



SAPIENZA
UNIVERSITÀ DI ROMA

“SAPIENZA” UNIVERSITY OF ROME
FACULTY OF INFORMATION ENGINEERING,
INFORMATICS AND STATISTICS
DEPARTMENT OF COMPUTER SCIENCE

Mathematical Logic for Computer Science

Lecture notes integrated with the book TODO

Author
Alessio Bandiera

April 18, 2025

Contents

Information and Contacts	1
1 Exercises	2
1.1 Homework 1	2
1.1.1 Theories	4

Information and Contacts

Personal notes and summaries collected as part of the *Mathematical Logic for Computer Science* course offered by the degree in Computer Science of the University of Rome "La Sapienza".

Further information and notes can be found at the following link:

<https://github.com/aflaag-notes>. Anyone can feel free to report inaccuracies, improvements or requests through the Issue system provided by GitHub itself or by contacting the author privately:

- Email: alessio.bandiera02@gmail.com
- LinkedIn: [Alessio Bandiera](#)

The notes are constantly being updated, so please check if the changes have already been made in the most recent version.

Suggested prerequisites:

TODO

Licence:

These documents are distributed under the [GNU Free Documentation License](#), a form of copyleft intended for use on a manual, textbook or other documents. Material licensed under the current version of the license can be used for any purpose, as long as the use meets certain conditions:

- All previous authors of the work must be **attributed**.
- All changes to the work must be **logged**.
- All derivative works must be **licensed under the same license**.
- The full text of the license, unmodified invariant sections as defined by the author if any, and any other added warranty disclaimers (such as a general disclaimer alerting readers that the document may not be accurate for example) and copyright notices from previous versions must be maintained.
- Technical measures such as DRM may not be used to control or obstruct distribution or editing of the document.

1

Exercises

1.1 Homework 1

placeholder

missing
intro-
duction

Definition 1.1: Languages and propositions

A **propositional language** is a — possibly infinite — set $\mathcal{L} = \{p_1, \dots, p_n\}$, where each p_i is called **propositional variable**. Given a propositional language \mathcal{L} , the set of **propositions** over \mathcal{L} , denoted with $\text{PROP}_{\mathcal{L}}$ is inductively defined as follows:

- each propositional variable in \mathcal{L} is a proposition, i.e. $\mathcal{L} \subseteq \text{PROP}_{\mathcal{L}}$
- if $A \in \text{PROP}_{\mathcal{L}}$, then $\neg A \in \text{PROP}_{\mathcal{L}}$
- if $A, B \in \text{PROP}_{\mathcal{L}}$, then $(A \wedge B), (A \vee B), (A \rightarrow B), (A \leftrightarrow B) \in \text{PROP}_{\mathcal{L}}$

In other words, the set of propositions over a language is the set of *formulas* that can be constructed from the initial variables of \mathcal{L} , by using the Boolean connectives. Note that a propositional language may have an infinite number of propositional variables, even *uncountably infinite*. For instance, the following is a valid propositional language

$$\mathcal{L} := \{p_r \mid r \in \mathbb{R}\}$$

However, since propositions are inductively constructed starting from variables, each proposition of any language \mathcal{L} will still be defined over a *finite* number of propositional variables of \mathcal{L} .

Now that we provided a formal definition for languages and propositions, we are ready to discuss **assignments**.

Definition 1.2: Assignment

Given a propositional language \mathcal{L} , an **assignment** α is a function $\alpha : \mathcal{L} \rightarrow \{0, 1\}$ that assigns either 0 or 1 to all of \mathcal{L} 's propositional variable.

