Limits on Expressibility in First-Order Logic

Soundness, Completeness and Compactness

Soundness, Completeness and Compactness

Recall:

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem.

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \vDash \alpha$ if and only if $\Gamma \vdash \alpha$.

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \vDash \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem.

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \vDash \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem. (Equivalent versions)

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \vDash \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem. (Equivalent versions)

1. If $\Gamma \vDash \alpha$, then $\Gamma_0 \vDash \alpha$ for some finite $\Gamma_0 \subseteq \Gamma$.

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \vDash \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem. (Equivalent versions)

1. If $\Gamma \vDash \alpha$, then $\Gamma_0 \vDash \alpha$ for some finite $\Gamma_0 \subseteq \Gamma$.

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \vDash \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem. (Equivalent versions)

1. If $\Gamma \vDash \alpha$, then $\Gamma_0 \vDash \alpha$ for some finite $\Gamma_0 \subseteq \Gamma$.
2. If Γ is unsatisfiable

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \vDash \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem. (Equivalent versions)

1. If $\Gamma \vDash \alpha$, then $\Gamma_0 \vDash \alpha$ for some finite $\Gamma_0 \subseteq \Gamma$.
2. If Γ is unsatisfiable

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \vDash \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem. (Equivalent versions)

1. If $\Gamma \vDash \alpha$, then $\Gamma_0 \vDash \alpha$ for some finite $\Gamma_0 \subseteq \Gamma$.
2. If Γ is unsatisfiable ($\Gamma \vDash \bot$),

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \vDash \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem. (Equivalent versions)

1. If $\Gamma \vDash \alpha$, then $\Gamma_0 \vDash \alpha$ for some finite $\Gamma_0 \subseteq \Gamma$.
2. If Γ is unsatisfiable ($\Gamma \vDash \bot$), then some finite subset $\Gamma_0 \subseteq \Gamma$ is unsatisfiable

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \vDash \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem. (Equivalent versions)

1. If $\Gamma \vDash \alpha$, then $\Gamma_0 \vDash \alpha$ for some finite $\Gamma_0 \subseteq \Gamma$.
2. If Γ is unsatisfiable ($\Gamma \vDash \bot$), then some finite subset $\Gamma_0 \subseteq \Gamma$ is unsatisfiable ($\Gamma_0 \vDash \bot$).

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \models \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem. (Equivalent versions)

1. If $\Gamma \models \alpha$, then $\Gamma_0 \models \alpha$ for some finite $\Gamma_0 \subseteq \Gamma$.
2. If Γ is unsatisfiable ($\Gamma \models \bot$), then some finite subset $\Gamma_0 \subseteq \Gamma$ is unsatisfiable ($\Gamma_0 \models \bot$).
3. If Γ is finitely satisfiable, then it is satisfiable.

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \models \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem. (Equivalent versions)

1. If $\Gamma \models \alpha$, then $\Gamma_0 \models \alpha$ for some finite $\Gamma_0 \subseteq \Gamma$.
2. If Γ is unsatisfiable ($\Gamma \models \bot$), then some finite subset $\Gamma_0 \subseteq \Gamma$ is unsatisfiable ($\Gamma_0 \models \bot$).
3. If Γ is finitely satisfiable, then it is satisfiable.

Soundness, Completeness and Compactness

Recall:

Soundness + Completeness Theorem. $\Gamma \models \alpha$ if and only if $\Gamma \vdash \alpha$.

Compactness Theorem. (Equivalent versions)

1. If $\Gamma \models \alpha$, then $\Gamma_0 \models \alpha$ for some finite $\Gamma_0 \subseteq \Gamma$.
2. If Γ is unsatisfiable ($\Gamma \models \bot$), then some finite subset $\Gamma_0 \subseteq \Gamma$ is unsatisfiable ($\Gamma_0 \models \bot$).
3. If Γ is finitely satisfiable, then it is satisfiable.

Let's discuss some consequences of the Compactness Theorem.

The class of infinite sets is first-order axiomatizable.

The class of infinite sets is first-order axiomatizable.

The sentence

The class of infinite sets is first-order axiomatizable.

The sentence

$$\sigma_{\geq 3}: \quad (\exists x_1)(\exists x_2)(\exists x_3)((x_1 \neq x_2) \wedge (x_1 \neq x_3) \wedge (x_2 \neq x_3)),$$

The class of infinite sets is first-order axiomatizable.

The sentence

$$\sigma_{\geq 3}: \quad (\exists x_1)(\exists x_2)(\exists x_3)((x_1 \neq x_2) \wedge (x_1 \neq x_3) \wedge (x_2 \neq x_3)),$$

which is a sentence in any language with equality,

The class of infinite sets is first-order axiomatizable.

The sentence

$$\sigma_{\geq 3}: \quad (\exists x_1)(\exists x_2)(\exists x_3)((x_1 \neq x_2) \wedge (x_1 \neq x_3) \wedge (x_2 \neq x_3)),$$

which is a sentence in any language with equality, expresses "there exists at least 3 elements".

The class of infinite sets is first-order axiomatizable.

The sentence

$$\sigma_{\geq 3}: \quad (\exists x_1)(\exists x_2)(\exists x_3)((x_1 \neq x_2) \wedge (x_1 \neq x_3) \wedge (x_2 \neq x_3)),$$

which is a sentence in any language with equality, expresses "there exists at least 3 elements". We can write similar sentences $\sigma_{\geq n}$ that express "there exists at least n elements".

The class of infinite sets is first-order axiomatizable.

The sentence

$$\sigma_{\geq 3}: \quad (\exists x_1)(\exists x_2)(\exists x_3)((x_1 \neq x_2) \wedge (x_1 \neq x_3) \wedge (x_2 \neq x_3)),$$

which is a sentence in any language with equality, expresses "there exists at least 3 elements". We can write similar sentences $\sigma_{\geq n}$ that express "there exists at least n elements". Let $\Sigma = \{\sigma_{\geq n} \mid n \in \mathbb{N}\}$.

The class of infinite sets is first-order axiomatizable.

The sentence

$$\sigma_{\geq 3}: \quad (\exists x_1)(\exists x_2)(\exists x_3)((x_1 \neq x_2) \wedge (x_1 \neq x_3) \wedge (x_2 \neq x_3)),$$

which is a sentence in any language with equality, expresses "there exists at least 3 elements". We can write similar sentences $\sigma_{\geq n}$ that express "there exists at least n elements". Let $\Sigma = \{\sigma_{\geq n} \mid n \in \mathbb{N}\}$. In any language L with equality, the class axiomatized by Σ is the class of infinite L-structures.

The class of infinite sets is first-order axiomatizable.

