Logic and Proofs

Goals:

- Build familiarity (or review) the structure of mathematical assertions, including:
  - propositional connectives
  - quantifiers
- Understand how the structure of an assertion can be a guide to finding a proof.

You presumably have had some introduction to mathematical logic in CSE 215, but I have a perspective on this that might not have been communicated well in that course.

Mathematical Objects and Assertions

The activity of mathematics involves formulating *assertions* about mathematical *objects*, and constructing *proofs* to demonstrate that the assertions are true.

- It is useful to imagine the existence of a "mathematical universe", which we assume to obey certain basic rules.
- The goal of mathematical study is to explore the objects in this universe and how they relate to each other.
- This is done by proving assertions about the objects, starting from *axioms* and using valid *inference rules*.

An assertion whose truth has been demonstrated by a proof is called a *theorem*.

Classification of Theorems

Mathematical writings (depending on their authors' tastes) often use names to classify theorems, *e.g.*:

Theorem: A major result that is of inherent interest.

Proposition: Often, a theorem that is cited without proof, or perhaps just a minor theorem.

Lemma: A result primarily used in proving other theorems.

Conjecture: An assertion that has not been proved, but which the author has good reason to believe is true.

There are others. One mathematician that I have read uses the term "Scholium" to refer to a theorem of minor importance.

Expressing Mathematical Assertions

Mathematical assertions are expressed using:

- *Predicates*, which are true/false statements about mathematical objects; *e.g.* "3 is an even number".
- *Propositional connectives:* $\vee$, $\wedge$, $\neg$, *etc.*
- *Quantifiers:* $\forall$ ("for all"), $\exists$ ("there exists")
- *Constants:* names with a fixed meaning; *e.g.* 3, π.
- *Variables:* names whose meaning depends on the context; *e.g.* x.

The use of variables (*esp.* with quantifiers) has some subtle aspects, which are a common source of confusion.

Variables and Predicates

Predicates express properties of or relationships between mathematical objects. *Variables* are used to represent the unspecified objects.

- $x \neq 0$: expresses a true/false statement about "x" (whatever it may be). Constant 0 has a fixed meaning.
- $x > y$: expresses a true/false relationship between "x" and "y" (whatever they may be).
- $1 + 1 = 2$: expresses an assertion whose truth or falsity does not depend on any variables (*i.e.* it is a *proposition*).

To say whether a predicate is true or false, we have to specify what the variables (if any) stand for.

Constants and Definitions

Constants are symbols that have been given a fixed meaning. Usually this is done using *definitions*.

- An *explicit definition* introduces a constant as a new (short) name for an object that is already denoted by some expression; *e.g.*

  "Define a *googol* to be 10^{100}."

  The constant being defined must not occur in the defining expression.

- An *implicit definition* defines something in terms of a property that it has; *e.g.*

  > "Define $\sqrt{3}$ to be the real number x such that $x^2 = 3$"

  We are required to prove that a unique such x exists.

- In a *recursive definition* the entity being defined also appears in the defining expression.

  > Define f by:
  >
  > $$f(n) = \begin{cases} 0, & \texttt{if } n = 0 \\ n + f(n-1), & \texttt{if } n > 0 \end{cases}$$

  In general, such "circular" definitions are meaningless unless we have shown existence and uniqueness.

Propositional Connectives

The *propositional connectives* allow us to combine assertions into compound assertions; *e.g.*

- $x \neq 3$ (it is not the case that $x = 3$)
- $x > 0 \wedge x < 1$ ($x > 0$ *and* $x < 1$)
- x *is even* $\leftrightarrow x$ *mod* $2 = 0$. *("if and only if")*
- *etc.* (I assume these are very familiar.)

The truth value of the compound assertion is determined by the truth values of the component assertions and the *truth tables* for the connectives.

Quantifiers

A *quantifier* is applied to an assertion and produces an assertion that depends on one fewer variables:

- $\forall x.\ x^2 \neq y$ ("for all x, x^2 is not equal to y")
  an assertion about y alone.
- $\exists x.\ x^2 = y$ ("there exists x such that x^2 equals y")
  also an assertion about y alone.

