PY1003
Lecture 8

Some points about terminology

- A **contradiction** is a sentence which not true on any interpretation.
- We can also describe such a sentence as **inconsistent**, and say that it **has no model**.
- A **set of sentences** is **inconsistent** just in case there is no interpretation which makes them all true. An inconsistent set of sentences, like a single inconsistent sentence, **has no model**.
- We would not describe such a set as ‘a contradiction’, since this term is reserved for single inconsistent sentences. However, we can say that the sentences in an inconsistent set, taken together, are **contradictory**.
- Note that one way for a set to be inconsistent is for it to contain an inconsistent sentence.

{A∧¬A, B} is an inconsistent set.
But this is not the only way for a set to be inconsistent. {A∧B, ¬A} is also an inconsistent set although every sentence in the set is consistent.

- A set of sentences is inconsistent iff the conjunction of all the sentences in the set is a contradiction. E.g. {A, ¬A} is inconsistent and A∧¬A is a contradiction; {A, B, C} is consistent and (A∧B)∧C is not a contradiction.

Reminder: exhibiting models

Starting sentence: ¬(A∧B)∧(A→B)

```
  ¬(A∧B)∧(A→B)
      ¬(A∧B)
       (A→B)
       /     \
    ¬A        ¬B
   /  \       /  \
 ¬A    B    ¬A    B
```

The tree is finished and has open branches, so this sentence is consistent.
The rightmost open branch tells us that one model is:
I(A)=F
I(B)=F

The middle open branch tells us that another is:
I(A)=F
I(B)=T

The leftmost open branch tells us that any interpretation which makes ¬A true (i.e. makes A false) will be a model – it doesn’t matter what value B takes. But we have already listed both the interpretations which make A false as models.

Compare the truth table:

	A	B	¬(A∧B)	A→B
1	T	T	F t	T
2	T	F	T f	F
3	F	T	T f	T
4	F	F	T f	T

Lines 3 and 4 represent the same two models for our starting set as we identified using the tree method.

Testing the validity of Sentential arguments with truth trees

● An argument is valid iff it is impossible for the premises to be true and the conclusion false
i.e. iff there is no interpretation which makes the premises true and the conclusion false
i.e. iff there is no interpretation which makes the premises true and the negation of the conclusion true.

● This means we can test whether an argument is valid by testing whether there is an interpretation which makes the premises true and the negation of the conclusion true
i.e. by testing whether the set consisting of the premises and the negation of the conclusion is a consistent set.

● For instance, we can test whether the argument
A→B, AvB ├ B is valid by testing whether {A→B, AvB, ¬B} is consistent. If the set is inconsistent, the argument is valid; there is no way for the premises to be true and the conclusion false. If the set is consistent, the argument is invalid: there is a way for the premises to be true and the conclusion false.

Example 1

Is the following argument valid?
A→B, AvB ├ B
To answer this, we test whether {A→B, AvB, ¬B} is consistent.

```
      A→B
      AvB
       ¬B
      /  \
    ¬A    B
    / \
   A   B
```

Every branch is closed, so the set comprising the premises plus the negated conclusion is inconsistent. Therefore the argument we started with is valid.

Example 2

Is the following argument valid?
A, A↔(BvC) ├ C
To answer this, we test whether {A, A↔(BvC), ¬C} is consistent.

```
           A
        A↔(BvC)
          ¬C
         /    \
        A      ¬A
   (BvC)      ¬(BvC)
    /  \
   B     C
```

The tree is finished and has an open branch, so the set is consistent. Therefore the argument we started with is invalid.

Exhibiting a counterexample to an argument

A counterexample to an argument is an interpretation which makes all the premises true and the conclusion false. A counterexample exists iff the argument is invalid.

The open branch on the tree above can be used to construct a model for the set of sentences {A, A↔(BvC), ¬C}:
I(A) = T
I(B) = T
I(C) = F
This interpretation is a counterexample to the argument A, A↔(BvC) ├ C. On this interpretation, all the premises are true and the conclusion is false.

Arguments from no premises

An argument is valid iff it is impossible for the premises to be true and the conclusion false. So an argument automatically counts as valid when it is impossible for the premises to be true (i.e. if the premises are inconsistent). An argument also automatically counts as valid when it is impossible for the conclusion to be false (i.e. if the conclusion is a tautology).

Sometimes we want to say that a sentence which cannot be false ‘follows from no premises’. What that means is that an argument with this sentence as its conclusion would be valid *whatever premises it had*. We express this by writing:

├ A→(AvB)

This is called 'an argument from no premises'. We can test this argument for validity, like any other, by testing whether it is impossible for its premises to be true and its conclusion false. Since there are no premises, however, this amounts to testing whether it is impossible for the conclusion to be false, i.e. whether the conclusion is a tautology.

The argument $\vdash A \rightarrow (A \vee B)$ is valid just in case the sentence $A \rightarrow (A \vee B)$ is a tautology.

$$
\begin{array}{c}
\neg(A \rightarrow (A \vee B)) \\
| \\
A \\
\neg(A \vee B) \\
| \\
\neg A \\
\underline{\neg B}
\end{array}
$$

<u>Summary of the procedure for testing for validity:</u>

1. List the premises (if any), followed by the negated conclusion.
2. Complete a truth tree for these sentences.
3. If all the branches close, the argument is valid.
4. If any branch remains open, the argument is invalid.
5. Open branches can be used to find counterexamples to the argument.