The sentence

$$\sigma_{\geq 3}: \quad (\exists x_1)(\exists x_2)(\exists x_3)((x_1 \neq x_2) \wedge (x_1 \neq x_3) \wedge (x_2 \neq x_3)),$$

which is a sentence in any language with equality, expresses "there exists at least 3 elements". We can write similar sentences $\sigma_{\geq n}$ that express "there exists at least n elements". Let $\Sigma = \{\sigma_{\geq n} \mid n \in \mathbb{N}\}$. In any language L with equality, the class axiomatized by Σ is the class of infinite L-structures. In particular, the class of infinite sets is an **axiomatic class**.

The class of finite sets is not first-order axiomatizable.

The class of finite sets is not first-order axiomatizable.

To obtain a contradiction to the claim in the title of this page,

The class of finite sets is not first-order axiomatizable.

To obtain a contradiction to the claim in the title of this page, assume that Δ is a set of first-order sentences in the language L of sets that axiomatizes the class of finite sets.

The class of finite sets is not first-order axiomatizable.

To obtain a contradiction to the claim in the title of this page, assume that Δ is a set of first-order sentences in the language L of sets that axiomatizes the class of finite sets. Expand the language to a language $L \cup C$ that includes an infinite set of new constants, $C = \{c_i \mid i \in \mathbb{N}\}$.

The class of finite sets is not first-order axiomatizable.

To obtain a contradiction to the claim in the title of this page, assume that Δ is a set of first-order sentences in the language L of sets that axiomatizes the class of finite sets. Expand the language to a language $L \cup C$ that includes an infinite set of new constants, $C = \{c_i \mid i \in \mathbb{N}\}$. Create a set Θ containing all sentences of the form

The class of finite sets is not first-order axiomatizable.

To obtain a contradiction to the claim in the title of this page, assume that Δ is a set of first-order sentences in the language L of sets that axiomatizes the class of finite sets. Expand the language to a language $L \cup C$ that includes an infinite set of new constants, $C = \{c_i \mid i \in \mathbb{N}\}$. Create a set Θ containing all sentences of the form

$$\theta_{ij} : \quad c_i \neq c_j.$$

The class of finite sets is not first-order axiomatizable.

To obtain a contradiction to the claim in the title of this page, assume that Δ is a set of first-order sentences in the language L of sets that axiomatizes the class of finite sets. Expand the language to a language $L \cup C$ that includes an infinite set of new constants, $C = \{c_i \mid i \in \mathbb{N}\}$. Create a set Θ containing all sentences of the form

$$\theta_{ij} : \quad c_i \neq c_j.$$

That is, $\Theta = \{\theta_{ij} \mid i < j\}$.

The class of finite sets is not first-order axiomatizable.

To obtain a contradiction to the claim in the title of this page, assume that Δ is a set of first-order sentences in the language L of sets that axiomatizes the class of finite sets. Expand the language to a language $L \cup C$ that includes an infinite set of new constants, $C = \{c_i \mid i \in \mathbb{N}\}$. Create a set Θ containing all sentences of the form

$$\theta_{ij} : \quad c_i \neq c_j.$$

That is, $\Theta = \{\theta_{ij} \mid i < j\}$.

Claim.

The class of finite sets is not first-order axiomatizable.

To obtain a contradiction to the claim in the title of this page, assume that Δ is a set of first-order sentences in the language L of sets that axiomatizes the class of finite sets. Expand the language to a language $L \cup C$ that includes an infinite set of new constants, $C = \{c_i \mid i \in \mathbb{N}\}$. Create a set Θ containing all sentences of the form

$$\theta_{ij} : \quad c_i \neq c_j.$$

That is, $\Theta = \{\theta_{ij} \mid i < j\}$.

Claim. $\Delta \cup \Theta$ is finitely satisfiable, but not satisfiable.

The class of finite sets is not first-order axiomatizable.

To obtain a contradiction to the claim in the title of this page, assume that Δ is a set of first-order sentences in the language L of sets that axiomatizes the class of finite sets. Expand the language to a language $L \cup C$ that includes an infinite set of new constants, $C = \{c_i \mid i \in \mathbb{N}\}$. Create a set Θ containing all sentences of the form

$$\theta_{ij} : \quad c_i \neq c_j.$$

That is, $\Theta = \{\theta_{ij} \mid i < j\}$.

Claim. $\Delta \cup \Theta$ is finitely satisfiable, but not satisfiable. (Check!)

The class of finite sets is not first-order axiomatizable.

To obtain a contradiction to the claim in the title of this page, assume that Δ is a set of first-order sentences in the language L of sets that axiomatizes the class of finite sets. Expand the language to a language $L \cup C$ that includes an infinite set of new constants, $C = \{c_i \mid i \in \mathbb{N}\}$. Create a set Θ containing all sentences of the form

$$\theta_{ij} : \quad c_i \neq c_j.$$

That is, $\Theta = \{\theta_{ij} \mid i < j\}$.

Claim. $\Delta \cup \Theta$ is finitely satisfiable, but not satisfiable. (Check!)

This contradiction to the Compactness Theorem shows that Δ cannot exist.

Similar results

Similar results

Let L be a language.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures,

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable,

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.
2. $\mathcal{K}$ is the class of 4-colorable graphs.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.
2. $\mathcal{K}$ is the class of 4-colorable graphs.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.
2. $\mathcal{K}$ is the class of 4-colorable graphs.
3. $\mathcal{K}$ is the class of posets.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.
2. $\mathcal{K}$ is the class of 4-colorable graphs.
3. $\mathcal{K}$ is the class of posets.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.
2. $\mathcal{K}$ is the class of 4-colorable graphs.
3. $\mathcal{K}$ is the class of posets.
4. $\mathcal{K}$ is the class of totally ordered posets.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.
2. $\mathcal{K}$ is the class of 4-colorable graphs.
3. $\mathcal{K}$ is the class of posets.
4. $\mathcal{K}$ is the class of totally ordered posets.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.
2. $\mathcal{K}$ is the class of 4-colorable graphs.
3. $\mathcal{K}$ is the class of posets.
4. $\mathcal{K}$ is the class of totally ordered posets.
5. $\mathcal{K}$ is the class of groups.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.
2. $\mathcal{K}$ is the class of 4-colorable graphs.
3. $\mathcal{K}$ is the class of posets.
4. $\mathcal{K}$ is the class of totally ordered posets.
5. $\mathcal{K}$ is the class of groups.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.
2. $\mathcal{K}$ is the class of 4-colorable graphs.
3. $\mathcal{K}$ is the class of posets.
4. $\mathcal{K}$ is the class of totally ordered posets.
5. $\mathcal{K}$ is the class of groups.
6. $\mathcal{K}$ is the class of abelian groups.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.
2. $\mathcal{K}$ is the class of 4-colorable graphs.
3. $\mathcal{K}$ is the class of posets.
4. $\mathcal{K}$ is the class of totally ordered posets.
5. $\mathcal{K}$ is the class of groups.
6. $\mathcal{K}$ is the class of abelian groups.

