3.29. Conjunctive and Disjunctive Normal Forms

1. Conjunctive Normal Form. We have seen that, armed with sentences in Disjunctive Normal Form (DNF), we can find an appropriate formal sentence for any given truth table. As noted earlier, since DNF sentences embedded basics within conjunctions, and those conjunctions within disjunctions, DNF imposes a **hierarchy of scope** on our three connectives. For in DNF a tilde takes only a sentence letter as its scope sentence; a conjunction takes only basics as scope; and a disjunction, with the widest scope of the three, takes those basics and conjunctions of basics as its scope.

But switching conjunctions and disjunctions in that hierarchy yields a different family of sentences in **Conjunctive Normal Form** (**CNF**). CNF embeds basics within disjunctions, and basics or disjunction of them within conjunctions. Here the tilde still has smallest scope (sentence letters); disjunctions have second smallest scope (basics); and conjunctions have the widest scope (basics or disjunctions of basics). Construction rules for CNF sentences are as follows.

Basics:

1. Sentence letters are basics.
2. Negations of sentence letters are basics.

Basic Disjunctions:

1. Basics are basic disjunctions.
2. If ● and ▲ are basic disjunctions,
   then (●∨▲) is a basic disjunction.

Sentences in Conjunctive Normal Form (CNF):

1. Basic Disjunctions are CNF sentences
2. If ● and ▲ are CNF sentences,
   then (●∧▲) is a CNF sentence.

The basics and basic disjunctions of old are in CNF. So all of the following are CNF sentences.

P	$(P \vee \sim P)$
$\sim P$	$(P \vee (\sim P \vee \sim Q))$
$(P \vee Q)$	$(\sim P \vee (\sim Q \vee R))$

Note that these also qualify as DNF sentences. Moreover, the basic conjunctions (however-many-barreled conjunctions of basics) which appeared in DNF also qualify as CNF sentences (again counting basics as mutant, one-part disjunctions). All of these are CNF sentences.

$(P \wedge \sim P)$

$(P \wedge Q \wedge \sim R)$

$(P \wedge Q \wedge \sim R \wedge \sim S)$

CNF and DNF part company over sentences with both wedges and vels. The following sentence, for example, qualifies as CNF, but not DNF.

$$(P \vee Q \vee R) \;\wedge\; (P \vee \sim Q \vee R) \;\wedge (\sim P \vee Q \vee \sim R)$$

Basics, basic conjunctions, and basic disjunctions thus form the common core of these two different sets of sentences.

Let us here introduce a bit of simplifying jargon: call the parts of a DNF sentence being disjoined together the **cell(s)** of that DNF sentence. So of the following DNF sentences the first sentence has three cells, the second has four cells, and the third (lacking any vels) is a one-celled DNF sentence.

$\underline{(P \wedge \sim P)} \vee \underline{(Q \wedge \sim Q)} \vee \underline{(R \wedge \sim R)}$

$\underline{(\sim P \wedge Q)} \vee \underline{(\sim Q \wedge R)} \vee \underline{(R \wedge S)} \vee \underline{(\sim S \vee P)}$

$P \wedge Q \wedge R \wedge S$

The parts of a CNF sentence being conjoined together will likewise count as cells. So the following are (respectively) a two-celled, a three-celled, and a one-celled CNF sentence.

$\underline{(\sim P \vee Q \vee R)} \wedge \underline{(P \vee Q \vee \sim R)}$
$\underline{(P \vee Q)} \wedge \underline{(\sim P \vee Q)} \wedge \underline{(\sim P \vee \sim Q)}$
$P \vee \sim P \vee Q$

Returning to our earlier point, about the overlap between these two families of sentences: the following counts as both a one-celled DNF sentence, and a four-celled CNF sentence.

$P \wedge Q \wedge R \wedge S$

Whereas the next sentence is both a three-celled DNF sentence and a one-celled CNF sentence.

$P \vee \sim P \vee Q$

2. Conjunctive Normal Form and Expressive Adequacy. By constructing a procedure to match each truth table with a CNF sentences, we can demonstrate that CNF also forms an expressively adequate family of sentences.

The procedure begins just as the DNF method did: attaching to a mystery truth table the truth tables for **N** many sentence letters – where the mystery truth table has $\mathbf{2^N}$ many valuations. So the following truth table, with 4 valuations, takes two sentence letters, “P” and “Q”.