The truth value of a quantified statement $\forall x.\ P(x, y)$ or $\exists x.\ P(x, y)$ depends on the value of y, but not on a particular value for x.

Quantifiers as Binders

A quantified variable can be systematically replaced by any desired variable (but not one already in use), without changing the meaning of the assertion:

- $\exists x.\ x^2 = y$ is equivalent to $\exists z.\ z^2 = y$.
- $\exists x.\ x^2 = y$ is not equivalent to $\exists y.\ y^2 = y$.

Notations with this kind of behavior are called *binders*. Other examples:

$$\int f(x)dx \qquad\qquad \{\text{int } x = 0;\ \text{return } x + 1;\}$$

Proofs

Proofs are constructed using *axioms* and *inference rules*:

- *Axioms* are basic assertions that are assumed to be *true*.

$$\forall A\ B.\ A = B \leftrightarrow (\forall X.\ X \in A \leftrightarrow X \in B)$$

- *Inference rules* are *valid* (*i.e.* truth-preserving) ways of drawing a *conclusion* from *premises*:

$$\frac{P \to Q \qquad P}{Q} \qquad\qquad \frac{P}{P \vee Q}$$

A *proof* can be defined to be a certain kind of tree with axioms at the leaves and instances of inference rules at the nodes. The *conclusion* is the assertion at the root.

“Fitch-style” Proofs

Fitch-style notation is a practical way of writing structured proofs.

- A application of an inference rule is written like this:

$$
\begin{array}{|ll}
P \to Q & \\
P & \\
Q & \qquad\qquad\qquad\qquad \to\mathbf{E}
\end{array}
$$

(from $P \to Q$ and P, infer Q)

- “Stacking” the premises vertically makes it possible to write large nested proofs conveniently, as we shall see.

Introduction and Elimination Rules

With each logical connective we can associate:

- *introduction rules*, which tell how to *introduce* that connective in a conclusion; *e.g.*

$$\begin{array}{|ll} P & \\ Q & \\ P \wedge Q & \qquad\qquad \wedge\mathbf{I} \end{array}$$

(conjunction introduction)

- *elimination rules*, which tell how to *use* that connective appearing in premises; *e.g.*

$$\begin{array}{|ll} P \wedge Q & \\ P & \qquad\qquad \wedge\mathbf{E}1 \end{array} \qquad\qquad \begin{array}{|ll} P \wedge Q & \\ Q & \qquad\qquad \wedge\mathbf{E}2 \end{array}$$

(conjunction elimination)

Reading Inference Rules

An inference rule

$$\begin{array}{|ll}
P & \\
Q & \\
P \land Q & \qquad\qquad\qquad \land\mathbf{I}
\end{array}$$

can be read in two ways:

- "If we have already obtained a proof of P and of Q, then we can combine them to form a a proof of $P \land Q$."
- "If our goal is to find a proof $P \land Q$, then we can do it by finding a proof of P and a proof of Q."

The second reading shows how the structure of a statement to be proved can guide the search for a proof.

Disjunction Introduction

$$\begin{array}{|ll} P & \\ P \vee Q & \vee\mathbf{I}1 \end{array}$$

$$\begin{array}{|ll} Q & \\ P \vee Q & \vee\mathbf{I}2 \end{array}$$

To prove $P \vee Q$, we may either prove P or prove Q.

We will consider elimination rules for disjunction shortly.

Rules for Implication

The elimination rule for implication is also called *modus ponens*:

$$
\begin{array}{|ll}
P \to Q & \\
P & \\
Q & \to\mathbf{E}
\end{array}
$$

The introduction rule looks slightly more complicated:

$$
\begin{array}{|lll}
\begin{array}{|l} \underline{P} \\ \vdots \\ Q \end{array} & & \begin{array}{l} \text{"assume } P\text{"} \\ \\ \text{"show } Q\text{"} \end{array} \\
P \to Q & \to\mathbf{I} & \text{"thus } P \to Q\text{"}
\end{array}
$$

To prove $P \to Q$, we may prove Q under assumption P.

