The (Downward) Löwenheim–Skolem Theorem states that every set of sentences with an infinite model has an enumerable model. It, too, is a consequence of the completeness theorem: the proof of completeness generates a model for any consistent set of sentences, and that model is enumerable. There is also an Upward Löwenheim–Skolem Theorem, which guarantees that if a set of sentences has a denumerable model it also has a non-enumerable model. Both theorems fail in second-order logic.

Theorem met.1. The Löwenheim–Skolem Theorem fails for second-order logic: There are sentences with infinite models but no enumerable models.

Proof. Recall that
\[ \text{Count} \equiv \exists z \exists u \forall X ((X(z) \land \forall x (X(x) \rightarrow X(u(x)))) \rightarrow \forall x X(x)) \]

is true in a structure \( \mathcal{M} \) iff \( |\mathcal{M}| \) is enumerable, so \( \neg \text{Count} \) is true in \( \mathcal{M} \) iff \( |\mathcal{M}| \) is non-enumerable. There are such structures—take any non-enumerable set as the domain, e.g., \( \mathcal{P}(\mathbb{N}) \) or \( \mathbb{R} \). So \( \neg \text{Count} \) has infinite models but no enumerable models. \( \Box \)

Theorem met.2. There are sentences with denumerable but no non-enumerable models.

Proof. \( \text{Count} \land \text{Inf} \) is true in \( \mathbb{N} \) but not in any structure \( \mathcal{M} \) with \( |\mathcal{M}| \) non-enumerable. \( \Box \)

Photo Credits

Bibliography