Similar results

Let L be a language. Let Σ be a set of sentences that axiomatize some class $\mathcal{K}$ of L-structures. If $\mathcal{K}$ contains arbitrarily large finite structures, then the class of infinite structures in $\mathcal{K}$ is nonempty and axiomatizable, but the class of finite structures in $\mathcal{K}$ is not axiomatizable.

Examples.

1. $\mathcal{K}$ is the class of graphs.
2. $\mathcal{K}$ is the class of 4-colorable graphs.
3. $\mathcal{K}$ is the class of posets.
4. $\mathcal{K}$ is the class of totally ordered posets.
5. $\mathcal{K}$ is the class of groups.
6. $\mathcal{K}$ is the class of abelian groups.

(Check!)

Upward Löwenheim-Skolem Theorem

Upward Löwenheim-Skolem Theorem

Theorem.

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures.

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure,

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$,

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$, then $\mathcal{K}$ contains an infinite structure $\mathbf{A}_\kappa$ of cardinality κ for every $\kappa \geq |L|$.

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$, then $\mathcal{K}$ contains an infinite structure $\mathbf{A}_\kappa$ of cardinality κ for every $\kappa \geq |L|$.
(In fact, the full L-S Theorem produces $\mathbf{A}_\kappa$ satisfying $|\mathbf{A}_\kappa| = \kappa$ and $\mathbf{A} \prec \mathbf{A}_\kappa$.)

(Sketch argument!)

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$, then $\mathcal{K}$ contains an infinite structure $\mathbf{A}_\kappa$ of cardinality κ for every $\kappa \geq |L|$.
(In fact, the full L-S Theorem produces $\mathbf{A}_\kappa$ satisfying $|\mathbf{A}_\kappa| = \kappa$ and $\mathbf{A} \prec \mathbf{A}_\kappa$.)

(Sketch argument!)

Corollary.

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$, then $\mathcal{K}$ contains an infinite structure $\mathbf{A}_\kappa$ of cardinality κ for every $\kappa \geq |L|$.
(In fact, the full L-S Theorem produces $\mathbf{A}_\kappa$ satisfying $|\mathbf{A}_\kappa| = \kappa$ and $\mathbf{A} \prec \mathbf{A}_\kappa$.)

(Sketch argument!)

Corollary. No infinite structure is describable up to isomorphism in its first-order language.

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$, then $\mathcal{K}$ contains an infinite structure $\mathbf{A}_\kappa$ of cardinality κ for every $\kappa \geq |L|$.
(In fact, the full L-S Theorem produces $\mathbf{A}_\kappa$ satisfying $|\mathbf{A}_\kappa| = \kappa$ and $\mathbf{A} \prec \mathbf{A}_\kappa$.)

(Sketch argument!)

Corollary. No infinite structure is describable up to isomorphism in its first-order language.

Special case (Cor).

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$, then $\mathcal{K}$ contains an infinite structure $\mathbf{A}_\kappa$ of cardinality κ for every $\kappa \geq |L|$.
(In fact, the full L-S Theorem produces $\mathbf{A}_\kappa$ satisfying $|\mathbf{A}_\kappa| = \kappa$ and $\mathbf{A} \prec \mathbf{A}_\kappa$.)

(Sketch argument!)

Corollary. No infinite structure is describable up to isomorphism in its first-order language.

Special case (Cor). The ordered field $\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$ is not describable up to isomorphism in its first-order language.

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$, then $\mathcal{K}$ contains an infinite structure $\mathbf{A}_\kappa$ of cardinality κ for every $\kappa \geq |L|$.
(In fact, the full L-S Theorem produces $\mathbf{A}_\kappa$ satisfying $|\mathbf{A}_\kappa| = \kappa$ and $\mathbf{A} \prec \mathbf{A}_\kappa$.)

(Sketch argument!)

Corollary. No infinite structure is describable up to isomorphism in its first-order language.

Special case (Cor). The ordered field $\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$ is not describable up to isomorphism in its first-order language.

Special case (L-S).

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$, then $\mathcal{K}$ contains an infinite structure $\mathbf{A}_\kappa$ of cardinality κ for every $\kappa \geq |L|$.
(In fact, the full L-S Theorem produces $\mathbf{A}_\kappa$ satisfying $|\mathbf{A}_\kappa| = \kappa$ and $\mathbf{A} \prec \mathbf{A}_\kappa$.)

(Sketch argument!)

Corollary. No infinite structure is describable up to isomorphism in its first-order language.

Special case (Cor). The ordered field $\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$ is not describable up to isomorphism in its first-order language.

Special case (L-S). The class of all connected graphs

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$, then $\mathcal{K}$ contains an infinite structure $\mathbf{A}_\kappa$ of cardinality κ for every $\kappa \geq |L|$.
(In fact, the full L-S Theorem produces $\mathbf{A}_\kappa$ satisfying $|\mathbf{A}_\kappa| = \kappa$ and $\mathbf{A} \prec \mathbf{A}_\kappa$.)

(Sketch argument!)

Corollary. No infinite structure is describable up to isomorphism in its first-order language.

Special case (Cor). The ordered field $\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$ is not describable up to isomorphism in its first-order language.

Special case (L-S). The class of all connected graphs ($G = \langle V; E(x, y) \rangle$)

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$, then $\mathcal{K}$ contains an infinite structure $\mathbf{A}_\kappa$ of cardinality κ for every $\kappa \geq |L|$.
(In fact, the full L-S Theorem produces $\mathbf{A}_\kappa$ satisfying $|\mathbf{A}_\kappa| = \kappa$ and $\mathbf{A} \prec \mathbf{A}_\kappa$.)

(Sketch argument!)

Corollary. No infinite structure is describable up to isomorphism in its first-order language.

Special case (Cor). The ordered field $\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$ is not describable up to isomorphism in its first-order language.

Special case (L-S). The class of all connected graphs ($G = \langle V; E(x, y) \rangle$) is not axiomatizable in its first-order language.

Upward Löwenheim-Skolem Theorem

Theorem. Let $\mathcal{K}$ be an axiomatic class of L-structures. If $\mathcal{K}$ contains an infinite structure, $\mathbf{A}$, then $\mathcal{K}$ contains an infinite structure $\mathbf{A}_\kappa$ of cardinality κ for every $\kappa \geq |L|$.
(In fact, the full L-S Theorem produces $\mathbf{A}_\kappa$ satisfying $|\mathbf{A}_\kappa| = \kappa$ and $\mathbf{A} \prec \mathbf{A}_\kappa$.)

(Sketch argument!)

Corollary. No infinite structure is describable up to isomorphism in its first-order language.

Special case (Cor). The ordered field $\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$ is not describable up to isomorphism in its first-order language.

Special case (L-S). The class of all connected graphs ($G = \langle V; E(x, y) \rangle$) is not axiomatizable in its first-order language.