Implication Introduction: cont'd

$\quad P$		"assume P"
$\quad \vdots$		
$\quad Q$		"show Q"
$P \rightarrow Q$	$\rightarrow\mathbf{I}$	"thus $P \rightarrow Q$"

- Assumption P is *local* to the inner proof scope.
- Assumption P *may not* be used outside that scope.
- We say that P is *discharged* upon leaving the scope.

In general, informal mathematical proofs involve frequent (often implicit) introduction and discharge of assumptions.

Disjunction Elimination

This rule involves *two* nested proofs:

$$
\begin{array}{|l}
Q \vee R \\
\begin{array}{|l}
Q \\ \hline
\vdots \\
P
\end{array} \\
\\
\begin{array}{|l}
R \\ \hline
\vdots \\
P
\end{array} \\
\\
P \qquad\qquad\qquad\qquad\qquad\qquad \vee\mathbf{E}
\end{array}
$$

The inner blocks amount to proofs of $Q \to P$ and $R \to P$.

This is proof by case analysis!

Using Assumptions in Inner Blocks

Although an assumption cannot be used outside of the block in which it is assumed, it can be "reiterated" in an inner block:

$$\begin{array}{lll}
 & 1.\ P & \\
 & \quad 2.\ Q & \\
 & \quad 3.\ P & \textbf{R:1} \\
 & 4.\ Q \to P & \to\textbf{I:2–3} \\
5.\ P \to (Q \to P) & & \to\textbf{I:1–4}
\end{array}$$

The line numbers permit precise justifications.

Rules for Biconditional

$$
\begin{array}{l|l}
\quad \begin{array}{|l} P \\ \hline \vdots \\ Q \end{array} & \\
& \\
\quad \begin{array}{|l} Q \\ \hline \vdots \\ P \end{array} & \\
& \\
P \leftrightarrow Q & \leftrightarrow \mathbf{I}
\end{array}
$$

$$
\begin{array}{|ll}
P \leftrightarrow Q & \\
P & \\
Q & \leftrightarrow \mathbf{E}1
\end{array}
$$

$$
\begin{array}{|ll}
P \leftrightarrow Q & \\
Q & \\
P & \leftrightarrow \mathbf{E}2
\end{array}
$$

These are consequences of the equivalence of $P \leftrightarrow Q$ and $(P \to Q) \wedge (Q \to P)$.

Rules for Negation

$$
\begin{array}{|ll}
\begin{array}{|l} P \\ \hline \vdots \\ Q \end{array} & \\
 & \\
\begin{array}{|l} P \\ \hline \vdots \\ \neg Q \end{array} & \\
\neg P & \qquad \neg\,\mathbf{I}
\end{array}
\qquad\qquad
\begin{array}{|ll}
\neg\neg P & \\
P & \qquad \neg\,\mathbf{E}
\end{array}
$$

Rule ($\neg\,\mathbf{I}$) expresses "Proof by Contradiction": if from assumption P we can prove both Q and $\neg Q$, then we may conclude $\neg P$.

Rules for Negation (cont'd.)

There are other formulations of rules for negation. If $\perp$ stands for “false”, then the following are valid:

$$
\begin{array}{|ll}
\begin{array}{|l} P \\ \hline \vdots \\ \perp \end{array} & \\
\neg P & \qquad \neg\,\mathbf{I}'
\end{array}
\qquad\qquad
\begin{array}{|ll}
P & \\
\neg P & \\
\perp & \qquad \neg\,\mathbf{E}'
\end{array}
$$

$$
\begin{array}{|ll}
\begin{array}{|l} \neg P \\ \hline \vdots \\ \perp \end{array} & \\
P & \qquad \mathbf{IP}
\end{array}
\qquad\qquad
\begin{array}{|ll}
\perp & \\
P & \qquad \mathbf{X}
\end{array}
$$

Universal Instantiation

The elimination rule for $\forall$ is called *universal instantiation*:

$$\begin{array}{|ll} \forall x.\, P(x) & \\ P(t) & \qquad\qquad \forall\,\mathbf{E} \end{array}$$

If we have $\forall x.\, P(x)$ then we may conclude $P(t)$, where t is *any* term of our choosing (it may even involve variables).