This definition can be inductively extended to **propositions** themselves: if A and B are two propositions, let $\hat{\alpha} : \text{PROP}_{\mathcal{L}} \rightarrow \{0, 1\}$ be a function such that

$$A \in \mathcal{L} \implies \hat{\alpha}(A) = \alpha(A)$$

$$\hat{\alpha}(\neg A) = \begin{cases} 1 & \hat{\alpha}(A) = 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{\alpha}(A \wedge B) = \begin{cases} 1 & \hat{\alpha}(A) = \hat{\alpha}(B) = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{\alpha}(A \vee B) = \begin{cases} 1 & \hat{\alpha}(A) = 1 \vee \hat{\alpha}(B) = 1 \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{\alpha}(A \rightarrow B) = \begin{cases} 0 & \hat{\alpha}(A) = 1 \wedge \hat{\alpha}(B) = 0 \\ 0 & \text{otherwise} \end{cases}$$

$$\hat{\alpha}(A \leftrightarrow B) = \begin{cases} 1 & \hat{\alpha}(A) = \hat{\alpha}(B) \\ 0 & \text{otherwise} \end{cases}$$

Since it can be proven that the extension $\hat{\alpha}$ of α is unique, we will refer to $\hat{\alpha}$ as α directly. Two propositions A and B are said to be **equivalent** — written as $A \equiv B$ — if and only if $\forall \alpha \quad \alpha(A) = \alpha(B)$.

Definition 1.3: Satisfiability

A proposition A is said to be **satisfiable** if there exists an assignment α of its propositional variables such that $\alpha(A) = 1$. If there is no assignment that satisfies A , A is said to be **unsatisfiable**, and if A is satisfied for any assignment α , then A is said to be a **tautology**.

We will denote with SAT, UNSAT and TAUT respectively the sets of all satisfiable propositions, all unsatisfiable propositions and all tautologies.

Using symbols, we have that

- $A \in \text{SAT} \iff \exists \alpha \quad \alpha(A) = 1$
- $A \in \text{UNSAT} \iff \nexists \alpha \quad \alpha(A) = 1 \iff \forall \alpha \quad \alpha(A) = 0$

- $A \in \text{TAUT} \iff \forall \alpha \quad \alpha(A) = 1 \iff \nexists \alpha \quad \alpha(A) = 0$

The concept of satisfiability is strictly related to the concept of **logical consequence**, which is defined as follows.

Definition 1.4: Logical consequence

Given the propositions A_1, \dots, A_n, A , we say that A is a **logical consequence** of A_1, \dots, A_n if whenever A_1, \dots, A_n are true, A is also true. We will indicate this concept as follows:

$$A_1, \dots, A_n \models A$$

From its definition, the concept of logical consequence can be alternatively be expressed in terms of *unsatisfiability* and *tautology*.

Theorem 1.1

Given the formulas A_1, \dots, A_n, A , the following statements are equivalent:

- $A_1, \dots, A_n \models A$
- $(A_1 \wedge \dots \wedge A_n \rightarrow A) \in \text{TAUT}$
- $(A_1 \wedge \dots \wedge A_n \wedge A) \in \text{UNSAT}$

1.1.1 Theories

Definition 1.5: Theory

Given a language \mathcal{L} , a **theory** T over \mathcal{L} is a — finite or infinite — set of propositions defined on \mathcal{L} , i.e. $T \subseteq \text{PROP}_{\mathcal{L}}$.

As a natural extension of the *satisfiability* property previously discussed, a theory T will be said to be **satisfiable** — written as $T \in \text{SAT}$ if and only if

$$\exists \alpha \quad \alpha(T) = 1$$

which is equivalent of saying that

$$\exists \alpha \quad \forall F \in T \quad \alpha(F) = 1$$

note that the assignment α must be the same for all the propositions F of T .

Additionally, for infinite theories we can define another property.

Definition 1.6: Finite satisfiability

An infinite theory T is said to be **finitely satisfiable** if and only if

$$\forall T' \subset T \text{ finite} \quad T' \in \text{SAT}$$

We will denote with **FINSAT** the set of all finitely satisfiable theories.

However, the following theorem will prove that *satisfiability* and *finite satisfiability* are actually **equivalent**.

