Introduction to Formal Logic

From Everyday Reasoning to Symbolic Systems

Brendan Shea, PhD

Intro to Logic

From Everyday Reasoning to Formal Logic

- We've spent the year studying how people **actually reason** in everyday life through informal logic.
- Today, we'll explore **formal logic**: a system that uses symbols and strict rules to analyze arguments.
- Think of formal logic as the mathematics of reasoning—it gives us precise tools to test whether arguments work.
- Just as algebra uses x and y instead of specific numbers, formal logic uses symbols instead of specific statements.

Key Insight

Formal logic strips away the content of arguments to focus purely on their structure.

What Makes Logic "Formal"?

- **Formal** means we follow exact rules, like a game with strict instructions that never change.
- We use **symbols** ($\wedge$, $\vee$, $\neg$, $\rightarrow$) instead of words like "and," "or," "not," and "if...then."
- The **validity** of an argument depends only on its form, not whether the statements are actually true.
- This approach eliminates ambiguity—each symbol has exactly one meaning, unlike words in natural language.

Example

In everyday language, "or" can be inclusive (pizza or salad or both) or exclusive (boy or girl). In formal logic, $\vee$ always means inclusive or.

Why Study Formal Logic?

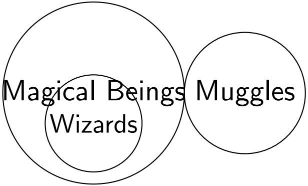
- Formal logic is the foundation of **computer programming**—every if-then statement in code uses logical principles.
- It helps us spot **invalid arguments** even when they sound convincing in everyday language.
- Mathematics and science rely on formal logic to prove theorems and test hypotheses rigorously.
- Understanding formal logic makes you a more **precise thinker** and better at constructing bulletproof arguments.

Three Systems We'll Explore

1. **Categorical Logic**: Deals with categories and membership
2. **Propositional Logic**: Connects whole statements
3. **Predicate Logic**: Analyzes internal structure of statements

Categories and Classes: The Building Blocks

- **Categorical logic** studies relationships between groups or classes of things.
- A **category** is any collection we can clearly define: wizards, muggles, Hogwarts students, magical creatures.
- We examine how categories relate through **inclusion** (all wizards are magical), **exclusion** (no muggles are wizards), and **overlap** (some students are wizards).
- Every statement in categorical logic makes a claim about the relationship between two categories.



The Four Standard Forms (A, E, I, O)

- **A-form (Universal Affirmative)**: "All S are P" — All Gryffindors are brave.
- **E-form (Universal Negative)**: "No S are P" — No Death Eaters are kind.
- **I-form (Particular Affirmative)**: "Some S are P" — Some wizards are Animagi.
- **O-form (Particular Negative)**: "Some S are not P" — Some students are not Quidditch players.

Key Terms

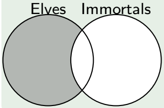
Universal	Refers to every member of a category
Particular	Refers to at least one member
Affirmative	States inclusion or membership
Negative	States exclusion or non-membership

Venn Diagrams: Visualizing Relationships

- **Venn diagrams** use overlapping circles to show relationships between categories visually.
- Shading indicates **empty regions** (no members exist there), while X marks show **existing members**.
- Each of our four forms (A, E, I, O) has a unique Venn diagram pattern.
- These diagrams help us test whether arguments are valid by checking if conclusions must follow from premises.

Example

"All Elves are Immortal" (A-form)



The left circle (Elves) outside the overlap is shaded, showing all Elves must be in the Immortal category.

Sample Argument: All Dogs Are Mammals

- Let's analyze a classic **syllogism** (an argument with two premises and a conclusion).
- **Premise 1**: All hobbits are Middle-earth dwellers.
- **Premise 2**: All Middle-earth dwellers are fictional characters.
- **Conclusion**: Therefore, Therefore, all hobbits are fictional characters.

Valid Argument Structure

This follows the pattern called **Barbara**: All A are B, All B are C, therefore All A are C. The conclusion must be true if both premises are true.

