Basic stuff about models

- Models can be seen as rules for interpreting determining the truth-value of sentences in quantificational logic.

- In that sense, they are extensions of the rules we had for determining the truth-values of sentences with operators in propositional logic. These rules for the connectives are still in play in quantificational logic.

- The reason we need models is that we want to do similar things to what we did in propositional logic: determine the validity of an argument, whether a sentence is true or false depending on the truth-value of its propositional variables, determine whether two sentences are consistent, etc.

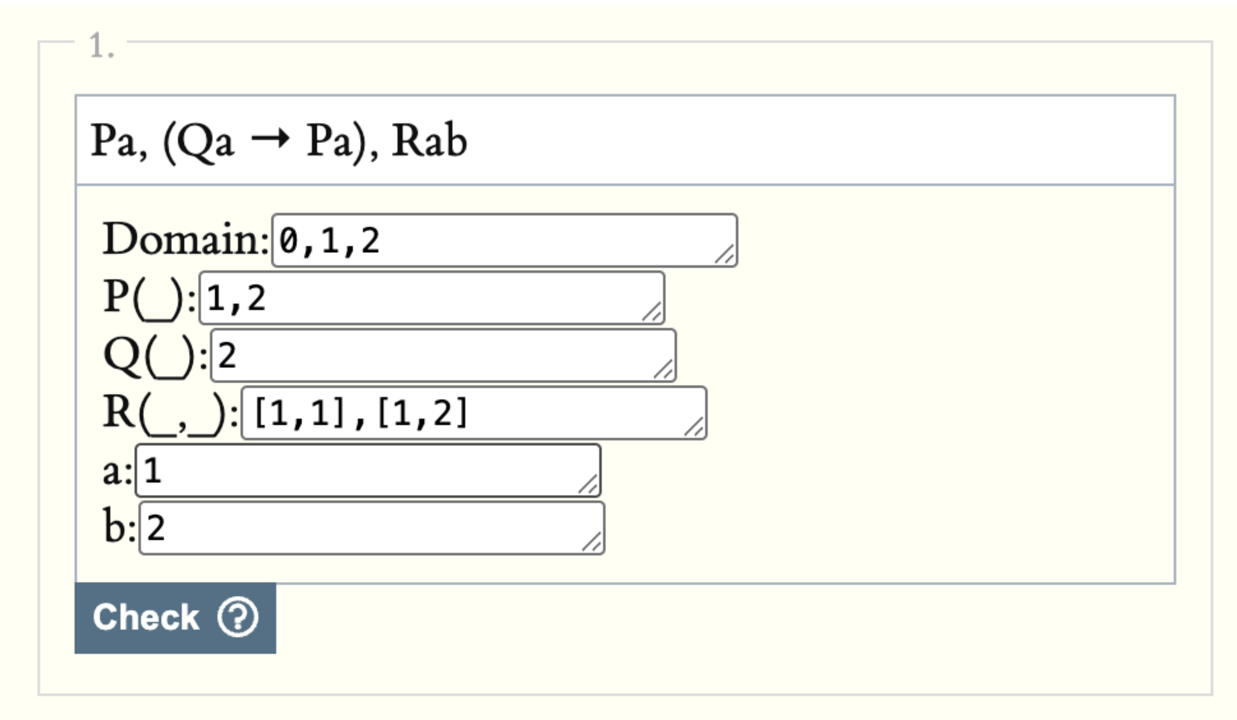
- In quantificational logic, we will sometimes say that a sentence is true (or false) *with respect to a model* or that a model *verifies* (or makes false) a sentence, since it is as if the model told us how to understand the domain of objects being talked about, what objects the constants refer to and of which objects the predicates are true.

- For example from English, 'Nurit' refers to me, and the predicate 'is a female Peruvian philosophy graduate student at USC' is true of me (but 'is a Peruvian philosophy graduate student at USC' is true of me and Rodrigo).

- More technically, models consist of a *domain*, a *denotation* for each constant of the language and an *extension* for each predicate of the language.

How to input models into Carnap

Here's an example of how to input a model into Carnap:



- In Carnap (but not in logic in general!), domains will always be sets of numbers. To put in a set, you will write its members; that is '0,1,2' in the image above. These are the objects that constants *can* refer to.

- To specify what a constant refers to (i.e., its denotation), just type a number next to the constant. In the image above, 'a' refers to 1 and 'b' refers to 2.

- One-place predicates are true of objects. To put in the set of objects a one-place predicate is true of, write its members. In the image above, 'P(_)' is true of 1 and 2, and 'Q(_)' is only true 1. (Note that no one-place predicate is true of 0.)

- Two-place predicates are true of ordered *pairs* of objects. To put in the set of pairs of objects a two-place predicate is true of, write its members. In the image above, 'R(_._)' is true of [1,1] and [1,2]. Note that it is not true of [2,1]. If we think of 'R' as 'loves', we can say, loosely speaking, that according to this model, 1 loves 2 but 2 doesn't love 1. This is something very important to keep in mind when giving the extension of two-place predicates.

- Though there aren't predicates with more argument places in the example above, similar reasoning goes for n-place predicates: the extension will be a set of ordered n-tuples of objects, and you will write down the ordered n-tuples.

What does the model in the example above show about $Pa, Qa \to Pa, Rab$?

