european social fund in the czech republic
EUROPEAN UNION
MINISTRY OF EDUCATION, YOUTH AND SPORTS
OP Education for Competitiveness
INVESTMENTS IN EDUCATION DEVELOPMENT

1 Inductive inference in predicate logic

We still use Prolog notation for both the data and the learned hypotheses.

Exercise 1.1: Let us have the three-digit numbers categorization (the same as in the lecture):

example	139	319	854	468	349	561	756	789	987	256	189	354
classif.	+	−	−	+	+	−	−	+	−	+	+	−

a) specify the domain knowledge

b) draw the relevant part of a specialization graph

c) compare the learned hypothesis with the decision tree for the original task

d) for the correct and complete solution: what (minimal) number of examples is needed for learning the decision tree? And how many for learning the Prolog hypothesis?

Exercise 1.2: Represent the following task using predicate logic (it has already been represented in propositional logic).
There are the following data with three attributes classified into two classes `true/false`:

`Size` ∈ {`small, medium, large`},
`Color` ∈ {`red, blue, green`},
`Shape` ∈ {`square, circle, triangle`}

`small`	`red`	`triangle`	`true`
`small`	`green`	`triangle`	`true`
`large`	`red`	`triangle`	`false`
`small`	`blue`	`circle`	`false`

Find one or more proper specializations and one or more proper generalizatons of the following clauses. Do not specialize the variable `Id`.

a) `p(Id) :- size(Id,large), color(Id,red).`

b) `p(Id) :- color(Id,red).`

c) `p(Id) :- size(Id,large), color(Id,red), shape(Id,circle).`

d) `p(Id).`

e) Find all specializations of the clause `p(Id) :- color(Id,red).` that cover the example `<large,red,square>`.

f) Find out whether (and how) the formulas from items a)–d) are in the generalization/specialization relation.

Exercise 1.3:

a) Draw the specialization graph for the predicate `member/2`.

b) Describe the changes in the graph for the predicate `last/2`.

c) Write lists of training examples for these predicates.

Exercise 1.4: Describe the construction of a specialization graph for `reverse/2`. The domain knowledge consists of the predicate `append/3`.