Theorem 1.2: Compactness theorem (1st version)

Given an infinite theory T , it holds that

$$T \in \text{SAT} \iff T \in \text{FINSAT}$$

Proof. In this proof we will assume that the propositions of the infinite theory T are *countably infinite*, however in its general form this theorem can be proved even without this assumption.

Since the direct implication of this statement is trivially true by definition, we just need to prove the converse implication.

Claim: Given a theory $T \in \text{FINSAT}$, and a proposition A , it must hold that $T \cup \{A\} \in \text{FINSAT}$ or $T \cup \{\neg A\} \in \text{FINSAT}$.

Proof of the Claim. By way of contradiction, assume that $T \cup \{A\}, T \cup \{\neg A\} \notin \text{FINSAT}$.

By definition of finite satisfiability, if $T \cup \{A\} \notin \text{FINSAT}$, then there must exist a *finite* sub-theory $T_0 \subset T \cup \{A\}$ such that $T_0 \in \text{UNSAT}$. Note that $T \in \text{FINSAT}$, therefore if $T \cup \{A\} \notin \text{FINSAT}$ then it must be that $A \in T_0$. Let \widehat{T}_0 be the theory such that $T_0 := \widehat{T}_0 \cup \{A\}$; then

$$T_0 := \widehat{T}_0 \cup \{A\} \in \text{UNSAT} \iff \forall \alpha \quad \alpha(T_0) = 0$$

which implies that

$$\forall \alpha \quad \alpha(\widehat{T}_0) = 1 \implies \alpha(A) = 0$$

Analogously, we can apply the same reasoning for $T \cup \{\neg A\}$, and we get that there must exist a *finite* sub-theory $T_1 \subset T \cup \{\neg A\}$ such that $T_1 := \widehat{T}_1 \cup \{\neg A\} \in \text{UNSAT}$, which implies that

$$\forall \alpha \quad \alpha(\widehat{T}_1) = 1 \implies \alpha(\neg A) = 0$$

Lastly, since $\widehat{T}_0 \cup \widehat{T}_1 \subset T \in \text{FINSAT}$, by finite satisfiability of T there must exist an assignment α such that $\alpha(\widehat{T}_0 \cup \widehat{T}_1) = 1$, and therefore $\alpha(\widehat{T}_0) = \alpha(\widehat{T}_1) = 1$. However, for the previous observations this implies that $\alpha(A) = \alpha(\neg A) = 0 \nmid$. \square

Since we are assuming that the propositions of T are *countably infinite*, and the number of variables in any proposition is finite by definition, we can fix an enumeration p_1, p_2, p_3, \dots on the — possibly infinite — propositional variables of T . Given this enumeration, define the following *chain* of sub-theories:

- $T_0 := T$
- $T_{i+1} := \begin{cases} T_i \cup \{p_i\} & T_i \cup \{p_i\} \in \text{FINSAT} \\ T_i \cup \{\neg p_i\} & T_i \cup \{\neg p_i\} \in \text{FINSAT} \end{cases}$

and note that, by definition, clearly

$$T =: T_0 \subseteq T_1 \subseteq T_2 \subseteq \dots$$

Moreover, let

$$T^* := \bigcup_{i \in \mathbb{N}} T_i$$

and note that since $\forall i \quad T_i \in \text{FINSAT}$ by definition, then it must be that $T^* \in \text{FINSAT}$ as well, as T^* is a chain defined *only* by inclusions of FINSAT theories.

Now, consider the following assignment:

$$\alpha^* : \{p_1, p_2, \dots\} \rightarrow \{0, 1\} : p_i \mapsto \begin{cases} 1 & p_i \in T^* \\ 0 & \neg p_i \in T^* \end{cases}$$

Note that this assignment is well defined, because by construction of T^* only one between $p_i \in T^*$ and $\neg p_i \in T^*$ can hold.

Claim: $\alpha^*(T) = 1$.

