Propositional Logic

MATH230

Te Kura Pāngarau
Te Whare Wānanga o Waitaha

Outline

Introduction

Lecturer	Robert Culling
Email	robert.culling@canterbury.ac.nz
Office	Jack Erskine 422
Office Hours	Wednesday 12 - 2pm

Email me to organise a time to meet and discuss the course outside office hours.

Course Overview

Following 2000+ years of work developing all of the rich ideas that comprise modern mathematics, some began to wonder about the *consistency* of the methods used. Indeed, there was debate about what "<u>the</u> methods of mathematics" were and camps were formed around these different philosophies.

In order to analyse the methods of proof and have a hope of showing them consistent, mathematicians sought to give very precise (so called, formal) treatments of mathematics.

Every man and his dog developed a formal logical system to model the methods of the practicing mathematician. Gottlob Frege, David Hilbert, and many others all advanced precise systems of formal proof.

For the purposes of this course we are going to study the method written by Gerhard Gentzen. He published this method in 1935 in the paper *Investigations into Logical Deductions*. This paper (at least a translation into English) is available in the library as part of the *The Collected Works of Gerhard Gentzen*.

Computation

At the same time mathematicians wondered if the process of proof, once suitably formalised, could be "computable". That is to ask, is there an *algorithm* that takes in hypotheses and a conclusion and returns a proof of the conclusion relying on the hypotheses.

In a similar way to following an algorithm to compute a division, one might be able to blindly follow an algorithm to produce a proof!

Computation

These questions were asked before anything like our modern computers were developed — Indeed the work of these mathematicians on the problems at the foundations of mathematics informed much of the development of modern computers and the languages we use to talk to them!

Again, every man and his dog developed a formal description of what they meant by computation, to try and determine whether the process of proof could be considered computable. Alan Turing, Stephen Kleene, and Kurt Gödel were among the first.

In this course we will focus on the language (λ-calculus) developed by Alonzo Church first published in 1936 as part of the paper *An Unsolvable Problem of Elementary Number Theory*.

Model of Computation

When defining what is meant by computation there are two, related, issues to think about:

- Developing a language to express computations, and
- Defining what it means to execute, step-by-step, such a computation.

Alonzo Church's λ-calculus achieves both of these goals in an elegant way.

Proofs and Programs

As mathematicians have followed these strands of thinking first laid by Gentzen and Church, they have come to realise that proof and computation are linked! Now we understand that:

$$\textbf{Proof} = \textbf{Program}$$

It is the goal of this course to formally define both sides of this equation and say why they are really equal!

Modern Proof Assistants

This work, started by mathematicians in the early 1900s, has been followed for over a century by others. It has informed the development of programming language theory and other aspects of computer science.

In the present day we now have a number of programming languages (and other software tooling) that allow for the expression and verification of proofs, because they have these ideas baked into them. Not just theorems about mathematics, but also formal verifications (proofs!) that pieces of software do what they claim.

In the final topic of the course we will use Lean to formalise some aspects of mathematics and computer science to see what it is like to prove theorems using these modern tools. **You will need to write code, but this will be covered in class.**

Getting Started with Lean

Although this is the final topic of the course, you can get started with Lean by visiting the following resources. You can try some gamified introductions here:

Lean Intro to Logic

Natural Number Game

Set Theory Game

You can get Lean running on your own computer following these instructions. Once Lean is installed you might like to try:

Mechanics of Proof

Theorem Proving in Lean 4

Mathematics in Lean

We will cover Lean in class sufficiently for the assignment.

Assessment

Labs	10%	8 Handin Problem Sets
Test	20%	Focus on natural deduction
Assignment	20%	Writing proofs in Lean
Exam	50%	Whole course, except Lean.

Further Reading

Dirk van Dalen, Logic and Structure.

Simon Thompson, Type Theory and Functional Programming.

Chapters 1 - 4.

Jeremy Avigad et al Logic and Proof[1].

Chapters 1 - 5, 7 - 9, and 17 - 18.

[1] Note: this uses an outdated version of Lean, but is still a good resource for the theory of this course.

