Philosophy 148 — Day 1: Introduction & Administration

• Administrative Stuff (*i.e.*, Syllabus)
  - Me & Raul (intros., personal data, office hours, etc.)
  - Prerequisites (Boolean logic, some simple algebra, no math phobia!)
  - Texts & Supplementary Readings (all online *via* website)
  - Requirements [Quiz (10), Assignments (30), Mid-Term (30), Final (30)]
  - Sections (determined this week, *via* index cards — meet next week)
    * Index Cards: Name, email, section time ranking. The 8 possible times are: Tu or Th: 9–10, 10–11, 1–2, or 2–3. Give a ranking of those among the 8 that you *can* do. Indicate those you *cannot* do.
  - Website (main source of course information — stay tuned!)
  - Tentative Schedule (somewhat loose, time-wise, but all readings set)
• Next: Brief Overview/Outline of the Course

Philosophy 148 — Day 1: Fundamental Underlying Questions

• I am writing a book on inductive logic (*a.k.a.*, confirmation theory).
• My main focus is on “quantitative generalizations” of deductive logic.
• The notion of *validity* is the deductive ideal for “logical goodness”.
• But, some invalid arguments seem “better”/“stronger” than others:
  - $P_1$. Someone is wise.
  - $P_2$. Someone is either wise or unwise.
  - $\therefore C_1$. Plato is wise.
  - $\therefore C_2$. Socrates is wise.
• The argument from $P_1$ to $C_1$ seems “better” than the one from $P_2$ to $C_2$.
• Is there a satisfying *explication* of this “better than” concept?
• And, if so, is this best understood a *logical* concept or an *epistemic* one or a *pragmatic* one, etc.? Moreover, if there is a *logical* “better than”, how is it related to *epistemology*? For that matter, how is *validity* related to epistemology? These are the sorts of questions in the air.

Philosophy 148 — Day 1: Course Overview/Outline

• The precise timing of the course is not fixed. But all readings are up.
• The *order* of topics in the course is also (more or less) set:
  - Review of Boolean Logic and Boolean Algebra [12A review + FBAs]
    * Propositional Logic
    * Monadic Predicate Logic
    * Finite Boolean Algebras [general logical framework for course]
  - Introduction of the (formal) Probability Calculus
    * Axiomatic Treatments
    * Algebraic Treatments
  - “Personalistic” Interpretations/Kinds of Probability
    * Pragmatic: betting odds / betting quotients / *rational* dob’s
    * Epistemic: degrees of *credence* / *justified* degrees of belief
  - Confirmation Theory and Inductive Logic
    * Deductive Approaches to Confirmation
      - Hempelian
      - Hypothetico-Deductive
    * Probabilistic Approaches to Confirmation
      - Logical (Carnapian)
      - Subjective/Personalistic (“Bayesian”)
  - The Paradoxes of Confirmation
    * The Raven Paradox
    * The Grue Paradox
  - Other Problems for Confirmation Theory (mainly, for “Bayesian” CT)
    * Old Evidence/Logical Omniscience/maybe others
  - Three *Psychological* Puzzles Involving Probability & Confirmation
    * The Base Rate Fallacy
    * The Conjunction Fallacy
    * The Wason Selection Task
Syntax of Sentential Logic (SL)

- The syntax of SL is simple. Its lexicon contains the following symbols:
  - Five sentential connectives (or sentential operators):