Practice: Translating Everyday Claims

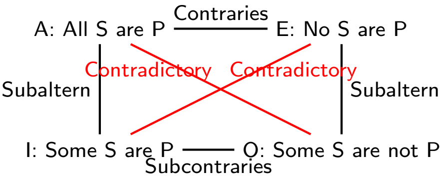
- Natural language is often ambiguous, so we must **standardize** statements into A, E, I, or O forms.
- "Jedi never use the dark side" becomes "**No Jedi are dark side users**" (E-form).
- "Not all heroes wear capes" becomes "**Some heroes are not cape-wearers**" (O-form).
- Watch for hidden quantifiers: "Dragons breathe fire" usually means "**All dragons** breathe fire" (A-form).

Translation Tips

"Only X are Y"	=	"All Y are X" (reversed!)
"X are always Y"	=	"All X are Y"
"There are X that Y"	=	"Some X are Y"
"Not every X is Y"	=	"Some X are not Y"

The Square of Opposition

- The **Square of Opposition** shows logical relationships between the four categorical forms.
- **Contradictories** (A-O, E-I) cannot both be true or both be false—exactly one must be true.
- **Contraries** (A-E) cannot both be true, but both can be false.
- **Subcontraries** (I-O) cannot both be false, but both can be true.



From Categories to Propositions

- While categorical logic focuses on relationships between groups, **propositional logic** works with complete statements.
- A **proposition** is any statement that can be true or false: "Harry defeated Voldemort," "It's raining," "2+2=4."
- Instead of analyzing internal structure, propositional logic examines how we **connect** statements using logical operators.
- This system is more flexible—we can analyze any argument made of statements, not just those about categories.

Key Shift

Categorical logic asks "What's inside statements?" while propositional logic asks "How do statements connect?"

Basic Connectives: AND, OR, NOT

- **Conjunction** ($\wedge$, AND): "Hermione is smart $\wedge$ Hermione is brave" — both parts must be true.
- **Disjunction** ($\vee$, OR): "Ron plays chess $\vee$ Ron plays Quidditch" — at least one part must be true.
- **Negation** ($\neg$, NOT): "$\neg$ Voldemort has a nose" — reverses the truth value.
- These **logical connectives** are the building blocks for creating complex propositions from simple ones.

Formal Notation

Connective	**Symbol**	**Example**
AND	$\wedge$	$p \wedge q$
OR	$\vee$	$p \vee q$
NOT	$\neg$	$\neg p$
IF...THEN	$\rightarrow$	$p \rightarrow q$
IF AND ONLY IF	$\leftrightarrow$	$p \leftrightarrow q$

Truth Tables: The Foundation

- A **truth table** shows all possible combinations of truth values for propositions and their results.
- Each row represents one possible scenario, and we calculate the truth value of complex expressions.
- Truth tables give us a mechanical way to determine when compound statements are true or false.
- They're like multiplication tables for logic—once you know them, you can solve any problem!

Example

Truth table for $p \wedge q$ (Frodo has the ring AND Sam helps him):

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

Sample Argument: If-Then Reasoning

- The **conditional** ($\rightarrow$) represents "if...then" statements: "If you're a wizard, then you can do magic."
- $p \rightarrow q$ is only false when p is true but q is false (you're a wizard but can't do magic).
- This captures cause-and-effect, rules, and logical consequences in a precise way.
- Be careful: "If p then q" does NOT mean "q only if p"—muggles might do magic too!

Common Confusion

"If you eat the poison apple, then you fall asleep" ($p \rightarrow q$)
This does NOT mean: "You fall asleep only if you eat the poison apple"
Sleeping pills could also make you fall asleep!

Complex Propositions: Building Bigger Claims

- We can combine multiple connectives to build **complex propositions** that capture sophisticated ideas.
- Use parentheses to show order of operations: $(p \wedge q) \rightarrow r$ differs from $p \wedge (q \rightarrow r)$.
- Each added connective doubles the rows in our truth table—complexity grows exponentially!
- Real arguments often involve many connected statements working together.

Example

"If Gandalf arrives and the weather is good, then we'll defeat Sauron or retreat to safety"

$$\text{Let } g = \text{Gandalf arrives, } w = \text{weather is good,}$$
$$d = \text{defeat Sauron, } r = \text{retreat to safety}$$
$$\text{Formula: } (g \wedge w) \rightarrow (d \vee r)$$

This has $2^4 = 16$ rows in its truth table!