Proof of the Claim. Let $A \in T$, and let p_{i_1}, \dots, p_{i_k} be the propositional variables that appear in A . Then, for each $j \in [k]$ let

$$p_{i_j}^* := \begin{cases} p_{i_j} & p_{i_j} \in T^* \\ \neg p_{i_j} & \neg p_{i_j} \in T^* \end{cases}$$

and consider the set $\{A, p_{i_1}^*, \dots, p_{i_k}^*\}$. Clearly, this is a finite subset of T^* — by definition of the various T_0, T_1, T_2, \dots that define T^* — hence $T^* \in \text{FINSAT}$ implies that there must exist an assignment β_A that satisfies this set, i.e.

$$\beta_A(A) = \beta_A(p_{i_1}^*) = \dots = \beta_A(p_{i_k}^*) = 1$$

Note that, for each $j \in [k]$, it holds that

$$p_{i_j} \in T^* \implies p_{i_j}^* = p_{i_j} \wedge \alpha^*(p_{i_j}) = 1$$

and $1 = \beta_A(p_{i_j}^*) = \beta_A(p_{i_j})$; analogously, it holds that

$$\neg p_{i_j} \in T^* \implies p_{i_j}^* = \neg p_{i_j} \wedge \alpha^*(p_{i_j}) = 0$$

and $1 = \beta_A(p_{i_j}^*) = \beta_A(\neg p_{i_j}) = \neg \beta_A(p_{i_j}) \implies \beta_A(p_{i_j}) = 0$. This proves that $\alpha^* \equiv \beta_A$ for all of A 's variables, therefore it must also be true that $\alpha^*(A) = \beta_A(A)$.

Hence, this shows that for any proposition $A \in T$ there is an assignment β_A , which satisfies A , that sets A 's variables as α^* does. Therefore, since this will be true for any of the T 's propositions, α^* satisfies T . \square

This claim proves that there exists an assignment α^* that satisfies T , hence $T \in \text{SAT}$, concluding the proof. \square

The statement of this theorem is equivalent to the following one.

Theorem 1.3: Compactness theorem (2nd version)

Given an infinite theory T , and a proposition A , it holds that

$$T \models A \iff \exists T' \subset T \text{ finite } T' \models A$$

Proof.

Direct implication. We will prove the equivalence of the two formulations by assuming that the first version implies the second version of the theorem, and vice versa. First, notice that if there exist a $T' \subset T$ finite sub-theory of T such that $T' \models A$ for some proposition A , then it is trivially true that $T \models A$ by definition of logical consequence. Hence, we just need to prove the direct implication of the second version of the theorem, assuming that the first version is true.

By way of contradiction, assume that $T \models A$ and for each $T' \subset T$ finite sub-theory of T , $T' \not\models A$. By definition of logical consequence, this implies that

$$\forall T' \subset T \text{ finite } \exists \alpha \quad \alpha(T') = 1 \wedge \alpha(A) = 0$$

or, equivalently, $\alpha(\neg A) = 1$. This implies that

$$\forall T' \subset T \text{ finite } \exists \alpha \quad \alpha(T' \cup \{\neg A\}) = 1$$

which means that any finite subset of $T \cup \{\neg A\}$ is SAT (recall that $T' \subset T$). Hence, by definition $T \cup \{\neg A\} \in \text{FINSAT}$, and since we are assuming the first version of the theorem, we have that

$$T \cup \{\neg A\} \in \text{FINSAT} \iff T \cup \{\neg A\} \in \text{SAT}$$

which implies that $T \not\models A$ by definition of logical consequence.

Converse implication. Consider a theory T and a proposition A ; by the second version of the theorem we know that $T \models A$ if and only if there is a finite sub-theory $T' \subset T$ such that $T' \models A$. Now, suppose that $T \in \text{UNSAT}$; therefore, by definition we have that $T \models 0$, which implies that

$$T \models 0 \iff \exists T' \subset T \text{ finite } T' \models 0$$

but again, $T' \models 0$ means that $T' \in \text{UNSAT}$. Hence, we proved that

$$T \in \text{UNSAT} \iff \exists T' \subset T \text{ finite } T' \in \text{UNSAT} \iff T \notin \text{FINSAT}$$

which is the contrapositive of the first version of the theorem.