"I think it may be accepted as a starting point that the central problem in the foundation of mathematics is the construction of a symbolic system within which the body of extant mathematics may be derived in accordance with sharply stated and immediately applicable formal rules, and that other questions must be regarded as secondary on the ground that they cannot be given a definite meaning, in fact do not have a definite subject, until such a construction is accomplished. [...] In practice the search is for a formalized basis which is simple and brief beside the entire body of concretely existing mathematics but capable, under its rules of inference, of yielding this entire body and (infinite) further material [...]."

— *Alonzo Church (1939[2])*

[2] "The Present Situation in the Foundation of Mathematics." Alonzo Church, 1939.

Theorems

Mathematicians make statements (claims of facts) about numbers and other abstract structures. Those statements that come with a proof are called *theorems*. It is the job of the mathematician to prove theorems.

Examples

$\forall n \geq 3, x^n + y^n = z^n$ has no non-trivial solutions.

4 is a prime number.

$1 + 1 = 2$

If a sequence of real numbers is bounded, then it converges.

If an increasing sequence of real numbers is bounded, then it converges.

Arguments and Proofs

Analysis of the correctness of a statement relies as much on the connective words (grammar) as it does the technical context specific definitions.

Example:

If p divides ab, then p divides a or p divides b.

Analysis of the correctness of a theorem relies as much on the connective words as it does the technical definitions.

Example

If a divides b and b divides a, then $a = b$ or $a = -b$.

Analysis of the correctness of a theorem relies as much on the connective words as it does the technical definitions.

Example

If $f(x)$ is continuous on $[a, b]$ and differentiable on (a, b), then there exists a $c \in (a, b)$ such that

$$f'(c) = \frac{f(b) - f(a)}{b - a}$$

Connective Tissue

It is the connective words, as much as the mathematical content words, that we have to analyse when deciding whether these statements are correct; whether these are theorems.

Propositional logic formalises the structure of these connective words. We will return to the context specific terms in later topics.

Example: Natural Language

- If Watson moves in with Holmes, then Holmes will be forever annoyed. Watson moved in with Holmes. Therefore, Holmes will be forever annoyed.

- If Watson can trap Moriarty, then Holmes can. Holmes can't trap Moriarty. Therefore, Watson can't.

- Either Holmes catches Moriarty or the world will fall into chaos. The world has fallen into chaos. Therefore, Holmes did not catch Moriarty.

Argument

An argument is a finite collection of declarative sentences (propositions), one of which is singled out as the conclusion, while the others are considered premises.

Premises are the evidence claiming to support the conclusion.

Example: Natural Language

- If Watson moves in with Holmes, then Holmes will be forever annoyed. Watson moved in with Holmes. Therefore, Holmes will be forever annoyed.

  Let's break this up into premises and conclusion:

Propositional Structures

An atomic proposition has no propositional substructure.

We saw above that some propositions do have extra structure: “If... , then....” and “Either .... or ... ” and “can’t” are important to the nature of the argument.

Such connectives are used to join atomic propositions into compound propositions.

Example: Natural Language

- Either Holmes catches Moriarty or the world will fall into chaos. The world has fallen into chaos. Therefore, Holmes did not catch Moriarty.

  Let's break this up into premises and conclusion and determine the atomic propositions.

Moving Away from Natural Language

It was hoped that mathematics could be written in such a precise manner that it could be routinely checked. Furthermore, it was thought that once mathematics was so formalised, that it could be shown consistent and complete; that is, not able to prove non-sense and able to prove (or refute) every statement.

Toward this end mathematicians (Frege and those that followed him) chose to write mathematics in the language of logic:

- Propositional Logic.
- First Order Predicate Logic.

Propositional Connectives

To express the same syntactic structure of an argument without the ambiguities of a natural language we use capital (English) letters to denote atomic propositions, called *propositional variables*. We use the following symbols to construct compound propositions:

- $\neg$: "It is not the case that... " or "Not... "
- $\wedge$: "Both... and ... "
- $\vee$: "Either... or ... "
- $\rightarrow$: "If... , then ... "
- $\leftrightarrow$: " ... if and only if ... "

These symbols, the propositional connectives, play the role of the connective tissue in the statements given on previous slides.

Example: NL to PL

- If Watson can trap Moriarty, then Holmes can. Holmes Can't trap Moriarty. Therefore Watson can't.