<table>
<thead>
<tr>
<th>Operator</th>
<th>Name</th>
<th>Logical Function</th>
<th>Used to translate</th>
</tr>
</thead>
<tbody>
<tr>
<td>‘∼’</td>
<td>tilde</td>
<td>negation</td>
<td>not, it is not the case that</td>
</tr>
<tr>
<td>‘&amp;’</td>
<td>ampersand</td>
<td>conjunction</td>
<td>and, also, moreover, but</td>
</tr>
<tr>
<td>‘∨’</td>
<td>vee</td>
<td>disjunction</td>
<td>or, either … or …</td>
</tr>
<tr>
<td>‘→’ (⊃)</td>
<td>arrow</td>
<td>conditional</td>
<td>if … then …, only if</td>
</tr>
<tr>
<td>‘↔’ (≡)</td>
<td>double arrow</td>
<td>biconditional</td>
<td>if and only if</td>
</tr>
</tbody>
</table>
- Parentheses ‘(’, ‘)’, brackets ‘[’, ‘]’, and braces ‘{’, ‘}’ for grouping.
- If a string of symbols contains anything other than these, it is not an SL sentence. And, only certain strings of these symbols are SL sentences.
- I assume you all know which SL strings are sentences and which are not.

Semantics of Sentential Logic: Truth Tables I

- Sentential Logic is truth-functional because the truth value of a compound $S$ is a function of the truth values of $S$’s atomic parts.
- All statement forms $p$ are defined by truth tables, which tell us how to determine the truth value of $p$’s from the truth values of $p$’s parts.
- Truth-tables provide a precise way of thinking about logical possibility. Each row of a truth-table can be thought of as a logical possibility. And, the actual world falls into exactly one of these rows/logical possibilities.
- In this sense, truth-tables provide a way to “see” logical space.
- Once we have an understanding of all the logically possible truth-values that and SL sentence can have (which truth-tables provide for us), testing the validity of SL arguments is easy — inspection of truth-tables!
- We just look for possible worlds (rows of the salient truth-table) in which all the premises are true and the conclusion is false.

Semantics of Sentential Logic: Truth Tables II

- We begin with negations, which have the simplest truth functions. The truth table for negation is as follows (we use T and F for true and false):

<table>
<thead>
<tr>
<th>$p$</th>
<th>~$p$</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>
- In words, this says that if $p$ is true than ~$p$ is false, and if $p$ is false, then ~$p$ is true. This is quite intuitive, and corresponds well to the English meaning of ‘not’. So, SL negation is like English negation.
- Examples:
  - It is not the case that Wagner wrote operas. (~$W$)
  - It is not the case that Picasso wrote operas. (~$P$)
- ‘~$W$’ is false, since ‘$W$’ is true, and ‘~$P$’ is true, since ‘$P$’ is false (like English).

Semantics of Sentential Logic: Truth Tables III

- Notice how we have four (4) rows in our truth table this time (not 2). There are four possible ways of assigning truth values to $p$ and $q$.
- The truth-functional definition of & is very close to the English ‘and’. A SL conjunction is true if both conjuncts are true; it’s false otherwise.
  - Monet and van Gogh were painters. ($M$ & $V$)
  - Monet and Beethoven were painters. ($M$ & $B$)
  - Beethoven and Einstein were painters. ($B$ & $E$)
- ‘$M$ & $V$’ is true, since both ‘$M$’ and ‘$V$’ are true. ‘$M$ & $B$’ is false, since ‘$B$’ is false. And, ‘$B$ & $E$’ is false, since ‘$B$’ and ‘$E$’ are both false (like English).
Semantics of Sentential Logic: Truth Tables IV

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>p ∨ q</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

- The truth-functional definition of ∨ is not as close to the English ‘or’. A SL disjunction is true if at least one disjunct is true; it’s false otherwise.
- In English, ‘A or B’ often implies that ‘A’ and ‘B’ are not both true. That is called exclusive or. In SL, ‘A ∨ B’ is not exclusive; it is inclusive (it is true if both disjuncts are true). We can express exclusive or in SL. How?
  - Either Jane Austen or René Descartes was a novelist. (J ∨ R)
  - Either Jane Austen or Charlotte Bronte was a novelist. (J ∨ C)
  - Either René Descartes or David Hume was a novelist. (R ∨ D)
- The first two disjunctions are true since at least one their disjuncts is true. The third disjunction is false, since both of its disjuncts are false.

