

Philosophy 148 — Announcements & Such

- Administrative Stuff
 - See the course website for all administrative information (also, note that lecture notes are posted the morning prior to each class):
`http://socrates.berkeley.edu/~fitelson/148/`
 - Section times (II): Those of you who can't make Tu @ 10–11, please fill-out an index card. New times: Mon or Wed 9–10 or 10–11.
- Last Time: Review of Boolean (Truth-Functional) Sentential Logic
 - Truth-Table definitions of connectives
 - Semantical (Metatheoretic) notions
 - * Individual Sentences: Logical Truth, Logical Falsity, etc.
 - * Sets of Sentences: Entailment (\models), Consistency, etc.
- Today: Finite Propositional Boolean Algebras, Review of Boolean (Truth-Functional) Predicate Logic, and a General Boolean Framework

Finite Propositional Boolean Algebras

- Sentences express *propositions*. We individuate propositions according to their logical content. If two sentences are logically equivalent, then they express the same proposition. [E.g., “ $A \rightarrow B$ ” and “ $\sim A \vee B$ ”]
- A *finite propositional Boolean algebra* is a finite set of *propositions* which is *closed* under the (Boolean) logical operations.
- A set S is *closed* under a logical operation λ if applying λ to a member (or pair of members) of S always yields a member of S .
- Example: consider a sentential language \mathcal{L} with three atomic letters “ X ”, “ Y ”, and “ Z ”. The set of propositions expressible using the logical connectives and these letters is a finite Boolean algebra of propositions.
- This Boolean algebra has $2^3 = 8$ *atomic propositions* or *states* (i.e., the rows of a 3-atomic sentence truth-table!). Question: How many propositions does it contain *in total*? [A: $2^8 = 256$ — explanation later]

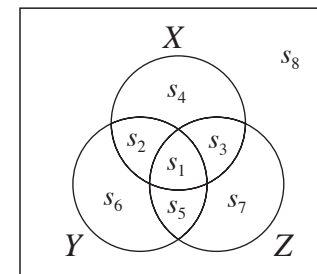
Propositional Boolean Algebras: States, Truth-Tables, and Venn Diagrams

- A *literal* is either an atomic sentence or the negation of an atomic sentence (e.g., “ A ” and “ $\sim A$ ” are the literals involving the atom “ A ”).
- A *state* of a Boolean algebra \mathcal{B} is a proposition expressed by a *state description* — a *maximal* conjunction of literals in a language $\mathcal{L}_{\mathcal{B}}$ describing \mathcal{B} (maximal: having exactly one literal for each atom of $\mathcal{L}_{\mathcal{B}}$).
- Consider an algebra \mathcal{B} described by a 3-atom language $\mathcal{L}_{\mathcal{B}}$ (X, Y, Z). The states of \mathcal{B} are described by the $2^3 = 8$ *state descriptions* of $\mathcal{L}_{\mathcal{B}}$:

- (s_1) $X \& Y \& Z$
- (s_2) $X \& Y \& \sim Z$
- (s_3) $X \& \sim Y \& Z$
- (s_4) $X \& \sim Y \& \sim Z$
- (s_5) $\sim X \& Y \& Z$
- (s_6) $\sim X \& Y \& \sim Z$
- (s_7) $\sim X \& \sim Y \& Z$
- (s_8) $\sim X \& \sim Y \& \sim Z$

- We can “visualize” the states of \mathcal{B} using a truth table or a Venn Diagram.

X	Y	Z	States
T	T	T	s_1
T	T	F	s_2
T	F	T	s_3
T	F	F	s_4
F	T	T	s_5
F	T	F	s_6
F	F	T	s_7
F	F	F	s_8



- Every proposition expressible in the sentential language $\mathcal{L}_{\mathcal{B}}$ can be expressed as a *disjunction of state descriptions* (how does this work?).
- Thus, every proposition expressible in $\mathcal{L}_{\mathcal{B}}$ can be “visualized” simply by shading combinations of the 8 state-regions of the Venn Diagram of \mathcal{B} .
- How many ways of shading the above Venn diagram are there?
 $2^8 = 256 =$ the number of disjunctions of the s_i (255) — plus 1 for \perp .
- That’s why there are $2^{2^3} = 2^8 = 256$ propositions (in total) in \mathcal{B} .