Grammar

Our language is further made up of *well-formed formulae* which we define inductively as follows:

[Well-Formed Formulae]

- **Atomic Formulae:** If α is a single propositional variable, then α is a wff.
- **Negation:** If α is a wff, then $\neg\alpha$ is a wff.
- **Binary Connective:** If α and β are wff and $*$ is a binary connective, then $(\alpha * \beta)$ is a wff.

Notation: We will refer to the totality of well-formed propositions as "Prop" and we will write "α : Prop" to denote the fact that α is a well-formed proposition.

Inductive Definition of Prop

This inductive definition can be stated in Backus-Naur form.

Examples

Which of the following are wff in propositional logic?

1. A
2. AB
3. $(A \to B)$
4. $A \to B \to C$
5. $((A \to B) \to C)$
6. $\neg Q$
7. $A \lor Q$
8. $A \to \neg B \lor C$

Binding Conventions:

- $\neg$ binds most tightly,
- $\vee$ and $\wedge$ bind more tightly than $\rightarrow$,
- $\rightarrow$ binds more tightly than $\leftrightarrow$.

Example: Parse the wff $A \rightarrow \neg B \vee C$.

Binding Conventions

Binding Conventions:

- $\neg$ binds most tightly,
- $\vee$ and $\wedge$ bind more tightly than $\rightarrow$,
- $\rightarrow$ binds more tightly than $\leftrightarrow$.

To disambiguate $\wedge$ and $\vee$ we group terms from the left. In this way, we say that $\wedge$ and $\vee$ associate to the left.

Example: Parse the wff $A \wedge B \wedge C$.

Syntax Trees

Using the binding convention above allows for each well-formed formula to be parsed into a syntax tree.

$A \wedge (B \vee C)$	$A \wedge \neg B \wedge C$	$A \rightarrow B \vee C \wedge D$

Hiding the Goods

By packing the statements:

"p divides ab" or "f is continuous on $[a, b]$"

into a propositional variable A, we have lost a lot of information from the statement that we're trying to analyse. It's no longer about primes or continuous functions.

For now, we will focus on the connectives alone. Studying the structure of the argument, rather than the mathematics.

Later we will introduce more structure to our logic which will allow us to bring back the mathematical content.

Example

"Thin is guilty," observed Watson, "because either Holmes is right and the vile Moriarty is guilty, or he (Holmes) is wrong and Thin did the job; but those scoundrels are either both guilty or both innocent; and, as usual, Holmes is correct".

Example from Richard Jeffrey, *Formal Logic: Its Scope and Limits*

Argument Structure

$$\frac{\begin{array}{c}\text{Proposition 1}\\ \text{Proposition 2}\\ \vdots \\ \text{Proposition n}\end{array}}{\text{Conclusion}}$$

Question: What makes for a "good argument"? What might we mean by a "good argument"? What dos it mean for the conclusion to follow from the hypotheses?

Truth and Proof

Semantic	If the conclusion is "true" whenever all hypotheses are "true", then the conclusion is said to be a semantic consequence of the hypotheses.
Syntactic	If there is a "proof" that the hypotheses "unfold" and "combine" towards the conclusion, then the conclusion is said to be a syntactic consequence of the hypotheses.

Theorem: γ is a semantic consequence of Σ if and only if γ is a syntactic consequence of Σ [**vDalen**].

Example

Provide a proof to show

$$P \wedge (Q \wedge R),\; R \to T \;\vdash\; P \wedge T$$

- How are we to "unfold" or make use of hypotheses?
- How are we to obtain the conclusion?
- What should such a proof look like?
- It should "reflect actual reasoning".

BHK Intepretation

Brouwer, Heyting, and Kolmogorov proposed the following (inductive) interpretation of what it should mean to prove statements involving propositional connectives:

$P \wedge Q$	to prove a conjunction we must provide both a proof of P and a proof of Q.
$P \to Q$	to prove an implication we must provide an algorithm for turning a proof of P into a proof of Q.
$P \vee Q$	to prove a disjunction we must provide either a proof of P or a proof of Q.
$\neg P$	to prove a negation we must provide an algorithm that turns a proof of P into a proof of $\perp$.

This presentation is taken from the Standford Encyclopedia of Philosophy article by Bridges, Palmgren, and Ishihara [**sep-mathematics-constructive**].

Natural Deduction Calculus

We are going to develop a proof method that is inline with the BHK interpretation of the logical connectives. This proof method was first presented by Gerhard Gentzen. As such it is often referred to as the Gentzen Calculus. We will call it “Natural Deduction”.

