hasn't been bound by a quantifier, or otherwise given a value or a set of replacement values, it is called "free." Statements containing free variables don't have a defined truth value, which means they are of somewhat limited use in writing proofs.

2 Notational stupidity

There's several conventions about inserting commas after the quantifier and/or parentheses around the following predicate. The style in these notes and the style in Rosen are both ok to copy from.

If you want to state a claim about two numbers, you can use two quantifiers as in:

$$\forall x \in \mathbb{R}, \forall y \in \mathbb{R}, x + y \geq x$$

This is usually abbreviated to

$$\forall x, y \in \mathbb{R}, x + y \geq x$$

This means "for all real numbers x and y, $x + y \geq x$" (which isn't true).

In such a claim, the two variables x and y might contain different values, but it's important to realize that they might also be equal.

We can do the same thing with existential quantifiers. Suppose that S is the set of all ASCII strings (aka strings of letters, numbers, and other standard keyboard symbols). Then the following states that there are two strings which are reversed versions of each other, e.g. x might be "UIUC" and y might be "CUIU".

$$\exists x, y \in S, x = \text{reverse}(y)$$

We could also say

$$\exists x \in S, x = \text{reverse}(x)$$

This is a much stronger claim. To prove it, we must find a string that is a reverse of itself, such as "ABBA."

3 Notation for 2D points

When manipulating 2D points, you have several options. You can write something like $\forall x, y \in \mathbb{Z}$ ("for any integers x and y") and then later refer to the ordered pair (x, y).

Or, people also write $\forall (x, y) \in \mathbb{Z}^2$ ("For any point (x, y) with integer coordinates"). For example, the following says that there is a point on the unit circle:

$$\exists (x, y) \in \mathbb{R}^2, x^2 + y^2 = 1$$

Or you can write something like

$$\exists p \in \mathbb{R}^2, p \text{ is on the unit circle}$$

Then, when you later need to make precise what it means to be "on the unit circle," you have to break up p into its two coordinates. So you say that that p has the form (x, y), where x and y are real numbers. And then assert that $x^2 + y^2 = 1$.

4 Negating statements with quantifiers

Suppose we have a universal claim like $\forall x \in \mathbb{R}, x^2 \geq 0$. This claim will be false if there is at least one real number x such that $x^2 < 0$. In general, a statement of the form "for all x in A, $P(x)$" is false exactly when there is some value x in A such that $P(x)$ is false. In other words, when "there exists x in A such that $P(x)$ is not true". In shorthand notation:

$$\neg(\forall x, P(x)) \equiv \exists x, \neg P(x)$$

Similarly,

$$\neg(\exists x, P(x)) \equiv \forall x, \neg P(x)$$

So this is a bit like the de Morgan's laws: when you move the negation across the operator, you change it to the other similar operator.

Last class, we saw how to move negation operators from the outside to the inside of expressions involving $\wedge$, $\vee$, and the other propositional operators. Together with these two new rules to handle quantifiers, we now have a mechanical procedure for working out the negation of any random statement in predicate logic.

So if we have something like

$$\forall x, P(x) \rightarrow (Q(x) \wedge R(x))$$

Its negation is

$$\begin{aligned}
\neg(\forall x, P(x) \rightarrow (Q(x) \wedge R(x))) &\equiv \exists x, \neg(P(x) \rightarrow (Q(x) \wedge R(x))) \\
&\equiv \exists x, P(x) \wedge \neg(Q(x) \wedge R(x))) \\
&\equiv \exists x, P(x) \wedge (\neg Q(x) \vee \neg R(x))
\end{aligned}$$

5 Proving a universal statement

Now, let's consider how to prove a claim like

> For every rational number q, $2q$ is rational.

First, we need to define what we mean by "rational".

> A real number r is *rational* if there are integers m and n, $n \neq 0$, such that $r = \frac{m}{n}$.

The simplest technique for proving a claim of the form $\forall x \in A, P(x)$ is to pick some representative value for x.[1]. Think about sticking your hand into the set A with your eyes closed and pulling out some random element. You use the fact that x is an element of A to show that $P(x)$ is true. Here's what it looks like for our example:

> Proof: Let q be any rational number. From the definition of "rational," we know that $q = \frac{m}{n}$ where m and n are integers and n is not zero. So $2q = 2\frac{m}{n} = \frac{2m}{n}$. Since m is an integer, so is $2m$. So $2q$ is also the ratio of two integers and, therefore, $2q$ is rational.

At the start of the proof, notice that we expanded the word "rational" into what its definition said. At the end of the proof, we went the other way: noticed that something had the form required by the definition and then asserted that it must be a rational.

Notice also that we spelled out the definition of "rational" but we just freely used facts from high school algebra as if they were obvious. In general, when writing proofs, you and your reader come to some agreement about what parts of math will be considered familiar and obvious, and which require explicit discussion. For this course, basic facts from high school math courses will be considered obvious.

6 A couple notational issues

WARNING!! Abuse of notation. Notice that our definitions said "if". If you take them literally as read, that means you can only use them in one direction. This isn't what's meant. Definitions are always intended to work in both directions, i.e. I meant to say "if and only if." This little misuse of "if" in definitions is very, very common.

Also notice that "iff" is shorthand for "if and only if", which is the same thing as $\leftrightarrow$.

7 Proving existential statements

Here's an existential claim:

[1] The formal name for this is "universal instantiation."

Claim 1 *There is an integer k such that $k^2 = 0$.*

An existential claim such as the following asserts the existence of an object with some set of properties. So it's enough to exhibit some specific concrete object, of our choosing, with the required properties. So our proof can be very simple:

Proof: Zero is such an integer. So the statement is true.

We could spell out a bit more detail, but it's really not necessary. Proofs of existential claims are often very short, though there are exceptions.

Notice one difference from our previous proofs. When we pick a value to instantiate a universally quantified variable, we have no control over exactly what the value is. We have to base our reasoning just on what set it belongs to. But when we are proving an existential claim, we get to pick our own favorite choice of concrete value, in this case zero.

8 Overview of proof methods

Next lecture, we'll see more examples of proofs, especially how to disprove existential and universal claims. Looking ahead a bit, suppose that we want to disprove a claim about all reals. This requires finding (at least) one real for which the claim fails. This is called a counter-example. If you need to prove that there can't exist a number with certain properties, you need to make a general argument that works for all numbers.

So, our general pattern for selecting the proof type is:

	prove	disprove
universal	general argument	specific counter-example
existential	specific example	general argument

Both types of proof start off by picking an element x from the domain of the quantification. However, for the general arguments, x is a random element whose identity you don't know. For the proofs requiring specific examples, you can pick x to be your favorite specific concrete value.