(Finite) Monadic Predicate Logic (MPL) I

- Consider the following two arguments:

① Socrates is wise.	② Everyone is happy.
∴ Someone is wise.	∴ Plato is happy.
- Intuitively, both ① and ② are *valid* (why?). But, if we try to translate these into sentential logic, we get the *invalid* SL forms:

① _{SL} S	② _{SL} H
∴ W	∴ P
- In SL, we are not able to express what the premises and conclusions of these kinds of arguments *have in common*.
- If it's not *atomic sentences* that the premises and conclusions of such arguments have in common, then what *is* it?
- This is what monadic predicate logic is about ...

(Finite) Monadic Predicate Logic II

- We need a *richer language* than SL — one which accurately captures the deeper *logical structure* of arguments like ① and ②. New Jargon:
- A **predicate** is something which *applies to* an object or *is true of* an object. *E.g.*, The predicate (**is**) **Wise** applies to Socrates.
- A **proper name** is a word or a phrase which *stands for*, or *refers to*, or *denotes* a person, place, or thing. *E.g.*, 'Socrates' is a proper name.
- Quantifiers** specify *quantities*. *E.g.*, 'someone' means *at least one* person and 'everyone' means *all* people. [think: "Some" and "All"]
- The collection of objects to which the quantifiers in a statement are *relativized* is called the **domain of discourse** of the statement. *In this course, we'll only work with finite domains* (*e.g.*, the domain of ravens).
- As we'll explain soon, when we are restricted to finite domains, monadic predicate logic is *almost* "sentential logic in disguise".

(Finite) Mondacic Predicate Logic III

- Among the atomic sentences of MPL (*in addition to SL sentence letters*) are (new) strings of the form '*Xn*', where '*X*' is a (monadic) predicate, and '*n*' is an individual constant (proper name).
- We use the lower-case letters '*a*'-'*s*' as individual constants ('*t*'-'*z*' are used as *variables* — we won't say too much about variables).
- Some examples of these new kinds of atomic sentences:
 - 'Branden is tall.' → '*Tb*'.
 - 'Honda is an automobile manufacturer.' → '*Ah*'.
 - 'New York is a city.' → '*Cn*'.
- As in SL, we can *combine* different MPL atomic sentences using the sentential connectives to yield complex sentences. For instance:
 - 'Branden is tall, but Ruth is not tall.' → '*Tb* & ~*Tr*'.

Quantifiers in (Finite) Mondaic Predicate Logic

- In finite domains, we can (almost) think of a universal claim ($\forall v$) ϕv as a *conjunction* which asserts that the predicate expression ϕ applies to *each object* in the domain [*i.e.*, as $\phi a \ \& \ (\phi b \ \& \ (\phi c \ \& \ \dots))$].
- Analogously, in finite domains, we can (almost) think of an existential claim ($\exists v$) ϕv as a *disjunction* which asserts that ϕ applies to *at least one object* in the domain [*i.e.*, as $\phi a \ \vee \ (\phi b \ \vee \ (\phi c \ \vee \ \dots))$].
- Upshot: when the size of the domain is finite (and, *known*), we can say everything we need to say about the domain using *sentential logic*.
- For each ϕ and each constant τ , we can construct an atomic sentence $\phi\tau$. And, if we think of these as the atomic sentences of a sentential language, then we can express every claim we need to express using conjunctions, disjunctions, *etc.* of these atomic sentences.
- So, we can use finite Boolean algebras of propositions for FMPL too...