This method will develop proofs by unfolding their hypotheses in a manner consistent with the BHK.

Example

Provide a proof to show

$$P \wedge (Q \wedge R),\ R \rightarrow T \ \vdash\ P \wedge T$$

Example: What would it require to deduce $A \wedge B$ in the course of a proof? Use the BHK!

Example: What would it require to deduce $A \wedge B$ in the course of a proof? Use the BHK!

$$\dfrac{\begin{array}{cc} \Sigma_1 & \Sigma_2 \\ \mathcal{D}_1 & \mathcal{D}_2 \\ A & B \end{array}}{A \wedge B} \wedge I$$

If we have a deduction for A from hypotheses Σ_1 and a deduction for B from hypotheses Σ_2, then *together* we should consider those deductions a proof for $A \wedge B$ from hypotheses $\Sigma_1 \cup \Sigma_2$.

$\wedge$ Elimination

Example: Suppose $A \wedge B$ were a premise in a proof. What can we conclude from such a premise?

$\wedge$ Elimination

Example: Suppose $A \wedge B$ were a premise in a proof. What can we conclude from such a premise?

$$\dfrac{\begin{array}{c}\Sigma \\ \mathcal{D} \\ A \wedge B\end{array}}{A}\ \wedge E_L \qquad\qquad \dfrac{\begin{array}{c}\Sigma \\ \mathcal{D} \\ A \wedge B\end{array}}{B}\ \wedge E_R$$

A proof of $A \wedge B$ can be extended to either a proof of A or a proof of B, relying on the same set Σ of hypotheses.

Example: Idempotence of $\wedge$

Provide a proof to show

$$A \wedge A \vdash A$$

Example: Commutativity of $\wedge$

Provide a proof to show

$$A \wedge B \vdash B \wedge A$$

Deductions

We define deductions (or derivations, or proofs) inductively according to the following rules:

- For each formula α,

$$\alpha$$

  is a deduction with conclusion α and premises $\{\alpha\}$.

- From given deductions, an application of a *rule of inference* yields a new deduction.

- Anything that is not a deduction by virtue of the above is *not* a deduction.

If there exists a deduction ${}^{\Sigma}_{\alpha}\mathcal{D}$ of α from Σ, then we say α is a *syntactic consequence of* (*derivable from*, or *provable from*) Σ and denote this $\Sigma \vdash \alpha$.

If we follow this idea for all of the logical connectives in propositional logic, then we can develop a method for writing proofs based on the *syntactic* structre of the logical connectives alone. We call this method of proof *Natural Deduction* and we are following Gerhard Gentzen's notation [**vDalen**, **thompson**].

We need to know how to (i) deduce and (ii) conclude from, each logical connective. In other words, for each logical connective we need to develop rules for introducing the logical connective and eliminating the logical connective.

So we will spend some time writing down the **Rules of Inference** for our logical connectives.

Hypothetical Reasoning

In mathematics we often prove statements of the following hypothetical form: "If ..., then ..."

Example: If f is differentiable at x_0, then it is continuous at x_0.

Proof: "Let f be a differentiable function..."

The proof of this implication will assume the hypothesis of differentiability and show that it implies continuity. In order to prove an implication $P \rightarrow Q$ the proof starts by assuming we know P and then using that to tell us about Q.

The conclusion is the entire implication, not just continuity.

Example: Hypothetical Reasoning

If $p|a$ and $p|(a+b)$, then $p|b$.

Example: Hypothetical Reasoning

$$\dfrac{\dfrac{\overline{A \wedge (B \wedge C)}^{\,1}}{B \wedge C}\,\wedge E_L}{B}\,\wedge E_L$$

Question: What does the deduction prove?

$\rightarrow$ Introduction

If ${}^{\Sigma}_{\beta}\mathcal{D}$ is a deduction of β from Σ, then

$$\dfrac{\begin{array}{c}\Sigma \cup \{\cancel{\alpha}\} \\ \mathcal{D} \\ \beta\end{array}}{\alpha \rightarrow \beta} \rightarrow I$$

is a deduction of $\alpha \rightarrow \beta$ from hypotheses $\Sigma \backslash \{\alpha\}$.

Note: As the assumption α is struck out after this deduction, we are free to use α *even if it is not in* Σ when using implication introduction.