Nonstandard natural numbers cannot be prohibited

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle$

Nonstandard natural numbers cannot be prohibited

$$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$$

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

To see this, let $\Delta = \mathrm{Th}(\mathbb{N})$ be the set of all sentences in the language L of the natural numbers that hold in $\mathbb{N}$.

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

To see this, let $\Delta = \mathrm{Th}(\mathbb{N})$ be the set of all sentences in the language L of the natural numbers that hold in $\mathbb{N}$. (Δ = the **theory** of $\mathbb{N}$.)

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

To see this, let $\Delta = \mathrm{Th}(\mathbb{N})$ be the set of all sentences in the language L of the natural numbers that hold in $\mathbb{N}$. (Δ = the **theory** of $\mathbb{N}$.) Let c be a new constant symbol.

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

To see this, let $\Delta = \mathrm{Th}(\mathbb{N})$ be the set of all sentences in the language L of the natural numbers that hold in $\mathbb{N}$. (Δ = the **theory** of $\mathbb{N}$.) Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_i \mid i \in \mathbb{N}\}$ and

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

To see this, let $\Delta = \mathrm{Th}(\mathbb{N})$ be the set of all sentences in the language L of the natural numbers that hold in $\mathbb{N}$. (Δ = the **theory** of $\mathbb{N}$.) Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_i \mid i \in \mathbb{N}\}$ and

$$\theta_i : \quad S^i(0) \neq c.$$

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

To see this, let $\Delta = \mathrm{Th}(\mathbb{N})$ be the set of all sentences in the language L of the natural numbers that hold in $\mathbb{N}$. (Δ = the **theory** of $\mathbb{N}$.) Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_i \mid i \in \mathbb{N}\}$ and

$$\theta_i : \quad S^i(0) \neq c.$$

Claim.

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

To see this, let $\Delta = \mathrm{Th}(\mathbb{N})$ be the set of all sentences in the language L of the natural numbers that hold in $\mathbb{N}$. (Δ = the **theory** of $\mathbb{N}$.) Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_i \mid i \in \mathbb{N}\}$ and

$$\theta_i : \quad S^i(0) \neq c.$$

Claim. $\Delta \cup \Theta$ is finitely satisfiable.

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

To see this, let $\Delta = \mathrm{Th}(\mathbb{N})$ be the set of all sentences in the language L of the natural numbers that hold in $\mathbb{N}$. (Δ = the **theory** of $\mathbb{N}$.) Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_i \mid i \in \mathbb{N}\}$ and

$$\theta_i : \quad S^i(0) \neq c.$$

Claim. $\Delta \cup \Theta$ is finitely satisfiable.

(Check!)

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

To see this, let $\Delta = \mathrm{Th}(\mathbb{N})$ be the set of all sentences in the language L of the natural numbers that hold in $\mathbb{N}$. (Δ = the **theory** of $\mathbb{N}$.) Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_i \mid i \in \mathbb{N}\}$ and

$$\theta_i : \quad S^i(0) \neq c.$$

Claim. $\Delta \cup \Theta$ is finitely satisfiable.

(Check!)

By Compactness, there is a model of $\Delta \cup \Theta$.

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

To see this, let $\Delta = \mathrm{Th}(\mathbb{N})$ be the set of all sentences in the language L of the natural numbers that hold in $\mathbb{N}$. (Δ = the **theory** of $\mathbb{N}$.) Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_i \mid i \in \mathbb{N}\}$ and

$$\theta_i : \quad S^i(0) \neq c.$$

Claim. $\Delta \cup \Theta$ is finitely satisfiable.

(Check!)

By Compactness, there is a model of $\Delta \cup \Theta$. Any such structure satisfies every L-sentence true of $\mathbb{N}$,

Nonstandard natural numbers cannot be prohibited

$\mathbb{N} = \langle \{0, 1, 2, \ldots\}; 0, S(x) \rangle = \langle \{0, S(0), SS(0), \ldots\}; 0, S(x) \rangle.$

First-order logic is not strong enough to say that every natural number is obtained from 0 by applying the successor function a finite number of times.

To see this, let $\Delta = \mathrm{Th}(\mathbb{N})$ be the set of all sentences in the language L of the natural numbers that hold in $\mathbb{N}$. (Δ = the **theory** of $\mathbb{N}$.) Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_i \mid i \in \mathbb{N}\}$ and

$$\theta_i : \quad S^i(0) \neq c.$$

Claim. $\Delta \cup \Theta$ is finitely satisfiable.

(Check!)

By Compactness, there is a model of $\Delta \cup \Theta$. Any such structure satisfies every L-sentence true of $\mathbb{N}$, and also has a "nonstandard" element c that cannot be obtained by applying the successor function to 0 a finite number of times.

Nonzero infinitesimal real numbers cannot be prohibited

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

First-order logic is not strong enough to say that $\mathbb{R}$ has no infinitely large positive elements or infinitely small positive elements

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

First-order logic is not strong enough to say that $\mathbb{R}$ has no infinitely large positive elements or infinitely small positive elements (= infinitesimals).

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

First-order logic is not strong enough to say that $\mathbb{R}$ has no infinitely large positive elements or infinitely small positive elements (= infinitesimals).

To see this, let $\Delta = \text{Th}(\mathbb{R})$ be the set of all sentences in the language L of the real numbers that hold in $\mathbb{R}$.

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

First-order logic is not strong enough to say that $\mathbb{R}$ has no infinitely large positive elements or infinitely small positive elements (= infinitesimals).

To see this, let $\Delta = \text{Th}(\mathbb{R})$ be the set of all sentences in the language L of the real numbers that hold in $\mathbb{R}$. Let c be a new constant symbol.

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

First-order logic is not strong enough to say that $\mathbb{R}$ has no infinitely large positive elements or infinitely small positive elements (= infinitesimals).

To see this, let $\Delta = \text{Th}(\mathbb{R})$ be the set of all sentences in the language L of the real numbers that hold in $\mathbb{R}$. Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_n \mid n \in \mathbb{N}\}$ and

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

First-order logic is not strong enough to say that $\mathbb{R}$ has no infinitely large positive elements or infinitely small positive elements (= infinitesimals).

To see this, let $\Delta = \text{Th}(\mathbb{R})$ be the set of all sentences in the language L of the real numbers that hold in $\mathbb{R}$. Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_n \mid n \in \mathbb{N}\}$ and

$$\theta_n : \quad n < c.$$

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

First-order logic is not strong enough to say that $\mathbb{R}$ has no infinitely large positive elements or infinitely small positive elements (= infinitesimals).

To see this, let $\Delta = \text{Th}(\mathbb{R})$ be the set of all sentences in the language L of the real numbers that hold in $\mathbb{R}$. Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_n \mid n \in \mathbb{N}\}$ and

$$\theta_n : \quad n < c.$$

Claim.