Finite Boolean Algebras and Finite Monadic Predicate Logic

- Consider the language \mathcal{L}_2^2 , with two monadic predicates F and G and two individual constants a and b . \mathcal{L}_2^2 has 16 state descriptions:

$Fa \ \& \ Ga \ \& \ Fb \ \& \ Gb$	$Fa \ \& \ Ga \ \& \ Fb \ \& \ \sim Gb$	$Fa \ \& \ Ga \ \& \ \sim Fb \ \& \ Gb$
$Fa \ \& \ Ga \ \& \ \sim Fb \ \& \ \sim Gb$	$Fa \ \& \ \sim Ga \ \& \ Fb \ \& \ Gb$	$Fa \ \& \ \sim Ga \ \& \ Fb \ \& \ \sim Gb$
$Fa \ \& \ \sim Ga \ \& \ \sim Fb \ \& \ Gb$	$Fa \ \& \ \sim Ga \ \& \ \sim Fb \ \& \ \sim Gb$	$\sim Fa \ \& \ Ga \ \& \ Fb \ \& \ Gb$
$\sim Fa \ \& \ Ga \ \& \ Fb \ \& \ \sim Gb$	$\sim Fa \ \& \ Ga \ \& \ \sim Fb \ \& \ Gb$	$\sim Fa \ \& \ Ga \ \& \ \sim Fb \ \& \ \sim Gb$
$\sim Fa \ \& \ \sim Ga \ \& \ Fb \ \& \ Gb$	$\sim Fa \ \& \ \sim Ga \ \& \ Fb \ \& \ \sim Gb$	$\sim Fa \ \& \ \sim Ga \ \& \ \sim Fb \ \& \ Gb$
	$\sim Fa \ \& \ \sim Ga \ \& \ \sim Fb \ \& \ \sim Gb$	

- This characterizes a Boolean algebra \mathcal{B} with 16 states. We cannot easily visualize \mathcal{B} with Venn diagrams. But, we can (and will) use truth-tables.
- Note: the total number of propositions in \mathcal{B} is very large ($2^{16} = 65536$). There are 65535 disjunctions of state-descriptions of \mathcal{L}_2^2 (plus 1 for \perp).

Fa	Ga	Fb	Gb	State Descriptions (s_i)
T	T	T	T	$Fa \ \& \ Ga \ \& \ Fb \ \& \ Gb$
T	T	T	F	$Fa \ \& \ Ga \ \& \ Fb \ \& \ \sim Gb$
T	T	F	T	$Fa \ \& \ Ga \ \& \ \sim Fb \ \& \ Gb$
T	T	F	F	$Fa \ \& \ Ga \ \& \ \sim Fb \ \& \ \sim Gb$
T	F	T	T	$Fa \ \& \ \sim Ga \ \& \ Fb \ \& \ Gb$
T	F	T	F	$Fa \ \& \ \sim Ga \ \& \ Fb \ \& \ \sim Gb$
T	F	F	T	$Fa \ \& \ \sim Ga \ \& \ \sim Fb \ \& \ Gb$
T	F	F	F	$Fa \ \& \ \sim Ga \ \& \ \sim Fb \ \& \ \sim Gb$
F	T	T	T	$\sim Fa \ \& \ Ga \ \& \ Fb \ \& \ Gb$
F	T	T	F	$\sim Fa \ \& \ Ga \ \& \ Fb \ \& \ \sim Gb$
F	T	F	T	$\sim Fa \ \& \ Ga \ \& \ \sim Fb \ \& \ Gb$
F	T	F	F	$\sim Fa \ \& \ Ga \ \& \ \sim Fb \ \& \ \sim Gb$
F	F	T	T	$\sim Fa \ \& \ \sim Ga \ \& \ Fb \ \& \ Gb$
F	F	T	F	$\sim Fa \ \& \ \sim Ga \ \& \ Fb \ \& \ \sim Gb$
F	F	F	T	$\sim Fa \ \& \ \sim Ga \ \& \ \sim Fb \ \& \ Gb$
F	F	F	F	$\sim Fa \ \& \ \sim Ga \ \& \ \sim Fb \ \& \ \sim Gb$