□

The compactness theorem can be proven to be equivalent to a special case of [König's lemma](#) [Kön27], which states the following.

Lemma 1.1: König's lemma (special case)

Every infinite tree contains either a vertex of infinite degree, or an infinite path.

Exercise 1.4 Let $\mathcal{L} = \{E(x, y)\}$ be the language of graphs.

1. For each fixed $n \in \mathbb{N}$, write a sentence C_n such that for any graph \mathcal{G} , $\mathcal{G} \models C_n$ if and only if \mathcal{G} contains a cycle of length n .
2. Prove using Compactness that the property of being *a cycle* is not expressible by a theory in \mathcal{L} over the class of graphs.

Solution. Let $\mathcal{L} = \{E(x, y)\}$ be the language of graphs.

1. The property “ \mathcal{G} contains a cycle of length n ” can be written as follows

$$C_n := \exists x_1 \dots \exists x_n \left(\bigwedge_{\substack{1 \leq i, j \leq n \\ i \neq j}} \neg(x_i = x_j) \right) \wedge \left(\bigwedge_{1 \leq i \leq n-1} E(x_i, x_{i+1}) \wedge E(x_n, x_1) \right)$$

In fact, the first conjunction implies that x_1, \dots, x_n are *distinct*, and the second conjunction describes the existence of the n -long *cycle* itself.

2. Consider the property $P_n :=$ “ \mathcal{G} is *a cycle* of length n ”. This property can be expressed by *extending* C_n as follows:

$$\begin{aligned} V_n &:= \forall y \bigvee_{1 \leq j \leq n} (y = x_j) \\ E_n &:= \bigwedge_{1 \leq i \leq n-1} \bigwedge_{\substack{1 \leq j \leq n: \\ j \neq i+1}} \neg E(x_i, x_j) \wedge \bigwedge_{2 \leq j \leq n} \neg E(x_n, x_j) \\ C'_n &:= \exists x_1 \dots \exists x_n \quad C_n \wedge V_n \wedge E_n \end{aligned}$$

where we have that

- V_n ensures that \mathcal{G} has *exactly* n vertices
- E_n ensures that the only edges present in \mathcal{G} are the ones that describe the cycle graph of n vertices
- C'_n describes our property P_n

Now, consider the property $P :=$ “ \mathcal{G} is *a cycle*”, and in particular $\neg P :=$ “ \mathcal{G} is not *a cycle*”. We observe that we can build the following infinite theory

$$T^{\neg P} := \{\neg C'_n \mid n \in \mathbb{N}_{\geq 3}\}$$

for which it is easy to see that

$$\mathcal{G} \models \neg P \iff \neg P(\mathcal{G}) \text{ holds}$$

meaning that $\neg P$ is expressible through $T^{\neg P}$.

Claim: $T^{\neg P} \in \text{FINSAT}$.

Proof of the Claim. Fix $T_0 \subseteq_{fin} T^{\neg P}$. We observe that $T_0 := \{\neg C'_{i_1}, \dots, \neg C'_{i_k}\}$

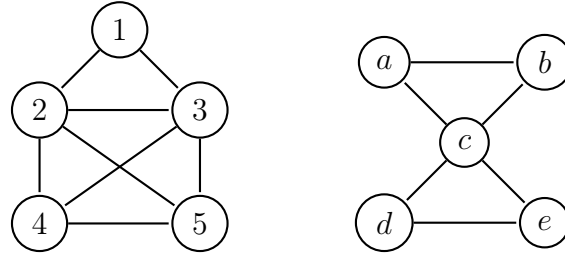
for some $i_1, \dots, i_k \in \mathbb{N}$. Now, if we consider $i^* := \max_{j \in [k]} i_j$, then the cycle graph that has $i^* + 1$ vertices is clearly a structure that satisfies T_0 . \square

Claim: P is not expressible by a theory in \mathcal{L} over the class of graphs.