$\rightarrow$ Elimination (MP)

If ${}^{\Sigma_1}_{\alpha \rightarrow \beta}\mathcal{D}_1$ and ${}^{\Sigma_2}_{\alpha}\mathcal{D}_2$ are deductions, then

$$\cfrac{\begin{array}{c}\Sigma_1\\ \mathcal{D}_1\\ \alpha \rightarrow \beta\end{array} \qquad\qquad \begin{array}{c}\Sigma_2\\ \mathcal{D}_2\\ \alpha\end{array}}{\beta} \rightarrow E$$

is a deduction of β from $\Sigma_1 \cup \Sigma_2$.

Example

Show $P \to Q,\ Q \to R \ \vdash \ P \to R$

Deduction Theorem

Theorem: $\Sigma \vdash \alpha \to \beta$ if and only if $\Sigma \cup \{\alpha\} \vdash \beta$

Proof

Deduction Theorem

What the deduction theorem has formalised is the idea that in order to prove an implication

$$\Sigma \ \vdash \alpha \to \beta$$

It is equivalent to assume α the antecedent and prove β the consequent.

$$\Sigma, \bar{\alpha} \ \vdash \ \beta$$

When coming to write a proof, one should first check the structure of the conclusion to be proved as this can help determine a first move in the proof — like the temporary assumption of an antecedent.

Example

Show that $P \to (\neg S \to L), \quad P \to \neg S, \quad P \vdash L$

Currying

Show $(A \wedge B) \to C \vdash A \to (B \to C)$

Example

Show $A \to B, A \to C, A \vdash B \wedge C$

$\vee$ Introduction

If ${}^{\Sigma}_{\alpha}\mathcal{D}$ is a derivation of α from Σ, then

$$\dfrac{\begin{array}{c}\Sigma\\ \mathcal{D}\\ \alpha\end{array}}{\alpha \vee \beta}\ \vee I_R \qquad\qquad \dfrac{\begin{array}{c}\Sigma\\ \mathcal{D}\\ \alpha\end{array}}{\beta \vee \alpha}\ \vee I_R$$

are derivations of $\alpha \vee \beta$ and $\beta \vee \alpha$ from Σ.

Note: We are free to choose β as, if we know α to be the case, then $\alpha \vee \beta$ is necessarily the case *for any* β.

$\vee$ Elimination

If ${}^{\Sigma_1}_{\alpha\vee\beta}\mathcal{D}_1$, ${}^{\Sigma_2}_{\alpha\to\gamma}\mathcal{D}_2$, and ${}^{\Sigma_2}_{\beta\to\gamma}\mathcal{D}_3$ are derivations, then

$$\dfrac{\begin{array}{c}\Sigma_1\\ \mathcal{D}_1\\ \alpha\vee\beta\end{array}\qquad\begin{array}{c}\Sigma_2\\ \mathcal{D}_2\\ \alpha\to\gamma\end{array}\qquad\begin{array}{c}\Sigma_2\\ \mathcal{D}_2\\ \beta\to\gamma\end{array}}{\gamma}\ \vee_E$$

is a derivation of γ from $\Sigma_1 \cup \Sigma_2 \cup \Sigma_3$.

Note: You can't remove one of the arguments from a disjunction. Knowledge of $\alpha \vee \beta$ is not sufficient to conclude either α or β alone. Following the BHK, a proof of $\alpha \vee \beta$ is either a proof of α or a proof of β, but without a record of which case we are in we can't assume either way. This means we need to account for both possibilities.

Example

Show $A \vee B,\ (A \vee C) \rightarrow D,\ B \rightarrow D\ \vdash D$

Example

Show $A \vee (B \wedge C) \vdash (A \vee B) \wedge (A \vee C)$

Common Mistake (!)

Positive Minimal Logic

The calculus developed so far with the introduction and elimination rules of the three connectives $\wedge, \vee$, and $\rightarrow$ is called positive minimal logic.

Many theorems can be (stated and) proved with these rules of inference alone.

There are many statements that one might expect to prove which require other modes of reasoning i.e. further rules of inference.