(Finite) Relational Predicate Logic (RPL)

- We won't work much (hardly at all) with relational predicate logic in this course, but a similar trick can be done to use finite relational logical languages to characterize finite Boolean algebras of propositions:
- Consider a language \mathcal{L} , which has one 2-place predicate "R" and two individual constants "a" and "b". \mathcal{L} also has 16 state descriptions:

$Raa \ \& \ Rab \ \& \ Rba \ \& \ Rbb$	$Raa \ \& \ Rab \ \& \ Rba \ \& \ \sim Rbb$	$Raa \ \& \ Rab \ \& \ \sim Rba \ \& \ Rbb$
$Raa \ \& \ Rab \ \& \ \sim Rba \ \& \ \sim Rbb$	$Raa \ \& \ \sim Rab \ \& \ Rba \ \& \ Rbb$	$Raa \ \& \ \sim Rab \ \& \ Rba \ \& \ \sim Rbb$
$Raa \ \& \ \sim Rab \ \& \ \sim Rba \ \& \ Rbb$	$Raa \ \& \ \sim Rab \ \& \ \sim Rba \ \& \ \sim Rbb$	$\sim Raa \ \& \ Rab \ \& \ Rba \ \& \ Rbb$
$\sim Raa \ \& \ Rab \ \& \ Rba \ \& \ \sim Rbb$	$\sim Raa \ \& \ Rab \ \& \ \sim Rba \ \& \ Rbb$	$\sim Raa \ \& \ Rab \ \& \ \sim Rba \ \& \ \sim Rbb$
$\sim Raa \ \& \ \sim Rab \ \& \ Rba \ \& \ Rbb$	$\sim Raa \ \& \ \sim Rab \ \& \ Rba \ \& \ \sim Rbb$	$\sim Raa \ \& \ \sim Rab \ \& \ \sim Rba \ \& \ Rbb$
	$\sim Raa \ \& \ \sim Rab \ \& \ \sim Rba \ \& \ \sim Rbb$	

- The truth-table for \mathcal{L} looks very much like the one for \mathcal{L}_2^2 , above, since each language (\mathcal{L} and \mathcal{L}_2^2) has four atomic sentences. Here's the table:

Raa	Rab	Rba	Rbb	State Descriptions (s_i)
T	T	T	T	$Raa \ \& \ Rab \ \& \ Rba \ \& \ Rbb$
T	T	T	F	$Raa \ \& \ Rab \ \& \ Rba \ \& \ \sim Rbb$
T	T	F	T	$Raa \ \& \ Rab \ \& \ \sim Rba \ \& \ Rbb$
T	T	F	F	$Raa \ \& \ Rab \ \& \ \sim Rba \ \& \ \sim Rbb$
T	F	T	T	$Raa \ \& \ \sim Rab \ \& \ Rba \ \& \ Rbb$
T	F	T	F	$Raa \ \& \ \sim Rab \ \& \ Rba \ \& \ \sim Rbb$
T	F	F	T	$Raa \ \& \ \sim Rab \ \& \ \sim Rba \ \& \ Rbb$
T	F	F	F	$Raa \ \& \ \sim Rab \ \& \ \sim Rba \ \& \ \sim Rbb$
F	T	T	T	$\sim Raa \ \& \ Rab \ \& \ Rba \ \& \ Rbb$
F	T	T	F	$\sim Raa \ \& \ Rab \ \& \ Rba \ \& \ \sim Rbb$
F	T	F	T	$\sim Raa \ \& \ Rab \ \& \ \sim Rba \ \& \ Rbb$
F	T	F	F	$\sim Raa \ \& \ Rab \ \& \ \sim Rba \ \& \ \sim Rbb$
F	F	T	T	$\sim Raa \ \& \ \sim Rab \ \& \ Rba \ \& \ Rbb$
F	F	T	F	$\sim Raa \ \& \ \sim Rab \ \& \ Rba \ \& \ \sim Rbb$
F	F	F	T	$\sim Raa \ \& \ \sim Rab \ \& \ \sim Rba \ \& \ Rbb$
F	F	F	F	$\sim Raa \ \& \ \sim Rab \ \& \ \sim Rba \ \& \ \sim Rbb$