Proof of the Claim. By way of contradiction, suppose that P is expressible, i.e. there is a theory T^P for which P can be expressed. Then, consider the theory $T := T^P \cup T^{\neg P}$. By the previous claim, we have that $T \in \text{FINSAT}$, and by Compactness this is true if and only if $T \in \text{SAT}$. However, this is a contradiction, because a graph cannot be and not be a cycle at the same time. \square

Finally, this last claim concludes the proof. \square

Exercise 2.1 Consider the following two structures \mathcal{G}_1 and \mathcal{G}_2 for the languages of graphs:



Write at least two sentences distinguishing the two structures. Discuss the EF-game played on these structures: for what k can the Duplicator win the k -rounds game? For what k can the Spoiler win?

Solution. Some properties that can distinguish these two structures are the following:

1. “ \mathcal{G} contains a vertex of degree 3”, which is represented by the following sentence of rank 5

$$\exists x_1 \exists x_2 \exists x_3 \exists x_4 \left(\bigwedge_{\substack{1 \leq i, j \leq 4 \\ i \neq j}} \neg(x_i = x_j) \right) \wedge \left(\bigwedge_{2 \leq i \leq 4} E(x_1, x_i) \right) \wedge \left(\forall y \quad \neg E(x_1, y) \vee \bigvee_{2 \leq j \leq 4} (y = x_j) \right)$$

2. “ \mathcal{G} contains edges as \mathcal{G}_1 ”, which is represented by the following sentence of rank 5

$$\begin{aligned} \exists x_1 \exists x_2 \exists x_3 \exists x_4 \exists x_5 \quad & E(x_1, x_2) \wedge E(x_1, x_3) \wedge \\ & E(x_2, x_3) \wedge E(x_2, x_4) \wedge E(x_2, x_5) \wedge \\ & E(x_3, x_4) \wedge E(x_4, x_5) \wedge \\ & E(x_4, x_5) \end{aligned}$$

we observe that the edges of \mathcal{G}_2 are not sufficient to distinguish the two sentences, because \mathcal{G}_2 is a subgraph of \mathcal{G}_1

3. “ \mathcal{G} contains a cycle of length 5”, which is represented by C_5 of the previous exercise, and has rank 5
4. “ \mathcal{G} contains a cycle of length 4”, which is represented by C_4 of the previous exercise, and has rank 4
5. “ \mathcal{G} contains K_4 as subgraph”, which is represented by the following sentence having rank 4

$$\exists x_1 \exists x_2 \exists x_3 \exists x_4 \left(\bigwedge_{\substack{1 \leq i, j \leq 4 \\ i \neq j}} \neg(x_i = x_j) \right) \wedge \left(\bigwedge_{\substack{1 \leq i, j \leq 4 \\ i \neq j}} E(x_i, x_j) \right)$$

These sentences *may seem* to suggest that the two structures are 3-equivalent, meaning that there is no sentence of rank 3 that can distinguish \mathcal{G}_1 from \mathcal{G}_2 . For now, let's focus on proving that they are *at least* 2-equivalent.

Claim: The Duplicator wins $G_2(\mathcal{G}_1, \mathcal{G}_2)$.