P	**Q**	**?**
1	1	1
1	0	0
0	1	0
0	0	1

But now instead of picking out the 'true' valuations (those with a 1), we pick out the 'false' ones (those valuations with a 0). For each such 'false' valuation we construct an **anti-valuation sentence** – a disjunction – according to this procedure.

- If the sentence letter is true in that valuation, the anti-valuation sentence should **include the negation of that sentence letter**.

- If the sentence letter is false in that valuation, the anti-valuation sentence should **include that sentence letter**.

So for the second valuation the matching sentence is "(~P ∨ Q)," and for the third valuation "(P ∨ ~Q)".

P	**Q**	**~P**	**~Q**	**?**	**(~P ∨ Q)**	**(P ∨ ~Q)**
1	1	0	0	1	1	1
1	**0**	0	1	**0**	**0**	1
0	**1**	1	0	**0**	1	**0**
0	0	1	1	1	1	1

Conjoining these anti-valuation sentences yields a two-celled CNF sentence taking our mystery truth table.

P	**Q**	**~P**	**~Q**	**?**	**(~P ∨ Q)**	**(P ∨ ~Q)**	**(~P ∨ Q) ∧ (P ∨ ~Q)**
1	1	0	0	1	1	1	1
1	0	0	1	**0**	**0**	1	**0**
0	1	1	0	**0**	1	**0**	**0**
0	0	1	1	1	1	1	1

Generally each sentence letter at the left of the table will appear exactly once in each anti-valuation sentence – as in the above example. So anti-valuation sentences, and conjunctions of them, will cover any mystery truth table except one: where the truth table has 1 in **every** valuation. In that case, any logical truth – such as "(P ∨ ~P)" – will fill the bill. (Since "(P ∨ ~P)" counts as a one-celled CNF sentence, choice of this sentence does not take us outside CNF.)

Thus our general procedure for matching a CNF sentence to a given truth table works like so.

- If the truth table is false in exactly one valuation, build an anti-valuation sentence false in that valuation.

- If the truth table is false in more than one valuation, build an anti-valuation conjunction (a conjunction of anti-valuation sentences) false in those valuations.

- If the truth table is false in no valuation, use “(P ∨ ~P)” as the matching sentence.

As this procedure covers every possible type of truth table, CNF is shown to provide an **expressively adequate** family of sentences.

3. CNF Meets DNF. Since DNF is likewise an expressively adequate family of sentences, each truth table will be matched with both a CNF and a DNF sentence. That means each DNF sentence has a corresponding CNF sentence. And in fact we can, provided a CNF or DNF sentence, construct its counterpart without appeal to truth tables. Such a procedure relies solely on the following formal equivalence, noted in our earlier discussion of The 3D Method.

Distribution: “(● ∧ (▲ ∨ ◆))” ≡ “((● ∧ ▲) ∨ (● ∧ ◆))”
“(● ∨ (▲ ∧ ◆))” ≡ “((● ∨ ▲) ∧ (● ∨ ◆))”

So, for instance, we begin with the two-celled CNF sentence constructed above.

1. **(~P ∨ Q) ∧ (P ∨ ~Q)**

Distribution pushes the left part of this conjunction, with the wedge, into the right disjunction.

1. $\mathbf{\underline{(\sim P \vee Q) \wedge} (P \vee \sim Q)}$
2. $\mathbf{(\underline{(\sim P \vee Q) \wedge} P) \vee (\underline{(\sim P \vee Q) \wedge} \sim Q)}$

A second round of distribution pushes the right part of each conjunction into its neighboring disjunction.

1. $\mathbf{(\sim P \lor Q) \land (P \lor \sim Q)}$
2. $\mathbf{((\sim P \lor Q)\underline{\land P}) \lor ((\sim P \lor Q)\underline{\land \sim Q})}$
3. $\mathbf{((\sim P\underline{\land P)} \lor (Q\underline{\land P})) \lor ((\sim P\underline{\land \sim Q}) \lor (Q\underline{\land \sim Q}))}$

Sentence (3) is a four-celled DNF sentence. Extraneous parentheses can be left off.