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

First-order logic is not strong enough to say that $\mathbb{R}$ has no infinitely large positive elements or infinitely small positive elements (= infinitesimals).

To see this, let $\Delta = \text{Th}(\mathbb{R})$ be the set of all sentences in the language L of the real numbers that hold in $\mathbb{R}$. Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_n \mid n \in \mathbb{N}\}$ and

$$\theta_n : \quad n < c.$$

Claim. $\Delta \cup \Theta$ is finitely satisfiable.

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

First-order logic is not strong enough to say that $\mathbb{R}$ has no infinitely large positive elements or infinitely small positive elements (= infinitesimals).

To see this, let $\Delta = \text{Th}(\mathbb{R})$ be the set of all sentences in the language L of the real numbers that hold in $\mathbb{R}$. Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_n \mid n \in \mathbb{N}\}$ and

$$\theta_n : \quad n < c.$$

Claim. $\Delta \cup \Theta$ is finitely satisfiable.

(Check!)

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

First-order logic is not strong enough to say that $\mathbb{R}$ has no infinitely large positive elements or infinitely small positive elements (= infinitesimals).

To see this, let $\Delta = \text{Th}(\mathbb{R})$ be the set of all sentences in the language L of the real numbers that hold in $\mathbb{R}$. Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_n \mid n \in \mathbb{N}\}$ and

$$\theta_n : \quad n < c.$$

Claim. $\Delta \cup \Theta$ is finitely satisfiable.

(Check!)

Any structure that satisfies $\Delta \cup \Theta$ will be elementarily equivalent to $\mathbb{R}$,

Nonzero infinitesimal real numbers cannot be prohibited

$\mathbb{R} = \langle \{\text{reals}\}; +, -, 0, \cdot, 1, < \rangle$

First-order logic is not strong enough to say that $\mathbb{R}$ has no infinitely large positive elements or infinitely small positive elements (= infinitesimals).

To see this, let $\Delta = \text{Th}(\mathbb{R})$ be the set of all sentences in the language L of the real numbers that hold in $\mathbb{R}$. Let c be a new constant symbol. In the language $L \cup \{c\}$, consider $\Delta \cup \Theta$ where $\Theta = \{\theta_n \mid n \in \mathbb{N}\}$ and

$$\theta_n : \quad n < c.$$

Claim. $\Delta \cup \Theta$ is finitely satisfiable.

(Check!)

Any structure that satisfies $\Delta \cup \Theta$ will be elementarily equivalent to $\mathbb{R}$, but will have an infinitely large positive element c and an infinitely small positive element $\epsilon = \frac{1}{c}$ satisfying $0 < \epsilon < \frac{1}{n}$ for every positive natural number n.

No Completeness Theorem for Second-Order Logic

No Completeness Theorem for Second-Order Logic

We have seen that

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness

and

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness
and
Compactness implies Inexpressibility.

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness
and
Compactness implies Inexpressibility.

If we write these implications in the contrapositive form,

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness
and
Compactness implies Inexpressibility.

If we write these implications in the contrapositive form, we get

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness
and
Compactness implies Inexpressibility.

If we write these implications in the contrapositive form, we get

Expressibility implies Compactness fails

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness
and
Compactness implies Inexpressibility.

If we write these implications in the contrapositive form, we get

Expressibility implies Compactness fails
and

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness
and
Compactness implies Inexpressibility.

If we write these implications in the contrapositive form, we get

Expressibility implies Compactness fails
and
Compactness fails implies Completeness fails.

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness
and
Compactness implies Inexpressibility.

If we write these implications in the contrapositive form, we get

Expressibility implies Compactness fails
and
Compactness fails implies Completeness fails.

Let's sketch the reasons why second-order logic allows us to accomplish the following:

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness
and
Compactness implies Inexpressibility.

If we write these implications in the contrapositive form, we get

Expressibility implies Compactness fails
and
Compactness fails implies Completeness fails.

Let's sketch the reasons why second-order logic allows us to accomplish the following:

1. axiomatize the class of connected graphs with a single sentence.

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness
and
Compactness implies Inexpressibility.

If we write these implications in the contrapositive form, we get

Expressibility implies Compactness fails
and
Compactness fails implies Completeness fails.

Let's sketch the reasons why second-order logic allows us to accomplish the following:

1. axiomatize the class of connected graphs with a single sentence.

No Completeness Theorem for Second-Order Logic

We have seen that

Completeness implies Compactness
and
Compactness implies Inexpressibility.

If we write these implications in the contrapositive form, we get

Expressibility implies Compactness fails
and
Compactness fails implies Completeness fails.

Let's sketch the reasons why second-order logic allows us to accomplish the following:

1. axiomatize the class of connected graphs with a single sentence.
2. describe $\mathbb{R}$ up to isomorphism with a single sentence.

Second-order quantification

Second-order quantification

Tasks:

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.
2. Explain why the class of connected graphs is axiomatizable:

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.
2. Explain why the class of connected graphs is axiomatizable:

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.
2. Explain why the class of connected graphs is axiomatizable:
   for all $S \subseteq V$,

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.
2. Explain why the class of connected graphs is axiomatizable:
   for all $S \subseteq V$, *either* $S = \emptyset$,

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.
2. Explain why the class of connected graphs is axiomatizable:
   for all $S \subseteq V$, *either* $S = \emptyset$, $S = V$,

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.
2. Explain why the class of connected graphs is axiomatizable:
   for all $S \subseteq V$, *either* $S = \emptyset$, $S = V$, *or* $\exists x, y \in V$ *with* $x \in S$ *and* $y \notin S$.

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.

2. Explain why the class of connected graphs is axiomatizable:
   for all $S \subseteq V$, *either* $S = \emptyset$, $S = V$, *or* $\exists x, y \in V$ *with* $x \in S$ *and* $y \notin S$.
   (Try to write this as a second-order sentence.)

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.
2. Explain why the class of connected graphs is axiomatizable:
   for all $S \subseteq V$, *either* $S = \emptyset$, $S = V$, *or* $\exists x, y \in V$ *with* $x \in S$ *and* $y \notin S$.
   (Try to write this as a second-order sentence.)
3. Explain why $\mathbb{R}$ is axiomatizable up to isomorphism:

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.
2. Explain why the class of connected graphs is axiomatizable:
   for all $S \subseteq V$, *either* $S = \emptyset$, $S = V$, *or* $\exists x, y \in V$ *with* $x \in S$ *and* $y \notin S$.
   (Try to write this as a second-order sentence.)
3. Explain why $\mathbb{R}$ is axiomatizable up to isomorphism:

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.

2. Explain why the class of connected graphs is axiomatizable:
   for all $S \subseteq V$, *either* $S = \emptyset$, $S = V$, *or* $\exists x, y \in V$ *with* $x \in S$ *and* $y \notin S$.
   (Try to write this as a second-order sentence.)