Assuming finitely many constant symbols, predicate logics (monadic or relational) characterize finite Boolean algebras of propositions.

Overview of Deductive Logic I

- Deductive Logic provides *formal theories* of validity (*following-from*). The logician thus *theoretically grounds* our *informal* validity notion(s).

- In English, there are various argument forms or patterns that are intuitively or informally valid. Here's a simple, *sentential* example:

Dr. Ruth is a man.

- (1) If Dr. Ruth is a man, then Dr. Ruth is 10 feet tall.
 \therefore Dr. Ruth is 10 feet tall.

- Intuitively, the conclusion of (1) *follows-from* its premises, since *if* the premises of (1) *were* true, then (1)'s conclusion would *have to be* true.
- Our simplest logical theory (SL) correctly classifies this argument form (and many other valid English forms) as *valid* — an SL “success story”:

$$(1_{SL}) \quad \begin{array}{l} p. \\ \text{If } p, \text{ then } q. \\ \therefore q. \end{array}$$

Overview of Deductive Logic II

- However, there are many English arguments that are (intuitively, or “absolutely”) valid, but their SL forms are *not* valid. For instance:

$$(2) \quad \begin{array}{l} \text{Socrates is wise.} \\ \therefore \text{Someone is wise.} \end{array}$$

- Intuitively, argument (2) is (“absolutely”) *valid*. But, if we try to translate this argument into SL, we get the following *invalid* SL form:

$$(2_{SL}) \quad \begin{array}{l} p \\ \therefore q \end{array}$$

- This motivates the richer logical language MPL, which subsumes SL, and which adds additional structure that allows us to “see” its validity:

$$(2_{MPL}) \quad \begin{array}{l} Ws \\ \therefore (\exists x)Wx \end{array}$$

Overview of Deductive Logic III

- The relational predicate logical language is even richer than either SL or MPL. Indeed, it subsumes both SL and MPL. This language (RPL) can formalize many more English validities. Example:

$$(3) \quad \begin{array}{l} \text{Everybody loves John.} \\ \therefore \text{Someone is loved by everyone.} \end{array}$$

- Adequately formalizing this argument requires the use of a *two-place predicate/relation*: ‘ Lxy ’, which reads ‘ x loves y ’ or ‘ y is loved by x ’.

$$(3_{RPL}) \quad \begin{array}{l} (\forall x)Lxj \\ \therefore (\exists y)(\forall x)Lxy \end{array}$$

- Exercise: try symbolizing (3) in MPL. You’ll get something like this:

$$(3_{MPL?}) \quad \begin{array}{l} (\forall x)Jx \\ \therefore (\exists x)Ex \end{array}$$

- We will not use RPL much in this course. Mainly, we’ll use just SL/MPL.

Overview of Deductive Logic IV

- Logical theories replace imprecise, informal notions of following-from with precise, theoretical validity concepts (in formal logical languages).

– **One Informal Validity Notion.**

- * If \mathcal{A} 's premises are true, then \mathcal{A} 's conclusion must also be true.