3. $\mathbf{(\sim P \land P) \lor (Q \land P) \lor (\sim P \land \sim Q) \lor (Q \land \sim Q)}$

This DNF sentence does indeed take the same truth table as the original in CNF. (Since disjunction is wherever at least one of its parts are true, the disjunction is true in valuations 1 and 4.)

P	**Q**	**~P**	**~Q**	**(~P ∨ Q)**	**(P ∨ ~Q)**	**(~P ∨ Q) ∧ (P ∨ ~Q)**
1	1	0	0	1	1	**1**
1	0	0	1	0	1	**0**
0	1	1	0	1	0	**0**
0	0	1	1	1	1	**1**

(~P ∧ P)	**(P ∧ Q)**	**(~P ∧ ~Q)**	**(Q ∧ ~Q)**	**(~P ∧ P) ∨ (P ∧ Q) ∨ (~P ∧ ~Q) ∨ (Q ∧ ~Q)**
0	**1**	0	0	**1**
0	0	0	0	**0**
0	0	0	0	**0**
0	0	**1**	0	**1**

And since the Distribution Law is an equivalence, holding in both directions, we could just as easily started with a DNF sentence and convert it into CNF.

1. $\mathbf{(Q \land P) \lor (\sim P \land \sim Q)}$

Two applications of Distribution yield a four-celled sentence in CNF.

2. $\mathbf{((Q \land P) \lor \sim P) \land ((Q \land P) \lor \sim Q)}$
3. $\mathbf{(Q \lor \sim P) \land (P \lor \sim P) \land (Q \lor \sim Q) \land (P \lor \sim Q)}$

4. DNF and CNF: Further Semantic Features. We can simply some of the last results, as well as appreciating some general features of DNF and CNF sentences, by first noting a pair of points about some of the sentences service as their cells.

DNF sentences have as cells basics, and basic conjunctions built out of these. Now basic conjunctions follow the general semantic rule governing all conjunctions: **a conjunction is true only where all its parts are true**. But since no valuation makes both a sentence and its negation true, a basic conjunction containing a sentence letter and its negation will be true in no valuations – hence a contradiction.

> **A basic conjunction containing a sentence letter and the negation of that sentence letter is a contradiction**.

So we can tell that the following basic conjunctions are contradictions, without needing to appeal to truth tables.

> P ∧ ~Q ∧ R ∧ ~P
> R ∧ S ∧ ~R

That has two practical consequences for DNF sentences.

First, note that **a disjunction of some sentence with a contradiction is equivalent to that sentence alone**.

For example, “P ∨ (Q ∧ ~Q)” is equivalent to “P”.

P	**Q**	**~Q**	**(Q ∧ ~Q)**	**P ∨ (Q ∧ ~Q)**
1	1	0	0	**1**
1	0	1	0	**1**
0	1	0	0	**0**
0	0	1	0	**0**

That allows us to simplify sentences in DNF: any contradictory cell can be eliminated without change of meaning to that DNF sentence.

So earlier, beginning with the two-celled CNF sentence "(~P ∨ Q) ∧ (P ∨ ~Q)," we obtained the following four-celled DNF sentence via Distribution.

<u>(~P ∧ P)</u> ∨ (Q ∧ P) ∨ (~P ∧ ~Q) ∨ <u>(Q ∧ ~Q)</u>

But since the first and fourth cells here are contradictions, the sentence as a whole is logically equivalent to the following simpler DNF sentence.

(Q ∧ P) ∨ (~P ∧ ~Q)

And this simpler sentence is indeed equivalent to the original CNF sentence.

P	**Q**	**~P**	**~Q**	**(~P ∨ Q)**	**(P ∨ ~Q)**	**(~P ∨ Q) ∧ (P ∨ ~Q)**
1	1	0	0	1	1	**1**
1	0	0	1	0	1	**0**
0	1	1	0	1	0	**0**
0	0	1	1	1	1	**1**

(Q ∧ P)	**(~P ∧ ~Q)**	**(Q ∧ P) ∨ (~P ∧ ~Q)**
1	0	**1**
0	0	**0**
0	0	**0**
0	1	**1**

Second, if every cell of a DNF sentence is a contradiction, then the DNF sentence as a whole is a contradiction. For a disjunction is false only when all its parts are false; but with a disjunction all of whose cells are contradictions, every valuation makes all those parts false.

If every cell of a DNF sentence is a contradiction, then the whole DNF sentence is a contradiction.