3. Explain why $\mathbb{R}$ is axiomatizable up to isomorphism:
   $\mathbb{R}$ *is the unique complete, ordered field.*

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.

2. Explain why the class of connected graphs is axiomatizable:
   for all $S \subseteq V$, *either* $S = \emptyset$, $S = V$, *or* $\exists x, y \in V$ *with* $x \in S$ *and* $y \notin S$.
   (Try to write this as a second-order sentence.)

3. Explain why $\mathbb{R}$ is axiomatizable up to isomorphism:
   $\mathbb{R}$ *is the unique complete, ordered field.*
   (Try to write this as a second-order sentence.)

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.
2. Explain why the class of connected graphs is axiomatizable:
   for all $S \subseteq V$, *either* $S = \emptyset$, $S = V$, *or* $\exists x, y \in V$ *with* $x \in S$ *and* $y \notin S$.
   (Try to write this as a second-order sentence.)
3. Explain why $\mathbb{R}$ is axiomatizable up to isomorphism:
   $\mathbb{R}$ *is the unique complete, ordered field.*
   (Try to write this as a second-order sentence.)

Either of the second two items is sufficient to derive that Second-Order Logic cannot have a Completeness Theorem.

Second-order quantification

Tasks:

1. discuss which quantities are n-th order objects.
2. Explain why the class of connected graphs is axiomatizable:
   for all $S \subseteq V$, *either* $S = \emptyset$, $S = V$, *or* $\exists x, y \in V$ *with* $x \in S$ *and* $y \notin S$.
   (Try to write this as a second-order sentence.)
3. Explain why $\mathbb{R}$ is axiomatizable up to isomorphism:
   $\mathbb{R}$ *is the unique complete, ordered field.*
   (Try to write this as a second-order sentence.)

Either of the second two items is sufficient to derive that Second-Order Logic cannot have a Completeness Theorem. (Why?)

Infinitary logic

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$,

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely,

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

1. The first-order atomic formulas are formulas of L.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

1. The first-order atomic formulas are formulas of L.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

1. The first-order atomic formulas are formulas of L.
2. If α is a formula of L, then $(\neg\alpha)$ is a formula of L.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

1. The first-order atomic formulas are formulas of L.
2. If α is a formula of L, then $(\neg\alpha)$ is a formula of L.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

1. The first-order atomic formulas are formulas of L.
2. If α is a formula of L, then $(\neg\alpha)$ is a formula of L.
3. If (α_i) is a $(< \kappa)$-sequence of formulas of L involving fewer than λ free variables altogether, then $\bigwedge \alpha_i$ and $\bigvee \alpha_i$ are formulas of L.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

1. The first-order atomic formulas are formulas of L.
2. If α is a formula of L, then $(\neg\alpha)$ is a formula of L.
3. If (α_i) is a $(< \kappa)$-sequence of formulas of L involving fewer than λ free variables altogether, then $\bigwedge \alpha_i$ and $\bigvee \alpha_i$ are formulas of L.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

1. The first-order atomic formulas are formulas of L.
2. If α is a formula of L, then $(\neg\alpha)$ is a formula of L.
3. If (α_i) is a $(< \kappa)$-sequence of formulas of L involving fewer than λ free variables altogether, then $\bigwedge \alpha_i$ and $\bigvee \alpha_i$ are formulas of L.
4. If α is a formula of L with $(< \lambda)$-many free variables and $X = (x_{i_j})$ is a sequence of $(< \lambda)$-many variables, then $(\forall X)\alpha$ and $(\exists X)\alpha$ are formulas of L.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

1. The first-order atomic formulas are formulas of L.
2. If α is a formula of L, then $(\neg\alpha)$ is a formula of L.
3. If (α_i) is a $(< \kappa)$-sequence of formulas of L involving fewer than λ free variables altogether, then $\bigwedge \alpha_i$ and $\bigvee \alpha_i$ are formulas of L.
4. If α is a formula of L with $(< \lambda)$-many free variables and $X = (x_{i_j})$ is a sequence of $(< \lambda)$-many variables, then $(\forall X)\alpha$ and $(\exists X)\alpha$ are formulas of L.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

1. The first-order atomic formulas are formulas of L.
2. If α is a formula of L, then $(\neg\alpha)$ is a formula of L.
3. If (α_i) is a $(< \kappa)$-sequence of formulas of L involving fewer than λ free variables altogether, then $\bigwedge \alpha_i$ and $\bigvee \alpha_i$ are formulas of L.
4. If α is a formula of L with $(< \lambda)$-many free variables and $X = (x_{i_j})$ is a sequence of $(< \lambda)$-many variables, then $(\forall X)\alpha$ and $(\exists X)\alpha$ are formulas of L.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

1. The first-order atomic formulas are formulas of L.
2. If α is a formula of L, then $(\neg\alpha)$ is a formula of L.
3. If (α_i) is a $(< \kappa)$-sequence of formulas of L involving fewer than λ free variables altogether, then $\bigwedge \alpha_i$ and $\bigvee \alpha_i$ are formulas of L.
4. If α is a formula of L with $(< \lambda)$-many free variables and $X = (x_{i_j})$ is a sequence of $(< \lambda)$-many variables, then $(\forall X)\alpha$ and $(\exists X)\alpha$ are formulas of L.

First-order languages are $\mathcal{L}_{\omega,\omega}$ languages.

Infinitary logic

Given infinite cardinal numbers $\lambda \leq \kappa$, we can define a class of infinitary languages where we may form conjunctions and disjunctions of $(< \kappa)$-many formulas at a time and apply first-order quantification in blocks of size $(< \lambda)$. These are called $\mathcal{L}_{\kappa,\lambda}$-languages.

More precisely, If L is an $\mathcal{L}_{\kappa,\lambda}$-language, then its formulas are constructed recursively by the rules:

1. The first-order atomic formulas are formulas of L.
2. If α is a formula of L, then $(\neg\alpha)$ is a formula of L.
3. If (α_i) is a $(< \kappa)$-sequence of formulas of L involving fewer than λ free variables altogether, then $\bigwedge \alpha_i$ and $\bigvee \alpha_i$ are formulas of L.
4. If α is a formula of L with $(< \lambda)$-many free variables and $X = (x_{i_j})$ is a sequence of $(< \lambda)$-many variables, then $(\forall X)\alpha$ and $(\exists X)\alpha$ are formulas of L.

First-order languages are $\mathcal{L}_{\omega,\omega}$ languages.

$\mathcal{L}_{\omega_1,\omega}$ languages

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty$$

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0}$$

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1}$$

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2}$$

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\neg\sigma_\infty := (\neg\sigma_{\geq 0}) \vee (\neg\sigma_{\geq 1}) \vee (\neg\sigma_{\geq 2}) \vee \cdots$$

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\neg\sigma_\infty := (\neg\sigma_{\geq 0}) \vee (\neg\sigma_{\geq 1}) \vee (\neg\sigma_{\geq 2}) \vee \cdots$$

This sentence expresses finiteness.