- It shows that Pa is true with respect to this model, since 'a' refers to 1 and 'P(_)' is true of 1 (and also of 2, but that doesn't matter for evaluating the truth-value of Pa).

- It shows that $Qa \to Pa$ is true with respect to this model. The reasoning here comes in two steps. First, 'a' refers to 1 and 'Q(_)' is true only of 1, so Qa is false. But a sentence of the form $\phi \to \psi$ is true when ϕ is false. Here, Qa is our false ϕ.

- It shows that Rab is true with respect to this model, since 'a' refers to 1, 'b' refers to 2, and 'R(_,_)' is true of [1,2] (and also of [1,1], but that that doesn't matter for evaluating the truth-value of Rab). Note that Rba is false with respect to this model (Comprehension question: why?).

Using models to show stuff: Creating a game-plan

What is the first thing I do when I'm asked to show that an argument is invalid?

- The first thing to do is to think about what I want to show: that the world could be such that the premises are true and the conclusion is false.
- This will show that the argument is not valid *in virtue of its logical form*.
- At the beginning of the semester we did this by instantiating the logical form with an argument in English with true premises and false conclusion.
- With quantificational logic, we've *cut the middleman* of English; instead, it is as if we connect the logical form directly with a way the world could be.
- So if the premises are true and the conclusion is false with respect to a model, then that model shows that the world could be such that the premises are true and the conclusion is false.
- **Note:** in order to show that an argument is invalid in propositional logic, we also showed that the world could be such that the argument's premises were true and the conclusion false. The difference is that in propositional logic we did so *via* identifying a row of the truth-table in which the premises were true and the conclusion was false.

What is the first thing I do when I'm asked to show that a set of sentences is consistent?

- The first thing to do is to think about what I want show: that the world could be such that all the sentences are true.

- This will show that the set of sentences is consistent *in virtue of their logical form*.
- We didn't do this at the beginning of the semester, but we could have by instantiating the logical form of each sentence with true sentences of English.
- With quantificational logic we've *cut the middleman* of English; instead, it is as if we connect the logical form directly with a way the world could be.
- So if all the sentences of the set are true with respect to a model, then that model shows that the world could be such that all the sentences are true.
- **Note:** in order to show that a set of sentences is consistent in propositional logic, we also showed that the world could be such that all the sentences are true. The difference is that in propositional logic we did so *via* identifying a row of the truth-table in which all the sentences are true.

Using models to show stuff: solving exercises

How do I go about showing $Pa, Qa \to Pa \vDash Qa$ is invalid?

- Recall: our goal is to offer a model with respect to which the premises of the argument are true and its conclusion is false. So we have to offer a model with respect to which $Fa \to Ga$ and Ga are true, and Fa is false.

- First, I'll fill out a model with respect to which Pa is true:

  - **Domain:** 0
  - a: 0
  - P(_): 0

- But I cannot stop there, since my model must be such that also $Qa \to Pa$ is true with respect to it. I can achieve this by making Qa false (since a sentence of the form $\phi \to \psi$ is true when ϕ is false), and I achieve that by making it so that 'Q(_)' is not true of anything:

  - **Domain:** 0
  - a: 0
  - P(_): 0
  - Q(_):

- Now, I must check that this also makes the conclusion false. But Qa is the conclusion! (Comprehension check: what if I had made Qa true?)

- Hence, my model is such that the premises are true and the conclusion is false with respect to it, and, so, I have shown that the argument is invalid.

How do I go about showing $\forall x(Px \rightarrow Qx)$ and $\exists x(Qx \wedge \neg Px)$ are consistent?

- Recall: our goal is to offer a model with respect to which both of these sentences are true. So we have to offer a model with respect to which both $\forall x(Px \rightarrow Qx)$ and $\exists x(Qx \wedge \neg Px)$ are true.

- How do I know whether some sentences with $\forall x$ and $\exists x$ is true with respect to a model? Here is the intuitive way to think about it:

  - A sentence like $\forall x Px$ is true with respect to a model if all the objects in the domain of the model are such that P is true of them. For example, $\forall x Px$ is true with respect to the following model:

    Domain: 0

    P(_): 0

  - But also with respect to the following model:

    Domain: 0,1

    P(_): 0,1

  - A sentence like $\exists x Px$ is true with respect to a model if at least one object in the domain of the model is such that P is true of them. For example, $\exists x Px$ is true with respect to the following model:

Domain: 0,1

P(_): 0

- But also with respect to the following model:

Domain: 0

P(_): 0

- And with respect to the following model:

Domain: 0,1

P(_): 0,1

- Now we are in a position to offer a model with respect to which both $\forall x(Px \rightarrow Qx)$ and $\exists x(Qx \wedge \neg Px)$ are true.
- The model has to be such that everything of which P is true is such that Q is also true of it, and where there is at least one thing Q is true of that P is not true of.
- The following model makes the first sentence true (Comprehension check: what about constants?):
  - **Domain:** 0

- P(_): 0
- Q(_): 0

- However, it doesn't make the second sentence true, since there is nothing that Q is true of of which P isn't also true. What we need is to add an object to our domain such that Q is true of it, but P isn't. We do this as follows:

  - **Domain:** 0,1
  - P(_): 0
  - Q(_): 0,1

- This model is such that both $\forall x(Px \to Qx)$ and $\exists x(Qx \land \neg Px)$ are true with respect to it.