Semantics of Sentential Logic: Truth Tables V

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>p → q</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

- The SL conditional (→) is farther from the English ‘only if’. An SL conditional is false iff its antecedent is true and its consequent is false.
- Consider the following English conditionals. [M = the moon is made of green cheese, O = life exists on other planets, and E = life exists on Earth]
  - If the moon is made of green cheese, then life exists on other planets. (M → O)
  - If life exists on other planets, then life exists on earth. (O → E)
- The SL translations of these sentences are both true.
  - ‘M → O’ is true because its antecedent ‘M’ is false.
  - ‘O → E’ is true because its consequent ‘E’ is true.
- This does not capture the English ‘if’. Remember: p → q ⊨∼p ∨ q.

Semantics of Sentential Logic: Truth Tables VI

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>p ↔ q</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>

- The SL biconditional ↔ inherits similar problems. An SL biconditional is true iff both of its components have the same truth value.
- Consider these two biconditionals. [M = the moon is made of green cheese, U = there are unicorns, E = life exists on Earth, and S = the sky is blue]
  - The moon is made of green cheese if and only if there are unicorns.
  - Life exists on earth if and only if the sky is blue.
- The SL translations of these sentences are both true.
  - ‘M ↔ U’ is true because M and U are false.
  - ‘E ↔ S’ is true because E and S are true.
- This does not capture the English ‘iff’. [p ↔ q ⊨ (p & q) ∨ (∼p & ∼q)]

Semantics of Sentential Logic: Truth Tables VII

- With the truth-table definitions of the five connectives in hand, we can now construct truth tables for arbitrary compound SL statements.
- A non-trivial example:

<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>r</th>
<th>(p &amp; (q ∨ r))</th>
<th>→</th>
<th>(p &amp; q)</th>
<th>∨</th>
<th>(p &amp; r)</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td></td>
<td>T</td>
<td></td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td></td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

- Thus, “(p & (q ∨ r)) → ((p & q) ∨ (p & r))” is a tautology.
### Logical Truth, Logical Falsity, and Contingency: Definitions

- **A statement is logically true (or tautologous)** if it is true regardless of the truth-values of its components. Example: \( p \rightarrow p \) is a tautology.
  
  \[
  \begin{array}{c|c}
  p & p \rightarrow p \\
  \hline
  T & T \\
  F & T \\
  \end{array}
  \]

- **A statement is logically false (or self-contradictory)** if it is false regardless of the truth-values of its components. Example: \( p \& \sim p \).
  
  \[
  \begin{array}{c|c|c}
  p & p \& \sim p \\
  \hline
  T & F & F \\
  F & F & F \\
  \end{array}
  \]

- **A statement is contingent** if its truth-value varies depending on the truth-values of its components. Example: \( A \) (or any atom) is contingent.
  
  \[
  \begin{array}{c|c}
  A & \sim A \\
  \hline
  T & T \\
  F & F \\
  \end{array}
  \]

### Equivalence, Contradictoriness, Consistency, and Inconsistency

- **Two statements are equivalent** (written \( p \equiv q \)) if they have the same truth-value in all possible worlds (i.e., in all rows of a simultaneous truth-table of both statements). For instance, \( A \rightarrow B \equiv \sim A \lor B \):
  
  \[
  \begin{array}{c|c|c|c}
  A & B & A \rightarrow B & \sim A \lor B \\
  \hline
  T & T & T & T \\
  T & F & F & F \\
  F & T & T & T \\
  F & F & F & F \\
  \end{array}
  \]

- **Two statements are contradictory** if they have opposite truth-values in all possible worlds (i.e., in all rows of a simultaneous truth-table of both statements). For instance, \( A \) and \( \sim A \):
  
  \[
  \begin{array}{c|c|c}
  A & \sim A \\
  \hline
  T & F \\
  F & T \\
  \end{array}
  \]

### Interpretations and Logical Equivalence

- **An interpretation** of an SL formula \( p \) is an assignment of truth-values to all of the sentence letters in \( p \).

- Each row of the truth-table of \( p \) is an interpretation of \( p \). Sometimes, I will also refer to rows of SL truth-tables as (logically) possible situations, or possible worlds.