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\neg\sigma_\infty := (\neg\sigma_{\geq 0}) \vee (\neg\sigma_{\geq 1}) \vee (\neg\sigma_{\geq 2}) \vee \cdots$$

This sentence expresses finiteness.
We have seen that finiteness is not expressible in any first-order language

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\neg\sigma_\infty := (\neg\sigma_{\geq 0}) \vee (\neg\sigma_{\geq 1}) \vee (\neg\sigma_{\geq 2}) \vee \cdots$$

This sentence expresses finiteness.
We have seen that finiteness is not expressible in any first-order language ($\mathcal{L}_{\omega,\omega}$),

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\neg\sigma_\infty := (\neg\sigma_{\geq 0}) \vee (\neg\sigma_{\geq 1}) \vee (\neg\sigma_{\geq 2}) \vee \cdots$$

This sentence expresses finiteness.
We have seen that finiteness is not expressible in any first-order language ($\mathcal{L}_{\omega,\omega}$), but here we see that it is expressible as an $\mathcal{L}_{\omega_1,\omega}$-sentence.

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\neg\sigma_\infty := (\neg\sigma_{\geq 0}) \vee (\neg\sigma_{\geq 1}) \vee (\neg\sigma_{\geq 2}) \vee \cdots$$

This sentence expresses finiteness.
We have seen that finiteness is not expressible in any first-order language ($\mathcal{L}_{\omega,\omega}$), but here we see that it is expressible as an $\mathcal{L}_{\omega_1,\omega}$-sentence. In this $\mathcal{L}_{\omega_1,\omega}$-langauge,

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\neg\sigma_\infty := (\neg\sigma_{\geq 0}) \vee (\neg\sigma_{\geq 1}) \vee (\neg\sigma_{\geq 2}) \vee \cdots$$

This sentence expresses finiteness.
We have seen that finiteness is not expressible in any first-order language ($\mathcal{L}_{\omega,\omega}$), but here we see that it is expressible as an $\mathcal{L}_{\omega_1,\omega}$-sentence. In this $\mathcal{L}_{\omega_1,\omega}$-langauge, for $\Gamma = \{\sigma_{\geq n} \mid n \in \mathbb{N}\}$,

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\neg\sigma_\infty := (\neg\sigma_{\geq 0}) \vee (\neg\sigma_{\geq 1}) \vee (\neg\sigma_{\geq 2}) \vee \cdots$$

This sentence expresses finiteness.
We have seen that finiteness is not expressible in any first-order language ($\mathcal{L}_{\omega,\omega}$), but here we see that it is expressible as an $\mathcal{L}_{\omega_1,\omega}$-sentence. In this $\mathcal{L}_{\omega_1,\omega}$-langauge, for $\Gamma = \{\sigma_{\geq n} \mid n \in \mathbb{N}\}$, we have $\Gamma \models \sigma_\infty$,

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\neg\sigma_\infty := (\neg\sigma_{\geq 0}) \vee (\neg\sigma_{\geq 1}) \vee (\neg\sigma_{\geq 2}) \vee \cdots$$

This sentence expresses finiteness.
We have seen that finiteness is not expressible in any first-order language ($\mathcal{L}_{\omega,\omega}$), but here we see that it is expressible as an $\mathcal{L}_{\omega_1,\omega}$-sentence. In this $\mathcal{L}_{\omega_1,\omega}$-langauge, for $\Gamma = \{\sigma_{\geq n} \mid n \in \mathbb{N}\}$, we have $\Gamma \models \sigma_\infty$, but $\Gamma_0 \not\models \sigma_\infty$ for any finite $\Gamma_0 \subseteq \Gamma$.

$\mathcal{L}_{\omega_1,\omega}$ languages

Example.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\sigma_\infty := \sigma_{\geq 0} \wedge \sigma_{\geq 1} \wedge \sigma_{\geq 2} \wedge \cdots$$

This sentence expresses infiniteness.
The following is an $\mathcal{L}_{\omega_1,\omega}$-sentence:

$$\neg\sigma_\infty := (\neg\sigma_{\geq 0}) \vee (\neg\sigma_{\geq 1}) \vee (\neg\sigma_{\geq 2}) \vee \cdots$$

This sentence expresses finiteness.
We have seen that finiteness is not expressible in any first-order language ($\mathcal{L}_{\omega,\omega}$), but here we see that it is expressible as an $\mathcal{L}_{\omega_1,\omega}$-sentence. In this $\mathcal{L}_{\omega_1,\omega}$-langauge, for $\Gamma = \{\sigma_{\geq n} \mid n \in \mathbb{N}\}$, we have $\Gamma \models \sigma_\infty$, but $\Gamma_0 \not\models \sigma_\infty$ for any finite $\Gamma_0 \subseteq \Gamma$. This shows that Compactness fails. It follows that the Completeness Theorem also fails for $\mathcal{L}_{\omega,\omega}$-languages if we insist that proofs be finite.

But what if proofs are allowed to be infinite?

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness,

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite.

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin)

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption.

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$.

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$. There is a proof calculus for $\mathcal{L}_{\omega_1,\omega}$-languages such that

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$. There is a proof calculus for $\mathcal{L}_{\omega_1,\omega}$-languages such that

$$\Gamma \models \alpha \qquad \Leftrightarrow \qquad \Gamma \vdash^* \alpha.$$

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$. There is a proof calculus for $\mathcal{L}_{\omega_1,\omega}$-languages such that

$$\Gamma \models \alpha \qquad \Leftrightarrow \qquad \Gamma \vdash^{*} \alpha.$$

The following are features of her proof calculus and of her theorem:

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$. There is a proof calculus for $\mathcal{L}_{\omega_1,\omega}$-languages such that

$$\Gamma \models \alpha \qquad \Leftrightarrow \qquad \Gamma \vdash^* \alpha.$$

The following are features of her proof calculus and of her theorem:

1. A proof is not finite sequences of statements, but is a sequence indexed by some countable ordinal $\delta < \omega_1$ or is a well-founded tree of countable height.

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$. There is a proof calculus for $\mathcal{L}_{\omega_1,\omega}$-languages such that

$$\Gamma \models \alpha \qquad \Leftrightarrow \qquad \Gamma \vdash^* \alpha.$$

The following are features of her proof calculus and of her theorem:

1. A proof is not finite sequences of statements, but is a sequence indexed by some countable ordinal $\delta < \omega_1$ or is a well-founded tree of countable height.

