1 A note on scores

In principle, our semantic universals of interest (i.e. `monotonicity`, `quantity`, and `conservativity`) are binary features. However, for our purposes we would like graded measures for those features. Therefore, we use some intuitively appealing measures that are based on the amount of information certain properties of the model give us with respect to our certainty of knowing whether or not a model with that property satisfies the quantifier in question. Our main criteria here, however, is simply the fact that if a quantifier satisfies the universal, then our scoring function for that property should map the quantifier to 1. But it is to some extent "arbitrary" how the scoring function behaves for quantifiers that do not fully satisfy the property in question.

For each measure we define some form of information that we deem relevant. For example, in the case of defining a score for `quantity`, we use the tuple that describes the cardinality of the 4 subsets in a model. More specifically, we measure how much information this 4-tuple provides with regards to knowing if a quantifier is true in that model. For a quantifier satisfying the `quantity` universal, we know that either all or none of the models that have the same "cardinality-4-tuple" will satisfy this quantifier.

1.1 Definitions

Let $\mathbb{M}$ be the space of the all models. For a given quantifier Q, let $1_Q : \mathbb{M} \to \{0, 1\}$ be the random variable that maps each model to a 0 or 1 if the quantifier is true or false on that model, respectively. Furthermore, let $\Gamma_m, \Gamma_q, \Gamma_c : \mathbb{M} \to \mathbb{N}^*$ be *signature functions* for `monotonicity`, `quantity`, and `conservativity`, respectively. *Signature functions* are random variables that map each model to some tuple with information relevant to the score in question. For example, in the case of `quantity`, Γ_q maps each model to a tuple that describes the cardinality of each of the 4 subsets in the model. Note that each signature function induces a partition of the model space.

1.1.1 Model representation

We represent models of size k here by a $2 \times k$ matrix, where the ith column denotes the i^{th} object of the model. Each object then is represented by a 2-vector, where the first value is a 1 iff this object is in A, and the second element denotes membership of B. So for example

$$\left|\begin{array}{c|c|c|c} 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{array}\right|$$

denotes a model where the first object is in $A \cap B$, the second element is in $B \setminus A$, the third element is in $A \setminus B$, and the last element is in $M \setminus (A \cup B)$.

1.2 Scoring function

Let $x \in \{m, q, c\}$, we then define a scoring function for x, S_x on a given quantifier Q as follows:

$$
\begin{aligned}
S_x(Q) &= 1 - \frac{H(1_Q|\Gamma_x)}{H(1_Q)} \\
&= 1 - \frac{\sum_{\gamma \in \operatorname{Im} \Gamma_x} P(\Gamma_x = \gamma) H(1_Q|\Gamma_x = \gamma)}{H(1_Q)}
\end{aligned}
$$

Where for each $\gamma \in \Gamma_x$:

$$
\begin{aligned}
&H(1_Q|\Gamma_x = \gamma) \\
&= \sum_{b \in \operatorname{Im} 1_Q} P(1_Q = b|\Gamma_x = \gamma) \log(\frac{1}{P(1_Q = b|\Gamma_x = \gamma)}) \\
&= P(1_Q = 0|\Gamma_x = \gamma) \log(\frac{1}{P(1_Q = 0|\Gamma_x = \gamma)}) + P(1_Q = 1|\Gamma_x = \gamma) \log(\frac{1}{P(1_Q = 1|\Gamma_x = \gamma)}) \\
&= h(P(1_Q = 1|\Gamma_x = \gamma))
\end{aligned}
$$

where h is the binary entropy function.

1.2.1 Quantity

For quantity, Γ_q computes a tuple that describes the quantities in each of the 4 subsets of the model. So for a given model, M, $\Gamma_q(M) := (|A|, |B|, o|A \cap B|,)$.

1.2.2 Conservativity

For conservativity, given a model M, $\Gamma_c(M)$ outputs a model like M, but with all objects that are not in A discarded. Since we are working with order-sensitive (i.e. index-based) quantifiers, the order of the members are preserved. So for example if we have the following model M:

1	0	1	0
1	1	0	0

the conservativity signature $\Gamma_c(M)$ for this model is:

1	1
1	0

The intuition behind this is that if a quantifier is perfectly conservative, "addition" or "insertion" of any objects to $\Gamma_c(M)$ that are not in A should not affect the truth value of this quantifier on this model, so:
$\Gamma_c(M) \in Q \leftrightarrow M \in Q$

1.2.3 Monotonicity

For monotonicy, for a particular quantifier Q, we say that:
$\Gamma_m(M) = 1$ iff there is a submodel M' such that $M' \in Q$.
Now in this order-sensitive context, we define a submodel as follows:
M' is a submodel of M iff $|A(M)| = |A(M')|$ and for each i^{th} object x'_i in $A(M')$:
if $x'_i \in B(M')$, then the i^{th} object x_i in $A(M)$ is in $B(M)$ as well. (Where A and B are the predicates that take a model and return the tuple of objects that belong to A or B respectively).
More concretely, for a given model M:

1	0	1	0
1	1	1	0

an example of a submodel M' would be:

1	1	0
1	0	0

or M'':

1	0	0	0	1	0	0	0
1	1	1	0	1	1	1	0

Clearly, if $M' \in Q$ for some upward monotone quantifier Q, then $M \in Q$ too. The same holds for M''.
Intuitively, this notion of a submodel captures the idea that if a quantifier is upward monotone (in the order context), and is true on a model, then adding/inserting objects to B should not affect the truth value of this quantifier on the model.
However, since we cannot nicely partition the model space based on the information of a submodel itself, we partition it based on if there is a submodel that belongs to the quantifier.

1.3 Note on difference in scores

Note that there is a difference in the way we measure monotonicity vs. quantity and conservativity. In the case of quantity and conservativity, for a quantifier Q we use a particular piece of information of the model that is independent of the Q, and measure the extent to which this piece of information reduces the entropy or 'uncertainty' about knowing whether $M \in Q$ or $M \notin Q$.
In the case of monotonicity, in contrast, the piece of information is quantifier-specific: is there a submodel M' of M such that $M' \in Q$. Clearly, if the answer is no, then it removes all uncertainty, because we then know that $M \notin Q$. If the answer is yes and the quantifier is monotone, we know for sure that $M \in Q$ too.