Falsum

We introduce the logical constant $\perp$ (falsum or absurdity) to define the syntactic form of the $\neg$ connective. We make the following definition:

$$\neg\alpha := \alpha \to \perp$$

$$\frac{\alpha \qquad \alpha \to \perp}{\perp}\ \mathsf{MP} \qquad\qquad \frac{\begin{array}{c}\overline{\cancel{\alpha}}\\ \mathcal{D}\\ \perp\end{array}}{\alpha \to \perp} \to I$$

Falsum $\perp$ is an atomic proposition which is to be thought of as denoting "absurdity" or "contradiction".

Example: Modus Tollens

Show $A \to B,\ \neg B \vdash \neg A$

Proving a Negative Statement

This means that in order to prove $\neg\alpha$ we must show that $\perp$ follows from the assumption of α. In other words, proving $\neg\alpha$ is equivalent to showing α leads to a contradiction. This process is often referred to as *proof by refutation*, or simply refutation.

Contradiction Implies Absurdity

Show $A \wedge \neg A \ \vdash \ \bot$

Minimal Logic

Together the rules of inference that we've given so far define *minimal* logic. They include much, but not all, of the logical inferences that practising mathematicians might use in a proof.

However, it is not universally agreed as to how minimal logic should be extended. There are philosophical differences among mathematicians and logicians about what other rules of inference should be included.

- Intuitionistic logic
- Classical logic
- Modal logic

What's Missing?

What should be done if the hypotheses yield a contradiction?

$$\cfrac{\begin{array}{c}\Sigma \\ \mathcal{D} \\ \bot\end{array}}{?}\ ?$$

Should we be able to prove the following?

$$\vdash\ P \vee \neg P$$

$$\neg\neg P\ \vdash\ P$$

$$\vdash\ (P \to Q) \vee (Q \to P)$$

Affirmative answers to these questions require further rules of inference.

Ex Falso Sequitur Quadlibet

So far, we have not made much mention of how to deal with the derivation of $\perp$ absurdity. Indeed, it has no introduction rule.

If ${}^{\Sigma}_{\perp}\mathcal{D}$ is a deduction of $\perp$ from Σ, then

$$\dfrac{\begin{array}{c}\Sigma \\ \mathcal{D} \\ \perp\end{array}}{\alpha}\ \mathsf{XF}$$

is a derivation of α from the assumptions Σ.

Anything you want follows from a falsehood.

Disjunctive Syllogism

Show $A \vee B, \neg B \ \vdash A$

XF as Null Disjunction

Disjunction has two introduction rules; both of which need to be taken into account when eliminating a disjunction.

Following this one can argue that since $\perp$ has no introduction rules, there is nothing to take account of when eliminating $\perp$ and hence XF allows one to conclude anything. All possible cases (i.e. all zero of them) lead to P, therefore we may conclude P follows from Falsum.

Intuitionistic Logic

Ex Falso Quodlibet extends the class of theorems provable in the natural deduction calculus. It is the logic of intuitionists and constructivists; mathematicians who believe proofs should have computational content.

$$\text{Minimal Logic} + \text{Ex Falso} = \text{Intitionistic Logic}$$

However, there are classically valid sequents, such as the LEM, which are not derivable in the intuitionistic calculus.

Double Negation Elimination

Show $\neg\neg A \vdash A$

Double Negation Elimination

Show $\neg\neg A \vdash A$

Ex falso does not give us a proof. In fact we have shown the following: $\{\neg\neg A, \neg A\} \vdash A$.

Summary

We have built the propositional calculus up in steps:

- Positive minimal logic,
- Minimal logic,
- Intuitionistic logic.

One can show, using non-classical semantics, that these logics are unable to prove some theorems that are fundamental to much of mathematics.

We will consider one more mode of reasoning.

Example

There exist irrational numbers x, y such that x^y is rational.

Reductio Ad Absurdum

If ${}^{\Sigma}_{\perp}\mathcal{D}$ is a deduction of $\perp$ from Σ, then

$$\cfrac{\begin{array}{c}\Sigma \cup \{\cancel{\neg\alpha}\} \\ \mathcal{D} \\ \perp\end{array}}{\alpha}\ \text{RAA}$$

is a derivation of α from the assumptions $\Sigma \backslash \{\neg\alpha\}$.