Valid vs. Invalid Categorical Arguments

- A **valid** argument guarantees that if the premises are true, the conclusion must be true.
- An **invalid** argument can have true premises but a false conclusion—the logic doesn't work.
- We test validity by checking whether the conclusion's Venn diagram is already contained in the premises' diagrams.
- Remember: validity is about structure, not truth—an argument can be valid with false premises!

Example

Invalid Argument:

1. All Time Lords have two hearts. (True in Doctor Who universe)
2. The Doctor has two hearts. (True)
3. Therefore, the Doctor is a Time Lord. (Doesn't follow!)

This commits the fallacy of **affirming the consequent**—other beings might also have two hearts.

Practice: Translating Natural Language

- Natural language hides logical structure, so we must identify the **atomic propositions** and **connectives**.
- "You can't apparate in Hogwarts" becomes $\neg p$ where $p =$ "You can apparate in Hogwarts."
- "Either die a hero or live long enough to become the villain" becomes $h \vee v$ (inclusive or!).
- "I'll help if and only if you're truly sorry" becomes $h \leftrightarrow s$ (biconditional).

Translation Strategy

1. Identify each distinct claim that can be true/false
2. Assign a letter to each atomic proposition
3. Find the logical connectives (and, or, not, if...then)
4. Build the formula using proper symbols

Common Valid Forms: Modus Ponens

- **Modus Ponens** ("method of affirming") is a fundamental valid argument form.
- Structure: If p then q; p is true; therefore q is true.
- This pattern appears everywhere in mathematical proofs, computer programs, and daily reasoning.
- It's so basic that denying it would make logical reasoning impossible!

Example

Modus Ponens in Action:

1. If you speak Parseltongue, then you can open the Chamber. ($p \to q$)
2. Harry speaks Parseltongue. (p)
3. Therefore, Harry can open the Chamber. (q)

Truth table confirms: whenever both premises are true, conclusion must be true.

Common Invalid Forms: Affirming the Consequent

- **Affirming the Consequent** looks similar to Modus Ponens but is invalid.
- Structure: If p then q; q is true; therefore p is true. (WRONG!)
- This fallacy assumes there's only one way for q to be true, ignoring other possibilities.
- It's tempting because it seems to work backwards from effects to causes.

Invalid Reasoning

1. If you're a Death Eater, then you have the Dark Mark. ($p \rightarrow q$)
2. Karkaroff has the Dark Mark. (q)
3. Therefore, Karkaroff is a Death Eater. (p) — INVALID!

Why invalid? Former Death Eaters keep their marks!

Using Truth Tables to Test Validity

- To test validity, we build a truth table with columns for all premises and the conclusion.
- An argument is **valid** if every row where ALL premises are true also has a true conclusion.
- If even one row has true premises but false conclusion, the argument is **invalid**.
- This method is foolproof but can be tedious for complex arguments with many propositions.

Example

Testing: $p \to q, \neg q \therefore \neg p$ (Modus Tollens)

p	q	$p \to q$	$\neg q$	$\neg p$	Valid?
T	T	T	F	F	—
T	F	F	T	F	—
F	T	T	F	T	—
F	F	T	T	T	✓

The Limits of Propositional Logic

- Propositional logic treats "Socrates is mortal" as a single, indivisible unit p.
- It can't express the connection between "Socrates is mortal" and "Plato is mortal"—they're just unrelated p and q.
- We lose important patterns: "All X are Y" requires a different proposition for each X!
- Mathematical statements like "For every number n, there exists $n + 1$" are impossible to express properly.

The Problem

In propositional logic, these require different symbols:

- p: Harry is a wizard
- q: Ron is a wizard
- r: Hermione is a wizard

We can't capture that they all share the property "is a wizard"!

Why We Need More Power

- Real reasoning often involves **generalizations** about entire groups or categories.
- We need to express **relationships** and **properties** that apply to many individuals.
- Mathematical proofs require statements about "all numbers" or "there exists a solution."
- **Predicate logic** provides tools to look inside propositions and work with their structure.

What Predicate Logic Adds

1. **Predicates**: Properties or relations (like "is red" or "is taller than")
2. **Variables**: Stand-ins for individuals (like x, y in algebra)
3. **Quantifiers**: "For all" ($\forall$) and "There exists" ($\exists$)
4. **Domain**: The universe of things we're talking about

Introducing Predicates and Subjects

- A **predicate** is a property or relation that can be true or false of individuals.
- **Subjects** are the specific individuals we're talking about (Harry, the number 7, this pencil).
- We write $P(a)$ to mean "individual a has property P"—like $Wizard(Harry)$ for "Harry is a wizard."
- Relations between multiple subjects use multiple places: $Loves(Romeo, Juliet)$ means "Romeo loves Juliet."