Universal Generalization

The introduction rule for $\forall$ is called *universal generalization*:

$$
\begin{array}{l|l}
\begin{array}{|l}
y \text{ is "fresh"} \\
\hline
\vdots \\
P(y)
\end{array} & \\
\forall x.\, P(x) & \qquad\qquad \forall\mathbf{I}
\end{array}
$$

If we can prove $P(y)$ where y is a "fresh" variable that doesn't appear anywhere else, then we may conclude $\forall x.\, P(x)$.

In informal mathematical writing, the variable y often introduced by: "Let y be a fixed arbitrarily chosen value."

Existential Generalization

The introduction rule for $\exists$ is called *existential generalization*:

$$\begin{array}{|ll} P(t) & \\ \exists x.\ P(x) & \qquad\qquad \exists\,\mathbf{I} \end{array}$$

If we can prove $P(t)$ for some *term t of our choosing, then we may conclude $\exists x.\ P(x)$.*

The term t is sometimes called a *witness* to $\exists x.P(X)$.

Existential Instantiation

The elimination rule for $\exists$ is called *existential instantiation*:

$$
\begin{array}{|l}
\exists x.\ P(x) \\
\quad \begin{array}{|l}
P(c) \\ \hline
Q
\end{array} \\
Q \qquad\qquad\qquad\qquad\qquad\qquad \exists\mathbf{E}
\end{array}
$$

Essentially, we are saying: "Choose *some* c for which $P(c)$ holds."

Important: *Neither c nor assumption $P(c)$ can be used outside of the proof block that introduces them.*

Derived Rules

Additional rules can be viewed as abbreviations for proofs that can be done with the rules already stated. For example:

$\neg\neg P$		
P	**DNE**	

1. $\neg\neg P$		
2. $\neg P$		
3. $\perp$		$\neg$ **E:1,2**
4. P	**IP:** 2–3	

(double negation elimination)

A Bigger Example

1. $\exists x.\ P(x)$	
2. $\neg\neg\forall x.\ \neg P(x)$	
3. $P(c)$	
4. $\forall x.\ \neg P(x)$	**DNE** 2
5. $\neg P(c)$	$\forall$ **E:4**
6. $\bot$	$\neg$ **E:3,5**
7. $\bot$	$\exists$ **E:1, 3–6**
8. $\neg\forall x.\ \neg P(x)$	**IP** 2–7
9. $\exists x.\ P(x) \rightarrow \neg\forall x.\ \neg P(x)$	$\rightarrow$ **I:1–8**

Informal Prose

We wish to show $\exists x.\ P(x) \rightarrow \neg\forall x.\ \neg P(x)$. Assume $\exists x.\ P(x)$. We claim $\neg\forall x.\ \neg P(x)$. Suppose, for the purpose of obtaining a contradiction, that $\neg\neg\forall x.\ \neg P(x)$ holds. Using $\exists x.\ P(x)$ obtain c such that $P(c)$ holds. From $\neg\neg\forall x.\ \neg P(x)$ it follows that $\forall x.\ \neg P(x)$ holds (eliminating the double negation), and hence in particular $\neg P(c)$ holds (taking x to be c). But now we have shown, under assumption $\neg\neg\forall x.\ \neg P(x)$, that $P(c)$ and $\neg P(c)$ both hold. This is a contradiction, so we conclude $\neg\forall x.\ \neg P(x)$. Since we have now shown $\neg\forall x.\ \neg P(x)$ under the assumption $\exists x.\ P(x)$, we may finally conclude $\exists x.\ P(x) \rightarrow \neg\forall x.\ \neg P(x)$, completing the proof.

Note: *We don't randomly assume things. Every assumption is introduced for a reason, and is eventually discharged!*

Summary

The purposes of going through this are:

- To help understand the use of quantifiers in mathematical assertions.
- To help understand the structure of proofs in the "natural deduction" style that mathematicians use.
- To show how the structure of an assertion to be proved can serve as a guide to finding a proof.
  - Note that there are still choices to be made, so finding a proof is not automatic.

We won't be doing lots of proofs at this level, but understanding "the rules of the game" can help sort out confusion.