Proof of the Claim. Let s_i and d_i be the i -th nodes chosen by the Spoiler and the Duplicator, respectively. Then, we can define the following strategy for the Duplicator:

- if $s_1 \in \{1, 4, 5\}$, then the Duplicator chooses $d_1 \in \{a, b, d, e\}$, otherwise if $s_1 \in \{2, 3\}$ then $d_1 = c$
- similarly, if $s_1 \in \{a, b, d, e\}$, then the Duplicator chooses $d_1 \in \{1, 4, 5\}$, otherwise if $s_1 = c$ then $d_1 \in \{2, 3\}$

Then, no matter the choice of s_2 , the Duplicator can always answer with a node d_2 that preserves the partial isomorphism, in fact:

- if $s_2 \sim s_1$, it is guaranteed that there is a vertex d_2 in the other structure such that $d_2 \sim d_1$ because $\delta(\mathcal{G}_1) = \delta(\mathcal{G}_2) = 2$ — and the same argument applies if $s_2 \sim d_1$ for finding a vertex $d_2 \sim s_1$
- if $s_2 \not\sim s_1$, the strategy that we provided for the Duplicator guarantees that there exists at least one vertex d_2 in the other structure such that $d_2 \not\sim d_1$ — and the same argument applies if $s_2 \not\sim d_1$ for finding a vertex $d_2 \not\sim s_1$

Thus, the Duplicator has a strategy to always win at least 2 rounds, therefore the Duplicator wins $G_2(\mathcal{G}_1, \mathcal{G}_2)$ by Ehrenfeucht's theorem. \square

Now that we proved that $\mathcal{G}_1 \equiv_2 \mathcal{G}_2$, is it true that they are also 3-equivalent? Unfortunately, the following claim proves that this is indeed false.

Claim: The Spoiler wins $G_3(\mathcal{G}_1, \mathcal{G}_2)$.

Proof of the Claim. The following is a strategy that guarantees the Spoiler to win in 3 rounds:

- let $s_1 \in \{4, 5\}$

- by the previous claim, we know that the strategy for the Duplicator to win at least 2 rounds is to choose $d_1 \in \{a, b, d, e\}$, thus we may assume that $d_1 \neq c$
- now, let $s_2 = 1$
- to preserve the partial isomorphism, we observe that
 - if $d_1 \in \{a, b\}$, then $d_2 \in \{d, e\}$
 - if $d_1 \in \{d, e\}$, then $d_2 \in \{a, b\}$
- now, it suffices for the Spoiler to choose s_3 in \mathcal{G}_2 such that $s_3 \sim d_2$ and $s_3 \neq c$: by construction of \mathcal{G}_2 , we see that $s_3 \approx d_1$, but all the vertices in $\{2, 3, 5\}$ are adjacent to s_1 , which would violate the partial isomorphism

□

In fact, we can actually find a property that distinguishes \mathcal{G}_1 from \mathcal{G}_2 which can be written through a sentence of rank 3: “there are two vertices x_1 and x_2 of \mathcal{G} such that for each third vertex x_3 there is a K_3 as subgraph of \mathcal{G} such that $V(K_3) = \{x_1, x_2, x_3\}$ ”

$$\exists x_1 \exists x_2 \forall x_3 \left(\bigwedge_{\substack{1 \leq i, j \leq 3 \\ i \neq j}} \neg(x_i = x_j) \right) \wedge E(x_1, x_2) \wedge E(x_2, x_3) \wedge E(x_3, x_1)$$

Let x_1, x_2 and x_3 be the three chosen vertices — and we may assume that $x_1 \sim x_2$ otherwise the sentence is trivially unsatisfied. Then, we observe that

- in \mathcal{G}_1 if $\{x_1, x_2\} = \{2, 3\}$, then for any other vertex $x_3 \in \{1, 4, 5\}$ we can always find a K_3 having x_1, x_2 and x_3 as its vertices
- in \mathcal{G}_2 we have two cases
 - if $\{x_1, x_2\} \subseteq \{a, b, c\}$, the property is unsatisfied for $x_3 \in \{d, e\}$
 - if $\{x_1, x_2\} \subseteq \{c, d, e\}$, the property is unsatisfied for $x_3 \in \{a, b\}$

In conclusion, we have that $\mathcal{G}_1 \equiv_2 \mathcal{G}_2$, and that $\mathcal{G}_1 \not\equiv_3 \mathcal{G}_2$.

□