Summary

This unit discussed monotonicity as an abstract yet very important aspect of natural language. Monotonicity is a mathematical property of functions and breaks down into two distinct notions.

Definition 1. Let $f : S \to T$ be a function from S to T. Assume that $\leq_S$ is an ordering of elements of S, whereas $\leq_T$ is an ordering of the elements of T. Then f is **monotonically increasing** (or **isotonic**) iff $x \leq_S y$ implies $f(x) \leq_T f(y)$.

Definition 2. Let f, S, T, $\leq_S$, and $\leq_T$ be as before. Then f is **monotonically decreasing** (or **antitonic**) iff $x \leq_S y$ implies $f(x) \geq_T f(y)$.

Monotonicity is only of interest if the domain and co-domain of a function aren't just flat sets but have additional structure that puts the elements into an order. A monotonically increasing function preserves order, a monotonically decreasing function reverses it. The structure may be a poset, a join semilattice, a meet semilattice, or a lattice.

Various phenomena can be analyzed as involving monotonicity:

- the No-Crossing Branches constraint in Autosegmental Phonology,
- morphological syncretism (adjectival gradation, person pronouns, case), and
- NPI licensing based on left/right downward entailment, and
- the crosslinguistic typology of the Person Case Constraint, and
- the Adjunct island Constraint, and
- the learning of SL grammars, and
- many more that we didn't have time to discuss.

Monotonicity thus seems to permeate all aspects of language, from phonology and morphology to syntax, semantics, and even learning. It is a formal universal that can be combined with substantive universals (e.g. person hierarchies, adjunct algebras) to reduce overgeneration. It adds a new facet to our formal understanding. While a purely computational approach often fails to distinguish the linguistically natural from the unnatural, the combination of monotonicity and substantive universals can often capture the relevant distinction.