So, for example, we know that the following DNF sentences are contradictions, without needing to construct a truth table for it.

> (P ∧ ~P ∧ Q) ∨ (~Q ∧ R ∧ Q) ∨ (P ∧ R ∧ ~P)
> (P ∧ X ∧ ~P) ∨ (~S ∧ S)
> (P ∧ ~P ∧ T)

Parallel morals hold for CNF sentences. Note that the cells of a CNF sentence are basics, or basic disjunctions. But a basic disjunction is a tautology just when it contains both a sentence letter and the negation of that sentence letter.

> **A basic disjunction containing a sentence letter and the negation of that sentence letter is a tautology**.

So even without building truth tables, we know that both of the following basic disjunctions are tautologies.

> P ∨ ~Q ∨ R ∨ ~P
> R ∨ S ∨ ~R

That spells two consequences for CNF sentences.

First, **the conjunction of some sentence with a tautology is logically equivalent to that sentence alone**. For example, “(P ∧ (Q ∨ ~Q))” is equivalent to “P”.

P	**Q**	**~Q**	**(Q ∧ ~Q)**	**P ∨ (Q ∧ ~Q)**
1	1	0	0	**1**
1	0	1	0	**1**
0	1	0	0	**0**
0	0	1	0	**0**

So any tautologous cell of a basic conjunction can be removed without changing the meaning of that basic conjunction. For instance, the first and third cells of the following conjunction are tautologies.

> **<u>(~P ∨ P)</u> ∨ (Q ∨ P) ∧ <u>(Q ∨ ~Q)</u> ∧ (~P ∨ ~Q)**

The whole sentence is thus logically equivalent to the following simpler conjunction.

$(Q \vee P) \wedge (\sim P \vee \sim Q)$

Second, if every cell of CNF sentence is a tautology, then the CNF sentence as a whole is a tautology. For a conjunction is only truth when all its parts are true. But if every cell of the sentence is a tautology, then every valuation will make the whole conjunction true.

> **If every cell of a CNF sentence is a tautology, then the whole CNF sentence is a tautology**.

That means, for instance, that all of the following are tautologies.

$(P \vee \sim P \vee Q) \wedge (\sim Q \vee R \vee Q) \wedge (P \vee R \vee \sim P)$
$(P \vee X \vee \sim P) \wedge (\sim S \vee S)$
$(P \vee \sim P \vee T)$

The last sentence brings up again the overlap between DNF and CNF. For "$(P \vee \sim P \vee T)$" is both a one-celled CNF sentence and a three-celled DNF sentence. So whereas we earlier stated a general result about all contradictions in DNF, we can now add the following.

> **If a DNF sentence contains no wedges, then it is a tautology just in case it contains a sentence letter and the negation of that sentence letter.**

Likewise our earlier contradictory one-celled DNF sentence is also a three-celled CNF sentence.

$(P \wedge \sim P \wedge T)$

And to our general observation about CNF tautologies we can now add this observation about CNF contradictions.

> **If a CNF sentence contains no vels, then it is a contradiction just in case it contains a sentence letter and the negation of that sentence letter.**

Now, every sentence in the Chapter Three formal language can be converted into DNF, and also into CNF, thanks to the 3D Method. And by way of these two conversions, we can determine whether the original sentence is a tautology, a contradiction, or neither.

- If the sentence converts into a DNF sentence whose every cell contains some sentence letter and its negation, then the sentence is a contradiction.
- If the sentence converts into a CNF sentence whose every cell contains some sentence letter and its negation, then the sentence is a tautology.
- If neither result occurs, then the sentence is neither a contradiction nor a tautology.

And once again, this general test will proceed entirely without appeal to truth tables or truth trees.

Summary: Conjunctive and Disjunctive Normal Forms

- A sentence in **Conjunctive Normal Form (CNF)** is a (however-many-place) conjunction of (however-many-place) disjunctions of basics**.**

- CNF forms an **expressively adequate** family of sentences.

- For each CNF sentence there is a logically equivalent sentence in **Disjunctive Normal Form (DNF)**.

- A sentence in DNF is a **contradiction** if (and only if) each of its cells contains a sentence letter and also the negation of that sentence letter.

- A sentence in CNF is a **tautology** if (and only if) each of its cells contains a sentence letter and also the negation of that sentence letter.