– **Some Corresponding Theoretical Validity Concepts.**

- * An English argument \mathcal{A} is *SL-valid* if there is no SL interpretation in which all of the premises of \mathcal{A}_{SL} are T and the conclusion of \mathcal{A}_{SL} is F, where \mathcal{A}_{SL} is the *SL form* of the English argument \mathcal{A} .
- * An English argument \mathcal{A} is *MPL-valid* if there is no MPL interpretation in which all of the premises of \mathcal{A}_{MPL} are T and the conclusion of \mathcal{A}_{MPL} is F, where \mathcal{A}_{MPL} is the *MPL form* of \mathcal{A} .
- * Similarly for *RPL-validity* ...

- The story of formal deductive logic does not end with RPL. . .

Overview of Deductive Logic V

- The full theory of first-order logic (LFOL) includes RPL, plus n -place predicates, the identity relation $=$, and also function symbols.
- LFOL can capture even more valid arguments than RPL. For instance, LFOL can capture arguments like the following mathematical one:

$$2 + 4 = 6$$
 (4) $3 \times 2 = 6$

$$\therefore 2 + 4 = 3 \times 2$$
- Indeed, LFOL can capture just about any argument in just about any branch of modern mathematics. That's a lot of expressive power.
- In PHIL 140A, we study the full theory of first-order logic (LFOL). There, we give a semantics for LFOL, and we show that there is a sound and complete proof theory for LFOL (but, no decision procedure for \models !).
- Of course, even full first-order logic (LFOL) has its limitations...

Overview of Deductive Logic VI

- Some arguments involve quantification over not only objects but *properties*. These arguments are *second-order* and \therefore beyond LFOL.
- Leibniz (sometimes) talked as if the following argument were valid:

(5) a and b have exactly the same (monadic) properties.
 $\therefore a$ and b are identical.
- In second-order logic (SOL), (5) would be formalized as follows:

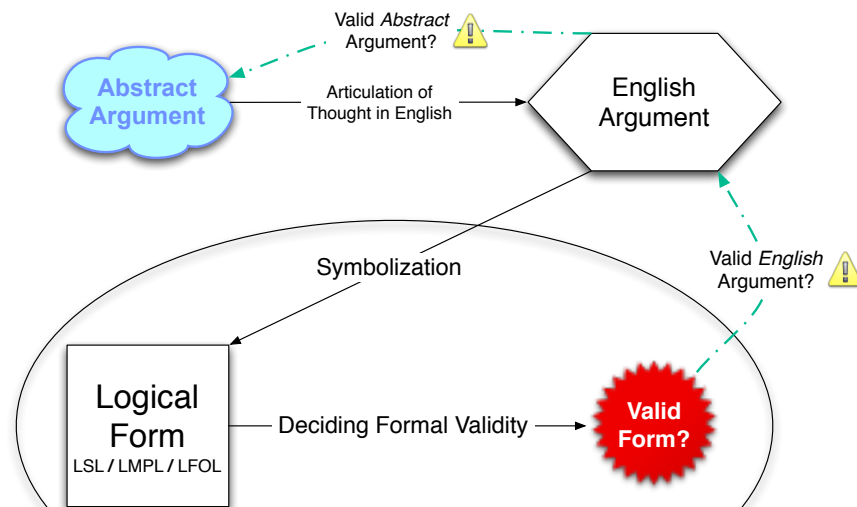
$$(5_{SOL}) (\forall P)(Pa \leftrightarrow Pb).$$

$$\therefore a = b.$$
- Note that the premise of (5) quantifies over (monadic) *predicates*.
- This is something that LFOL is not designed to do.
- We could also have an SOL which allows quantification over *relations*.
- Second-order logic is beyond 140A. It is touched upon (a little) in 140B.