Example

Breaking down "Gandalf is wise and helps Frodo":

- Predicates: $Wise(x) =$ "x is wise", $Helps(x, y) =$ "x helps y"
- Subjects: $g =$ Gandalf, $f =$ Frodo
- Formula: $Wise(g) \wedge Helps(g, f)$

From "Socrates is mortal" to Mx

- The classic statement "Socrates is mortal" becomes $Mortal(socrates)$ or Ms for short.
- We can now express patterns: $Mortal(socrates)$, $Mortal(plato)$, $Mortal(aristotle)$ all use the same predicate!
- **Predicate symbols** (usually capital letters) represent properties: M, P, Q.
- **Constant symbols** (usually lowercase) represent specific individuals: a, b, c.

Building Our Vocabulary

$Jedi(x)$	x is a Jedi
$Teaches(x, y)$	x teaches y
$Stronger(x, y)$	x is stronger than y
$yoda$, $luke$, $obiwan$	Individual constants

Variables: The x in Predicate Logic

- **Variables** (usually x, y, z) are placeholders that can represent any individual in our domain.
- Just like in algebra where x can be any number, in logic x can be any person, object, or entity.
- $P(x)$ is not a complete statement—it's a **formula with a free variable** that becomes true/false when we specify x.
- Variables let us express general patterns before we commit to specific individuals.

Example

The formula $Wizard(x) \wedge Teaches(x, harry)$ means "x is a wizard and x teaches Harry"

- If $x =$ Dumbledore: TRUE (he's a wizard who teaches Harry)
- If $x =$ Snape: TRUE (also a wizard teacher)
- If $x =$ Hagrid: FALSE (teaches Harry but isn't a wizard)

The Universal Quantifier: $\forall$ (For All)

- The **universal quantifier** $\forall$ means "for all" or "for every" individual in our domain.
- $\forall x P(x)$ reads as "For all x, $P(x)$ is true" or "Everything has property P."
- This captures universal claims like "All ravens are black" as $\forall x(Raven(x) \rightarrow Black(x))$.
- The quantifier **binds** the variable—$\forall x P(x)$ is a complete statement that's either true or false.

Important Pattern

Universal statements usually use implication:

- "All wizards can do magic" $= \forall x(Wizard(x) \rightarrow CanDoMagic(x))$
- NOT: $\forall x(Wizard(x) \wedge CanDoMagic(x))$ — this says everything is a wizard!

The Existential Quantifier: $\exists$ (There Exists)

- The **existential quantifier** $\exists$ means "there exists at least one" or "for some."
- $\exists x P(x)$ reads as "There exists an x such that $P(x)$ is true" or "Something has property P."
- This captures claims like "Some dragons are friendly" as $\exists x(Dragon(x) \wedge Friendly(x))$.
- To make an existential statement true, we only need to find ONE example that works.

Contrast with Universal

Universal	**Existential**
$\forall x(Elf(x) \to Immortal(x))$	$\exists x(Elf(x) \wedge Mortal(x))$
"All elves are immortal"	"Some elf is mortal"
Uses implication ($\to$)	Uses conjunction ($\wedge$)
False if ONE counterexample	True if ONE example exists

Sample Translation: "All Students Study"

- English: "All students study" seems simple but hides logical complexity.
- First identify predicates: $Student(x)$ = "x is a student", $Studies(x)$ = "x studies."
- The logical form is: $\forall x(Student(x) \rightarrow Studies(x))$.
- Read carefully: "For any individual x, IF x is a student, THEN x studies."

Example

Let's check if our translation works correctly:

- Hermione (student who studies): $Student(h) \rightarrow Studies(h)$ is $T \rightarrow T = T$ ✓
- Dobby (non-student): $Student(d) \rightarrow Studies(d)$ is $F \rightarrow ? = T$ ✓
- Ron (student who doesn't study): $Student(r) \rightarrow Studies(r)$ is $T \rightarrow F = F$ ×

The formula is false only when we find a student who doesn't study!

