2. Propositional and predicate calculus

Propositional calculus
- It is a language of assertions that evaluate to true or false
- Assertion formulas include:
  - true, false
  - Symbols (p, q, r, ...) that may stand for assertions such as “roses are red”
  - Negations: ¬p
  - Conjunctions: p ∧ q
  - Disjunctions: p ∨ q
  - Implications: p → ¬(q ∨ ¬p)

Semantics
- Let p, q, be formulas
- Wherever p is true, ¬p is false
- (p ∧ q) is true when both p or q are true
- (p ∨ q) is true when either p or q or both are true
- (p → q) is true when p is false or q is true
- (p ⇔ q) or (p ≡ q) or (p iff q) is true when p and q have the same values under all interpretations (truth assignments)

Interpretations
- An interpretation of a set of formulas is an assignment of truth values to the symbols
- Example: An interpretation of (p ∧ ¬(p ∨ q)) is p = true, q = false. Under this interpretation, (p ∧ ¬(p ∨ q)) is false
- A formula is satisfiable if it has an interpretation under which it is true

Truth tables
- A truth table lists the values of a formula under all possible interpretations
- Example:
<table>
<thead>
<tr>
<th>p</th>
<th>q</th>
<th>(p ∧ ¬q)</th>
</tr>
</thead>
<tbody>
<tr>
<td>f</td>
<td>f</td>
<td>f</td>
</tr>
<tr>
<td>f</td>
<td>t</td>
<td>f</td>
</tr>
<tr>
<td>t</td>
<td>f</td>
<td>f</td>
</tr>
<tr>
<td>t</td>
<td>t</td>
<td>f</td>
</tr>
</tbody>
</table>
- Formulas with the same truth table are equivalent: for formulas φ and ϕ, (φ ⇔ ϕ) or (ϕ = φ) or (ϕ iff φ)

Predicate calculus
- Predicate calculus extends propositional calculus by enabling quantifiers and formulas in predicate (functional) form
- A predicate is a Boolean function
- Examples:
  - (∀x) x < x + 1 universal quantifier
  - prime(5) 1-ary predicate
  - (∃x) x > 1 existential quantifier
- Arity of a predicate is its number of parameters

Variables in the predicate calculus
- Suppose x is a variable in the domain of natural numbers
- In expressions such as sum(x, 5), odd(x), prime(x), x is a placeholder that stands for an unknown constant
- “sum(x, 5) = 8”, “odd(x) = true” are meaningless without having a value for x
- “x = 3 → sum(x, 5) = 8”, “x = 1 → odd(x) = true” are meaningful and true statements

Quantifiers
- Meaningful:
  - (∃x) x + 5 = 8
  - (∀x) x + 1 = x
- Quantifiers bind variables so that they can be used meaningfully in sentences

Semantics of predicate calculus
- In a real-world domain, there are relations:
  - female(h_clinton) unary
  - married(b_obama, m_obama) binary
- Predicates female, married here express these relations
- For a domain D, an interpretation over D is an assignment of entities in D to constants, assignment of non-empty subsets of D to variables, with each predicate mapping D^n → (t, f)
Truth value (semantics) of predicate-calculus expressions

- Given expression E, and interpretation I for E over domain D, truth value is as in propositional calculus, in addition to:
  - For sentence S,
    - \((\exists x) S\) is true iff some assignment under I has an assignment to x s.t. S is true
    - \((\forall x) S\) is true iff S is true for all assignments of values to x under I

Examples

1. If it doesn’t rain, Rose will go to the Park
   \(\neg \text{weather}(\text{rain}) \rightarrow \text{go}(\text{Rose}, \text{park})\)

2. Spark is a Spaniel and a good dog
   \(\text{isa}(\text{Spark}, \text{Spaniel}) \land \text{good-dog}(\text{Spark})\)

3. All basketball players are tall
   \((\forall x) (\text{basketball-player}(x) \rightarrow \text{tall}(x))\)

4. Nobody likes taxes
   \(\neg (\exists x) (\text{likes}(x, \text{taxes}))\)

Satisfiability, validity, consistency

- An interpretation satisfies a sentence if it makes the sentence true.
- If I satisfies sentence S for every set of user assignments, then I is a model of S
- S or a set of sentences is satisfiable if some I satisfies it
- An unsatisfiable set of expressions is unsatisfiable
- If S is true for all interpretations, then S is valid
- \((\forall x) (P(x) \lor \neg P(x))\) is valid
- A sentence that is not true under any interpretation is inconsistent
- \((\forall x) (P(x) \land \neg P(x))\) is inconsistent

Inference

- Expression x logically follows from a set S of sentences iff every interpretation that satisfies S also satisfies x
- Inference rule: A validity-maintaining procedure for deriving sentences from other sentences
- Proof procedure: An inference rule and an algorithm for applying it to a set of sentences to yield a new sentence

Soundness and completeness

- Sound: an inference rule that produces from set S only sentences that logically follow from S
- Complete: a rule that can infer from S every sentence that logically follows from S
- Examples:
  - Modus Ponens: \((p \land (p \rightarrow q)) \rightarrow q\)
  - Modus Tollens: \((p \rightarrow q) \land \neg q) \rightarrow \neg p\)
  - And elimination, and introduction, universal instantiation

Unification

- An algorithm for determining what substitutions are needed to make two sentences match
- Cases for unify \((E1, E2)\):
  - \(E1 = E2\): Return \(\emptyset\) (no substitution)
  - \(E1\) is variable: return \(\{E2 / E1\}\)
  - \(E2\) is variable: return \(\{E1 / E2\}\)
  - Otherwise decompose E1, E2, return a composition of unification of the components

Predicate calculus and knowledge representation

- Predicate calculus is a way to represent knowledge of
  - Objects
  - Relations
- Inference rules enable derivation of new knowledge

---

74 Ibid., p. 58
75 Ibid., p. 60
76 Ibid., p. 62
77 Idem.
78 Ibid., pp. 64-65
79 Ibid., pp. 66
80 Ibid., p. 79