- A tautology (contradiction) is an SL sentence whose truth value is \( T \) (\( F \)) on all of its interpretations (i.e., an SL sentence which is true (false) in all (logically) possible worlds).

- Two SL sentences are said to be logically equivalent if they have the same truth-value on all (joint) interpretations.

- I’ll abbreviate “\( p \) and \( q \) are logically equivalent” as “\( p \equiv q \)” (i.e., \( p \) follows from \( q \) (\( q \Rightarrow p \)), and \( q \) follows from \( p \) (\( p \Rightarrow q \))).
Logical Equivalence: Example #1

• I said that \( p \rightarrow q \) is logically equivalent to \( \sim p \lor q \).
• The following truth-table establishes this equivalence:

<table>
<thead>
<tr>
<th>( p )</th>
<th>( q )</th>
<th>( \sim p )</th>
<th>( \lor )</th>
<th>( p \rightarrow q )</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>

• The truth-tables of \( \sim p \lor q \) and \( p \rightarrow q \) are the same.

Logical Equivalence: Example #2

• \( p \leftrightarrow q \) is an abbreviation for \( (p \rightarrow q) \land (q \rightarrow p) \).
• The following truth-table shows it is a legitimate abbreviation:

<table>
<thead>
<tr>
<th>( p )</th>
<th>( q )</th>
<th>( p \rightarrow q )</th>
<th>( q \rightarrow p )</th>
<th>( p \leftrightarrow q )</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

• \( p \leftrightarrow q \) and \( (p \rightarrow q) \land (q \rightarrow p) \) have the same truth-table.

Some More Logical Equivalences

• Here is a simultaneous truth-table which establishes that

\[
A \rightarrow B \equiv (A \land B) \lor (\sim A \land \sim B)
\]

<table>
<thead>
<tr>
<th>( A )</th>
<th>( B )</th>
<th>( A \rightarrow B )</th>
<th>( (A \land B) \lor (\sim A \land \sim B) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
</tbody>
</table>

• Can you prove the following equivalences with truth-tables?
  - \( \sim (A \land B) \equiv \sim A \lor \sim B \)
  - \( \sim (A \lor B) \equiv \sim A \land \sim B \)
  - \( A \rightarrow (A \land B) \lor (A \land \sim B) \)
  - \( \sim A \equiv A \land (B \rightarrow \sim B) \)
  - \( \sim A \equiv A \lor (B \land \sim B) \)

Logical Equivalence, Contradictoriness, etc.: Relationships

• What are the relationships between “\( p \) and \( q \) are equivalent”, “\( p \) and \( q \) are consistent”, “\( p \) and \( q \) are contradictory”, “\( p \) and \( q \) are inconsistent”?

  
<table>
<thead>
<tr>
<th>Equivalent</th>
<th>Contradictory</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \downarrow )</td>
<td>( \uparrow )</td>
</tr>
<tr>
<td>Consistent</td>
<td>Inconsistent</td>
</tr>
</tbody>
</table>

• Answers:
  1. Equivalent \( \equiv \) Consistent \( (p \land \sim p \text{ and } q \land \sim q) \)
  2. Consistent \( \equiv \) Equivalent \( (p \rightarrow q \text{ and } p \land q) \)
  3. Contradictory \( \Rightarrow \) Inconsistent \( (\text{why?}) \)
  4. Inconsistent \( \equiv \) Contradictory \( (\text{example?}) \)
Truth-Tables and Deductive Validity I

- Remember, an argument is valid if it is impossible for its premises to be true while its conclusion is false. Let $p_1, \ldots, p_n$ be the premises of a SL argument, and let $q$ be the conclusion of the argument. Then, we have:

  \[
  \begin{array}{c}
p_1 \\
  \vdots \\
  p_n \\
  \vdots \\
  q
  \end{array}
  \]

  is valid if and only if there is no row in the simultaneous truth-table (interpretation) of $p_1, \ldots, p_n$, and $q$ which looks like:
  \[
  \begin{array}{c|c|c|c}
  \text{atoms} & \text{premises} & \text{conclusion} \\
  \hline
  \text{\ldots} & p_1 & \ldots & p_n \\
  \text{\ldots} & T & T & T & F
  \end{array}
  \]

Finite Propositional Boolean Algebras I

- A finite propositional Boolean algebra is a finite set of propositions which is closed under the logical operations and satisfies the laws of SL.

