It's now time to give our formal language a semantics.

The meaning of not:

p	¬p
T	F
F	T

The meaning of or:

p	q	(p ∨ q)
T	T	T
T	F	T
F	T	T
F	F	F

The meaning of and:

p	q	(p ∧ q)
T	T	T
T	F	T
F	T	T
F	F	F

The meaning of iff:

p	q	(p ≡ q)
T	T	T
T	F	F
F	T	F
F	F	T

How do we account for the meaning of if then?

Like so

p	q	(p ⊃ q)
T	T	T
T	F	F
F	T	T
F	F	T

Three explanations.

(p ⊃ q) = ¬p ∨ q.

(p ⊃ q) = it's false that p is true but q is false; true otherwise = ¬(p ∧ ¬q)

And also this:

We want to p ⊃ q to license the following two inferences.

Modus Ponens

1. P ⊃ Q
2. P
3. So Q

Modus Tollens

1. P ⊃ Q
2. ¬Q
3. So ¬P

But we don't want our analysis of *if then* to license the following inference.

Affirming the consequent (No good)

1. P ⊃ Q
2. Q
3. So P

How do we set up the truth tables for P ⊃ Q such that we get MP, MT, but not Affirming the consequent?

For MP we need a truth table that looks like this.

p	q	(p ⊃ q)
T	T	T
T	F	F
F	T	?
F	F	?

Whenever (p ⊃ q) is true and p is true, we want q to also be true.

So we set (p ⊃ q) to be true when p is true and q is true, and false when p is true but q is false.

For MT we need a truth table that looks like this.

p	q	(p ⊃ q)
T	T	?
T	F	F
F	T	?
F	F	T

Whenever (p ⊃ q) is true and q is false, we want p to be false.

So for p is true and q is false, we want (p ⊃ q) to be false. And for p is false and q is false, we want (p ⊃ q) to be true.

So any two-place operator that licenses MP and MT will have the following truth table.

p	q	(p ⊃ q)
T	T	T
T	F	F
F	T	?
F	F	T

So the question becomes: what do we say when p is false but q is true?

If we let (p ⊃ q) be false when both p and q are false, then the only time (p ⊃ q) and q are both true, p is true. But that's the inference *affirming the consequent*, which is no good.

Notice also that defining (p ⊃ q) in this way will make (p ⊃ q) equivalent to (p ≡ q).

p	q	(p ⊃ q) (not correct)
T	T	T
T	F	F
F	T	F
F	F	T

So to avoid having (p ⊃ q) license (p ⊃ q), q, therefore p, we have to make it so that when p is false and q is true, (p ⊃ q) is true.

p	q	(p ⊃ q)
T	T	T
T	F	F
F	T	T
F	F	T

On this second way of doing things, when (p ⊃ q) is true and q is true, it's not forced that p is also true.

p	q	(p ⊃ q)
T	T	T
T	F	F
F	T	T
F	F	T

Hence the above is the correct semantics for ⊃.