Sample Translation: "Some Books Are Interesting"

- English: "Some books are interesting" claims at least one interesting book exists.
- Predicates: $Book(x)$ = "x is a book", $Interesting(x)$ = "x is interesting."
- The logical form is: $\exists x(Book(x) \land Interesting(x))$.
- Read as: "There exists an x such that x is a book AND x is interesting."

Why Conjunction, Not Implication?

Compare these formulas:

- $\exists x(Book(x) \land Interesting(x))$ — "Some book is interesting" ✓
- $\exists x(Book(x) \rightarrow Interesting(x))$ — "There's something such that IF it's a book THEN it's interesting"

The second is true whenever ANY non-book exists (like a pencil)!

Combining Quantifiers and Connectives

- We can mix quantifiers with all our logical connectives for complex statements.
- "No Death Eaters are kind" becomes $\forall x(DeathEater(x) \rightarrow \neg Kind(x))$.
- "Only wizards can see Thestrals" becomes $\forall x(SeesThestrals(x) \rightarrow Wizard(x))$.
- Parentheses matter: $\forall x(P(x) \wedge Q(x))$ differs from $\forall x P(x) \wedge \forall x Q(x)$!

Example

"Not all that glitters is gold" translation:

1. First attempt: $\neg\forall x(Glitters(x) \rightarrow Gold(x))$
2. Equivalent: $\exists x(Glitters(x) \wedge \neg Gold(x))$
3. Read as: "Something glitters but isn't gold"

Both forms say the same thing using logical equivalences!

Multiple Quantifiers: Order Matters!

- When using multiple quantifiers, their **order changes the meaning** dramatically.
- $\forall x \exists y Loves(x, y)$ means "Everyone loves someone" (each person might love someone different).
- $\exists y \forall x Loves(x, y)$ means "There's someone whom everyone loves" (one universally loved person).
- Read quantifiers from left to right, with each one setting up context for the next.

The Difference Illustrated

$\forall x \exists y Teaches(x, y)$	Every teacher has (at least) one student
$\exists y \forall x Teaches(x, y)$	One super-student is taught by everyone
$\exists x \forall y Teaches(x, y)$	One super-teacher teaches everyone
$\forall y \exists x Teaches(x, y)$	Every student has (at least) one teacher

Practice: "Everyone Loves Someone"

- This classic example shows why quantifier order is crucial for correct translation.
- Ambiguous English: "Everyone loves someone" has two possible meanings!
- **Reading 1**: Each person loves at least one person (maybe different for each).
- **Reading 2**: There's one special person whom everyone loves.

Example

The two translations:

1. $\forall x \exists y Loves(x, y)$ — "For each x, there exists a y that x loves"
   - Harry loves Ginny, Ron loves Hermione, Snape loves Lily...
2. $\exists y \forall x Loves(x, y)$ — "There exists a y such that every x loves y"
   - Everyone loves Dumbledore (or pizza, or baby Yoda...)

English is ambiguous; predicate logic is precise!

Common Translation Pitfalls

- **Pitfall 1**: Using $\wedge$ instead of $\rightarrow$ with universal quantifiers.
- **Pitfall 2**: Using $\rightarrow$ instead of $\wedge$ with existential quantifiers.
- **Pitfall 3**: Forgetting that "only" reverses the direction of implication.
- **Pitfall 4**: Misplacing negations—$\neg\forall x$ is very different from $\forall x\neg$!

Common Mistakes

Wrong	**Right**
"All cats meow": $\forall x(Cat(x) \wedge Meows(x))$	$\forall x(Cat(x) \rightarrow Meows(x))$
"Some cats meow": $\exists x(Cat(x) \rightarrow Meows(x))$	$\exists x(Cat(x) \wedge Meows(x))$
"Only cats meow": $\forall x(Cat(x) \rightarrow Meows(x))$	$\forall x(Meows(x) \rightarrow Cat(x))$
"Not all cats meow": $\forall x\neg(Cat(x) \rightarrow Meows(x))$	$\neg\forall x(Cat(x) \rightarrow Meows(x))$

The Domain of Discourse

- The **domain of discourse** is the set of all individuals our variables can represent.
- Different domains can make the same formula true or false—context matters!
- Common domains: all people, all numbers, all living things, all objects in Middle-earth.
- Sometimes we restrict domains to simplify formulas: "All are brave" vs. "All hobbits are brave."