If absurdity follows from $\neg\alpha$, then we may conclude α **and discharge** $\neg\alpha$ **from our assumptions.**

Double Negation

Show $\neg\neg A \vdash A$

Law of Excluded Middle

Show $\vdash A \vee \neg A$

Classical Logic

The class of theorems one can prove increases with the addition of RAA. In fact, the class of theorems provable in classical logic includes all theorems of intuitionistic logic.

$$\text{Minimal Logic} + \text{RAA} = \text{Classical Logic}$$

One can derive the Ex Falso rule of inference using RAA.

Common Misconception

Many proofs that claim to use reductio ad absurdum are really *refutations by contradiction.*

- Irrationality of $\sqrt{2}$
- Infinitude of primes
- No smallest positive rational number

These proofs just use implication introduction to prove a negation.

Example

Show $A \rightarrow B, \;\; A \rightarrow \neg B \;\vdash \neg A$

Derived Rules of Inference

Proofs can be simplified by using results already proved. You may, in the course of a proof, use any result that has been proven in class or previously in a tutorial. However, when substituting previous proofs, you must bring all of the premises with the conclusion.

We have already seen this with the use of *modus tollens* (MT) in some examples.

Making use of (an instance of) LEM instead of RAA can make proofs more straight forward.

This can help keep proofs manageable and neat.

Example: Substituting LEM

Show $A \to B \vdash \neg A \vee B$

Intuitionistic to Classical

$$\text{Classical} = \text{IL} + \text{RAA} = \text{IL} + \text{LEM} = \text{IL} + \text{DNE}$$

We described the passage from intuitionistic logic to classical by the addition of the RAA rule of inference. We can get a logic of equivalent power in a number of ways.

One could declare for each P, $P \vee \neg P$ as a theorem.

One could add a double negation elimination rule of inference.

Adding any of these to minimal logic gives you the same set of theorems as classical logic as we defined it above.

See tutorial to prove this.

Departure from BHK

Notice that the addition of RAA has forced us to lose the BHK interpretation of our proofs. For each proposition P LEM is a theorem:

$$\vdash P \vee \neg P$$

The BHK asserts a proof of $A \vee B$ must consist either of a proof of A or a proof of B. But the classical proof of $\vdash P \vee \neg P$ does not contain that information; it does not tell us which of P or $\neg P$ is provable.

The inclusion of RAA allows for proofs of some apparently harmless theorems like DNE and RAA. However this power is not without its consequences...

Example

Provide a proof to show

$$\vdash (A \to B) \vee (B \to A)$$

Dealer's Choice

Choice of logic (i.e. rules of inference) is left to the mathematician. Therefore, we should be more specific when we assert one Prop is a syntactic consequence of a set Σ. This is a *relative* notion and so one should quote the logic used in the derivation.

We say well-formed formulae are *syntactically equivalent* if both

$$\alpha \vdash \beta \quad \text{and} \quad \beta \vdash \alpha$$

Examples

- $A \vee B \dashv\vdash B \vee A$
- $A \to B \ \dashv\vdash\ \neg A \vee B$

Logical equivalences should be stated *with respect to a logic*.
Typically the weakest logic for which it can be proved.

Theorems

We say a well-formed formula α is a theorem if there exists a natural deduction $\mathcal{D}$ from no assumptions i.e. $\Sigma = \emptyset$ and we denote this as $\vdash \alpha$.

Example: Law of the Excluded Middle

Example: $\vdash A \to (B \to A)$

Note: This should be stated *with respect to a logic.* Typically the weakest logic for which it can be proved.

Different Logics

When we say one logic is weaker relative to another, what we are saying is that the set of theorems is a subset of the other.

Reference to "classical logic" as a whole might be referring to the collection of the rules of inference, or it might be referring to the collection of all theorems of classical logic.

$$\text{Positive Minimal} \subset \text{Minimal} \subset \text{Intuitionistic} \subset \text{Classical}$$

Example of Equivalence

If α and β are syntactically equivalent, then $\vdash \alpha \leftrightarrow \beta$.

Equivalence of Theorems

If $\vdash \alpha$ and $\vdash \beta$, then $\vdash \alpha \leftrightarrow \beta$

Further Reading

Dirk van Dalen, Logic and Structure.

Simon Thompson, Type Theory and Functional Programming.

Chapters 1 - 4.

Jeremy Avigad et al, Logic and Proof[3].

Chapters 1 - 5, 7 - 9, and 17 - 18.

---

[3] Note: this uses an outdated version of Lean, but is still a good resource for the theory of this course.