- Propositions are the things expressed by sentences (abstract entities, distinct from sentences). If two sentences are logically equivalent, then they express the same proposition. E.g., “$A \rightarrow B$” and “$\neg A \lor B$”.

- A set $S$ is closed under logical operations if applying a logical operation to a member (or pair of members) of $S$ always yields a member of $S$.

- Example: consider a sentential language with three atomic letters “$X$”, “$Y$”, and “$Z$”. The set of propositions expressible using the logical connectives and these three atomic letters forms a finite Boolean algebra.

- This Boolean algebra has $2^3 = 8$ atomic propositions or states (i.e., rows of a 3-sentence truth-table!). Question: How many propositions does it contain in total? Answer: $2^8 = 256$ (255 plus the contradiction). Why?

Truth-Tables and Deductive Validity II

- $A \rightarrow B$ is valid:
  \[
  \begin{array}{cccc}
  A & B & A & \rightarrow & B \\
  \hline
  T & T & T & T & T \\
  T & F & T & T & F \\
  F & T & F & F & T \\
  F & F & F & F & F
  \end{array}
  \]

- $A \rightarrow B$ is invalid:
  \[
  \begin{array}{cccc}
  A & B & A & \rightarrow & B \\
  \hline
  T & T & T & T & T \\
  T & F & T & T & T \\
  F & T & F & T & F \\
  F & F & F & F & F
  \end{array}
  \]

Finite Propositional Boolean Algebras II

- A literal is either an atomic sentence or the negation of an atomic sentence (e.g., “$A$” and “$\neg A$” are literals involving the atom “$A$”).

- A state of a Boolean algebra $B$ is a proposition expressed by a maximal conjunction of literals in a language $L_B$ describing $B$ (“maximal”: “containing exactly one literal for each atomic sentence in $B$”).

- Consider an algebra $B$ described by a 3-atom language $L_B$ (“$X$”, “$Y$”, “$Z$”). The states of $B$ are described by the $2^3 = 8$ state descriptions of $L_B$:

  \[
  \begin{array}{c}
  (s_1) \ X \& \ Y \& Z \\
  (s_2) \ X \& Y \& \neg Z \\
  (s_3) \ X \& \neg Y \& Z \\
  (s_4) \ X \& \neg Y \& \neg Z \\
  (s_5) \ \neg X \& Y \& Z \\
  (s_6) \ \neg X \& Y \& \neg Z \\
  (s_7) \ \neg X \& \neg Y \& Z \\
  (s_8) \ \neg X \& \neg Y \& \neg Z
  \end{array}
  \]
• We can “visualize” the states of \( B \) using a truth table or a Venn Diagram.

\[
\begin{array}{ccc|c}
X & Y & Z & \text{States} \\
\hline
T & T & T & s_1 \\
T & T & F & s_2 \\
T & F & T & s_3 \\
T & F & F & s_4 \\
F & T & T & s_5 \\
F & T & F & s_6 \\
F & F & T & s_7 \\
F & F & F & s_8 \\
\end{array}
\]

• Everything that can be expressed in the sentential language \( L_B \) can be expressed as a disjunction of state descriptions (think about why).

• Thus, every proposition expressible in \( L_B \) can be “visualized” simply by shading combinations of the 8 state-regions of the Venn Diagram of \( B \). It because of this that we can use Venn Diagrams to establish Boolean Laws.

• \( p \models q \) (in \( B \)) iff every shaded region in the Venn Diagram representation of \( p \) (in \( B \)) is also shaded in the Venn Diagram representation of \( q \) (in \( B \)).