Example

Consider $\forall x Magical(x)$ ("Everything is magical")

- Domain = {Harry, Hermione, Dumbledore}: TRUE ✓
- Domain = {Harry, Hermione, Uncle Vernon}: FALSE ×
- Domain = All characters in Harry Potter: FALSE ×

The same formula, different truth values based on domain!

Sample Argument: Socrates Revisited

- Now we can properly express the classic Socrates syllogism in predicate logic.
- **Premise 1**: All humans are mortal — $\forall x(Human(x) \rightarrow Mortal(x))$
- **Premise 2**: Socrates is human — $Human(socrates)$
- **Conclusion**: Socrates is mortal — $Mortal(socrates)$

Why This Works

1. From Premise 1, we know: $Human(socrates) \rightarrow Mortal(socrates)$
2. From Premise 2, we know: $Human(socrates)$ is true
3. By Modus Ponens: If $p \rightarrow q$ and p, then q
4. Therefore: $Mortal(socrates)$ must be true!

This is called **Universal Instantiation** followed by Modus Ponens.

From English to Symbols: Step by Step

- **Step 1**: Identify the domain (what are we talking about?).
- **Step 2**: List needed predicates and assign letters (keep a dictionary!).
- **Step 3**: Identify logical structure (universal/existential, and/or/not, if/then).
- **Step 4**: Build the formula piece by piece, checking each part makes sense.

Example

Translating: "Every wizard has a wand that chooses them"

1. Domain: All people and objects in the wizarding world
2. Predicates: $Wizard(x)$, $Wand(y)$, $Chooses(y, x)$
3. Structure: Universal (every wizard) + Existential (has a wand)
4. Formula: $\forall x(Wizard(x) \rightarrow \exists y(Wand(y) \wedge Chooses(y, x)))$

Practice: "No Reptiles Are Warm-Blooded"

- This negative universal statement requires careful translation to avoid errors.
- Predicates: $Reptile(x)$ = "x is a reptile", $WarmBlooded(x)$ = "x is warm-blooded"
- The correct translation: $\forall x(Reptile(x) \rightarrow \neg WarmBlooded(x))$
- Alternative equivalent form: $\neg \exists x(Reptile(x) \wedge WarmBlooded(x))$

Understanding the Forms

$\forall x(Reptile(x) \rightarrow \neg WarmBlooded(x))$	"All reptiles are not warm-blooded"
$\neg \exists x(Reptile(x) \wedge WarmBlooded(x))$	"No warm-blooded reptile exists"

Both say the same thing: you can't find a reptile that's warm-blooded!

Nested Quantifiers: "Every Student Has a Favorite Teacher"

- Complex statements often involve relationships between multiple quantified variables.
- Predicates: $Student(x)$, $Teacher(y)$, $Favorite(x, y)$ = "y is x's favorite teacher"
- Translation: $\forall x(Student(x) \rightarrow \exists y(Teacher(y) \land Favorite(x, y)))$
- Read: "For every x, if x is a student, then there exists a y such that y is a teacher and y is x's favorite."

Example

Breaking down the formula structure:

$$\forall x\ (Student(x) \rightarrow (\exists y\ Teacher(y) \land Favorite(x, y)))$$

Universal

Existential

Testing Validity in Predicate Logic

- Testing validity in predicate logic is more complex than using truth tables.
- We can use **natural deduction** rules like Universal Instantiation and Existential Generalization.
- Another method uses **semantic interpretations**: find a domain and interpretation where premises are true but conclusion is false.
- For simple arguments, we can still check validity by applying logical rules systematically.

Validity Test Example

Test: "All elves are immortal. Legolas is an elf. Therefore, someone is immortal."

1. $\forall x(Elf(x) \rightarrow Immortal(x))$ [Premise 1]
2. $Elf(legolas)$ [Premise 2]
3. $Elf(legolas) \rightarrow Immortal(legolas)$ [Universal Instantiation on 1]
4. $Immortal(legolas)$ [Modus Ponens on 2,3]
5. $\exists x Immortal(x)$ [Existential Generalization on 4] ✓

Mathematical Statements in Predicate Logic

- Predicate logic is the language of mathematics—every theorem can be expressed using quantifiers.
- "Every positive number has a square root": $\forall x(Positive(x) \to \exists y(y^2 = x))$
- "There exists a largest prime": $\exists x(Prime(x) \land \forall y(Prime(y) \to y \leq x))$
- Mathematical proofs are essentially chains of valid predicate logic arguments!