Overview of Deductive Logic VII

- All the logics I've mentioned are *classical* deductive logics. Not all logicians think classical logics capture our intuitive validity notions.
- Classical logics all share the following two properties:
 - (i) All arguments with contradictory premises (e.g., $p \ \& \ \sim p$) are valid.
 - (ii) All arguments with tautological conclusions (e.g., $p \ \vee \ \sim p$) are valid.
- Some logicians think (i) and/or (ii) are *counterexamples* to the classical theory of validity (as an explication of our informal "following-from").
- Such logicians propose alternative formal theories of validity (\models^*).
- Usually, non-classical logicians reject the classical (truth-functional) theory of the *conditional*. They adopt a non-classical conditional (\rightarrow^*) which obeys constraints like the deduction theorem (relative to \models^*).

$$p \models^* q \text{ if and only if } \models^* p \rightarrow^* q$$
- These and other foundational questions about deductive logic are addressed in Philosophical Logic (142). We'll see *some* overlap here...



Why study logic *formally* or *symbolically*?

- In ordinary contexts, we want to know if arguments expressed in *English* are valid or invalid. But, in formal logic, we only study arguments expressible in *formal* languages (SL, MPL, etc.). *Why?*
- Analogous question: What we want from natural science is to understand natural systems. But, our theories (strictly) apply only to systems faithfully describable in mathematical terms.
- Although formal models are *idealizations* which abstract away some aspects of natural systems, they are *useful idealizations* that help us understand *many* natural relationships and regularities.
- Studying arguments expressible in formal languages allows us to develop and use powerful tools for testing validity, etc. We can't capture *all* valid arguments this way. But, we can grasp *many*.
- We will take the same attitude toward inductive logic as well ...

Inductive Logic — Basic Motivation and Ideas

- Intuitively, not all “logically good” arguments are deductively valid. Some invalid arguments seem (intuitively) logically *better than* others:

(6) p . Someone is wise.	(7) r . Someone is either wise or unwise.
$\therefore q$. Socrates is wise.	$\therefore q$. Socrates is wise.
- *Inductive* logic should *theoretically ground* our intuition that (6) is a *logically stronger* argument than (7) is. Neither argument is *valid*.
- More ambitiously, an inductive logician might aim for a theory of “the *degree* to which the premises of an argument *confirm* its conclusion”.
- This ambitious project would aim to characterize a *function* $c(\mathcal{C}, \mathcal{P})$. And, an intuitive requirement would be that this function be such that:

$$c(q, p) > c(q, r)$$
- This course is (mainly) about *inductive logic*. We will examine how *probabilities* might be used to *quantitatively generalize* deductive logic.

Logic and Epistemology — A Prelude I

- As I mentioned, some have worried about the adequacy of classical logic as a formal explication of our informal “following-from” relation.
- Here's a fact about classical deductive logic that may seem “odd”:

(†) If p and q are (classically) logically inconsistent, then the argument from p and q to r is (classically) valid — *for any* r .
- There's *something* “odd” about the fact that *everything follows-from inconsistent premises*, according to the classical formal explication of following-from. But, what, exactly, is supposed to be “odd” about it?
- Here's an *epistemological* principle that is downright *crazy*:

(‡) If one's beliefs are inconsistent (and one knows that they are), then one should believe everything (*i.e.*, every proposition).
- It is clear that (‡) is false. There are things I *know* to be false, and I shouldn't believe those things — no matter what else is true of me.

Logic and Epistemology — A Prelude II

- OK, (‡) is clearly false. So? What does that have to do with (†)?
- After all, (†) is about *logic*, and (‡) is about *epistemology*.
- Perhaps those worried about (†) are assuming that logic and epistemology are connected, or bridged by something like:

(*) If an agent S 's belief set B is such that $B \models p$ (and S knows that $B \models p$), then it would be reasonable for S to infer/believe p .
- If (*) were true, then (†) would imply (‡), and — as a result — classical logicians who accepted (*) would seem to be stuck with (‡) too.
- More precisely, classical logicians who believe (*) should find it reasonable to believe (‡). But, they don't (at least, they shouldn't!).
- ☞ But, *this* doesn't *force* classical logicians to give up (†). They could give up (*) instead. In such contexts, logic (alone) doesn't seem to tell us whether to infer something new, or reject something we already believe.