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$. There is a proof calculus for $\mathcal{L}_{\omega_1,\omega}$-languages such that

$$\Gamma \models \alpha \qquad \Leftrightarrow \qquad \Gamma \vdash^* \alpha.$$

The following are features of her proof calculus and of her theorem:

1. A proof is not finite sequences of statements, but is a sequence indexed by some countable ordinal $\delta < \omega_1$ or is a well-founded tree of countable height.
2. The calculus has (countably) infinitary inference rules.

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$. There is a proof calculus for $\mathcal{L}_{\omega_1,\omega}$-languages such that

$$\Gamma \models \alpha \qquad \Leftrightarrow \qquad \Gamma \vdash^* \alpha.$$

The following are features of her proof calculus and of her theorem:

1. A proof is not finite sequences of statements, but is a sequence indexed by some countable ordinal $\delta < \omega_1$ or is a well-founded tree of countable height.
2. The calculus has (countably) infinitary inference rules.

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$. There is a proof calculus for $\mathcal{L}_{\omega_1,\omega}$-languages such that

$$\Gamma \models \alpha \qquad \Leftrightarrow \qquad \Gamma \vdash^* \alpha.$$

The following are features of her proof calculus and of her theorem:

1. A proof is not finite sequences of statements, but is a sequence indexed by some countable ordinal $\delta < \omega_1$ or is a well-founded tree of countable height.
2. The calculus has (countably) infinitary inference rules.
3. Karp's Completeness Theorem is proved only for Γ countable.

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$. There is a proof calculus for $\mathcal{L}_{\omega_1,\omega}$-languages such that

$$\Gamma \models \alpha \qquad \Leftrightarrow \qquad \Gamma \vdash^* \alpha.$$

The following are features of her proof calculus and of her theorem:

1. A proof is not finite sequences of statements, but is a sequence indexed by some countable ordinal $\delta < \omega_1$ or is a well-founded tree of countable height.
2. The calculus has (countably) infinitary inference rules.
3. Karp's Completeness Theorem is proved only for Γ countable.

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$. There is a proof calculus for $\mathcal{L}_{\omega_1,\omega}$-languages such that

$$\Gamma \models \alpha \qquad \Leftrightarrow \qquad \Gamma \vdash^* \alpha.$$

The following are features of her proof calculus and of her theorem:

1. A proof is not finite sequences of statements, but is a sequence indexed by some countable ordinal $\delta < \omega_1$ or is a well-founded tree of countable height.
2. The calculus has (countably) infinitary inference rules.
3. Karp's Completeness Theorem is proved only for Γ countable.

This turns out to be a best possible for infinitary logics.

But what if proofs are allowed to be infinite?

Although the failure of Compactness usually signals the failure of Completeness, this conclusion was derived from the assumption that proofs must be finite. In 1959, Carol Karp (a Ph.D. student of Leon Henkin) set aside this assumption. Her theorem is:

Karp's Completeness Theorem for $\mathcal{L}_{\omega_1,\omega}$. There is a proof calculus for $\mathcal{L}_{\omega_1,\omega}$-languages such that

$$\Gamma \models \alpha \qquad \Leftrightarrow \qquad \Gamma \vdash^* \alpha.$$

The following are features of her proof calculus and of her theorem:

1. A proof is not finite sequences of statements, but is a sequence indexed by some countable ordinal $\delta < \omega_1$ or is a well-founded tree of countable height.
2. The calculus has (countably) infinitary inference rules.
3. Karp's Completeness Theorem is proved only for Γ countable.

This turns out to be a best possible for infinitary logics.
(No Completeness Theorem for $\mathcal{L}_{\kappa,\lambda}$ if $\kappa > \omega_1$ or $\lambda > \omega$.)

Second-order logic versus $\mathcal{L}_{\omega_1,\omega}$

Second-order logic versus $\mathcal{L}_{\omega_1,\omega}$

Neither subsumes the other.

Second-order logic versus $\mathcal{L}_{\omega_1,\omega}$

Neither subsumes the other. It can be shown that the completeness of the real numbers is not expressible in $\mathcal{L}_{\omega_1,\omega}$, but the class of connected graphs is axiomatizable in $\mathcal{L}_{\omega_1,\omega}$.

Second-order logic versus $\mathcal{L}_{\omega_1,\omega}$

Neither subsumes the other. It can be shown that the completeness of the real numbers is not expressible in $\mathcal{L}_{\omega_1,\omega}$, but the class of connected graphs is axiomatizable in $\mathcal{L}_{\omega_1,\omega}$.

Define a path of length n predicate:

Second-order logic versus $\mathcal{L}_{\omega_1,\omega}$

Neither subsumes the other. It can be shown that the completeness of the real numbers is not expressible in $\mathcal{L}_{\omega_1,\omega}$, but the class of connected graphs is axiomatizable in $\mathcal{L}_{\omega_1,\omega}$.

Define a path of length n predicate:

$$\text{Path}_n(x, y): \quad (\exists \mathbf{z})(E(x, z_1) \wedge \cdots \wedge E(z_{n-1}, y).).$$

Second-order logic versus $\mathcal{L}_{\omega_1,\omega}$

Neither subsumes the other. It can be shown that the completeness of the real numbers is not expressible in $\mathcal{L}_{\omega_1,\omega}$, but the class of connected graphs is axiomatizable in $\mathcal{L}_{\omega_1,\omega}$.

Define a path of length n predicate:

$$\text{Path}_n(x, y): \quad (\exists \mathbf{z})(E(x, z_1) \wedge \cdots \wedge E(z_{n-1}, y).).$$

Assert connectedness:

Second-order logic versus $\mathcal{L}_{\omega_1,\omega}$

Neither subsumes the other. It can be shown that the completeness of the real numbers is not expressible in $\mathcal{L}_{\omega_1,\omega}$, but the class of connected graphs is axiomatizable in $\mathcal{L}_{\omega_1,\omega}$.

Define a path of length n predicate:

$$\text{Path}_n(x, y): \quad (\exists \mathbf{z})(E(x, z_1) \wedge \cdots \wedge E(z_{n-1}, y).).$$

Assert connectedness:

$$\gamma: \quad (\forall x)(\forall y)\left((x \neq y) \rightarrow \bigvee_n \text{Path}_n(x, y)\right).$$

Second-order logic versus $\mathcal{L}_{\omega_1,\omega}$

Neither subsumes the other. It can be shown that the completeness of the real numbers is not expressible in $\mathcal{L}_{\omega_1,\omega}$, but the class of connected graphs is axiomatizable in $\mathcal{L}_{\omega_1,\omega}$.

Define a path of length n predicate:

$$\text{Path}_n(x, y): \quad (\exists \mathbf{z})(E(x, z_1) \wedge \cdots \wedge E(z_{n-1}, y).).$$

Assert connectedness:

$$\gamma: \quad (\forall x)(\forall y)\left((x \neq y) \rightarrow \bigvee_n \text{Path}_n(x, y)\right).$$