Example

The statement "Between any two different real numbers, there's another real number":

$$\forall x \forall y((Real(x) \land Real(y) \land x \neq y) \to \\ \exists z(Real(z) \land ((x < z \land z < y) \lor (y < z \land z < x))))$$

This captures the density property of real numbers!

Scientific Claims as Formal Arguments

- Scientific hypotheses and laws can be expressed precisely using predicate logic.
- "All metals conduct electricity": $\forall x(Metal(x) \rightarrow ConductsElectricity(x))$
- "Some mutations are beneficial": $\exists x(Mutation(x) \wedge Beneficial(x))$
- The scientific method tests these logical claims against empirical evidence.

Scientific Reasoning Pattern

1. Hypothesis: $\forall x(P(x) \rightarrow Q(x))$ (All P's have property Q)
2. Test case: Find some a where $P(a)$ is true
3. Prediction: $Q(a)$ should be true
4. If $\neg Q(a)$: Hypothesis is falsified!

This is the logical structure behind Karl Popper's falsificationism.

Computer Science and Logic

- Every computer program is built on formal logic—if statements, loops, and boolean operations.
- Database queries use predicate logic: SQL's WHERE clause is essentially a logical formula!
- **Program verification** proves code correctness using predicate logic assertions.
- Artificial Intelligence systems use logic for knowledge representation and reasoning.

Common Logical Errors in Everyday Life

- **Hasty Generalization**: Concluding $\forall x P(x)$ from just a few examples.
- **Denying the Antecedent**: From $p \to q$ and $\neg p$, concluding $\neg q$ (invalid!).
- **Quantifier Confusion**: Mixing up "all" and "some" in complex statements.
- **Correlation/Causation**: Observing $\forall x(P(x) \leftrightarrow Q(x))$ but claiming P causes Q.

Real-World Example

"Everyone who recovered took the medicine, so the medicine cured them."

- Observation: $\forall x(Recovered(x) \to TookMedicine(x))$
- Invalid conclusion: $\forall x(TookMedicine(x) \to Recovered(x))$
- This reverses the implication! (Affirming the consequent)

The Power and Limits of Formal Logic

- **Power**: Formal logic gives us certainty—valid arguments guarantee true conclusions from true premises.
- **Power**: It's universal—the same rules work for math, science, philosophy, and daily reasoning.
- **Limit**: Logic alone can't tell us which premises are actually true in the real world.
- **Limit**: Many important human concepts (beauty, justice, meaning) resist complete formalization.

Remember

- Logic is a tool for preserving truth, not discovering it
- Validity $\neq$ truth (valid arguments can have false conclusions if premises are false)
- The best reasoning combines formal logic with empirical evidence
- Even informal arguments benefit from understanding formal structure

Connecting Back to Informal Logic

- Formal logic provides the **skeleton** that informal logic fleshes out with content.
- Understanding formal structure helps you spot weak arguments even in everyday language.
- The precision of formal logic trains you to be more careful with words like "all," "some," "only," and "if."
- Both formal and informal logic are essential tools for critical thinking.

Example

Informal argument: "Since all the swans I've seen are white, all swans must be white."
Formal analysis reveals the problem:

- Premise: $\forall x((\mathit{Swan}(x) \wedge \mathit{SeenByMe}(x)) \rightarrow \mathit{White}(x))$
- Conclusion: $\forall x(\mathit{Swan}(x) \rightarrow \mathit{White}(x))$
- This is invalid! The conclusion doesn't follow from the premise.

Where to Go From Here

- **Modal Logic**: Explores necessity and possibility ("It's possible that..." "It must be that...")
- **Metalogic**: Studies logic itself—proving things about logical systems.
- **Non-Classical Logics**: Fuzzy logic, many-valued logic, paraconsistent logic for special contexts.
- **Applied Logic**: Database theory, automated theorem proving, formal verification of software.

Your Journey Forward

- Practice translating complex statements into logical notation
- Look for logical structure in academic papers and arguments
- Try constructing formal proofs for simple mathematical facts
- Remember: Like learning a language, logic gets easier with practice!

You now have the tools to analyze any argument with mathematical precision!