Knowledge in Learning

Chapter 19, Section 1

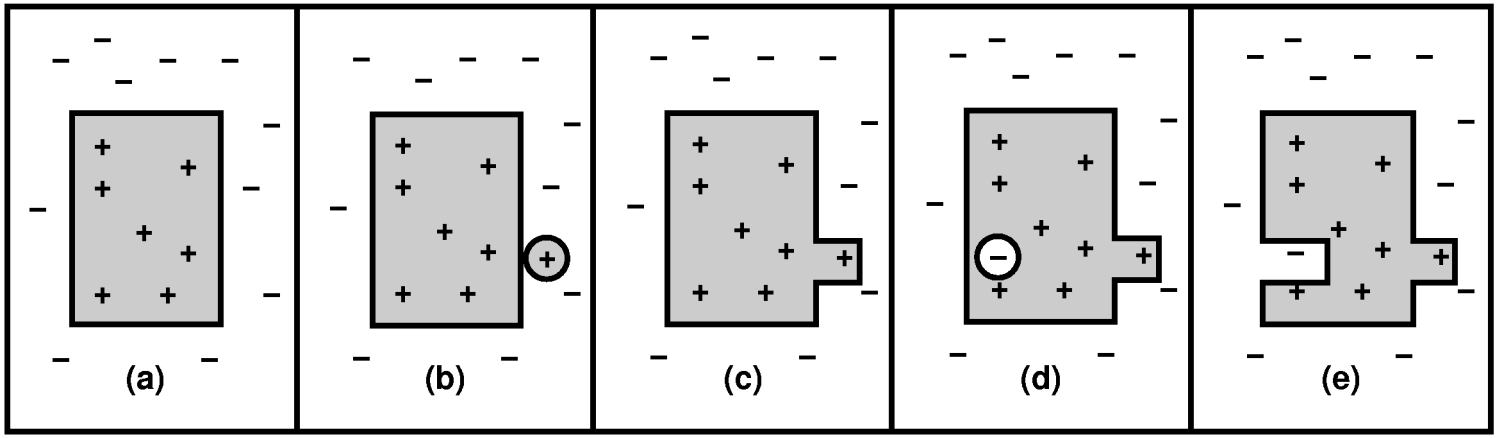
© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

Outline

- Incremental learning
- Version space learning

© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

Incrementally updating a hypothesis



© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

Specialization vs. generalization

- **False positive**: the hypothesis classifies an example as positive when it really is negative
  - We need to **specialize** the hypothesis to exclude the negative example
- **False negative**: the hypothesis classifies an example as negative when it really is positive
  - We need to **generalize** the hypothesis to include the positive example

© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

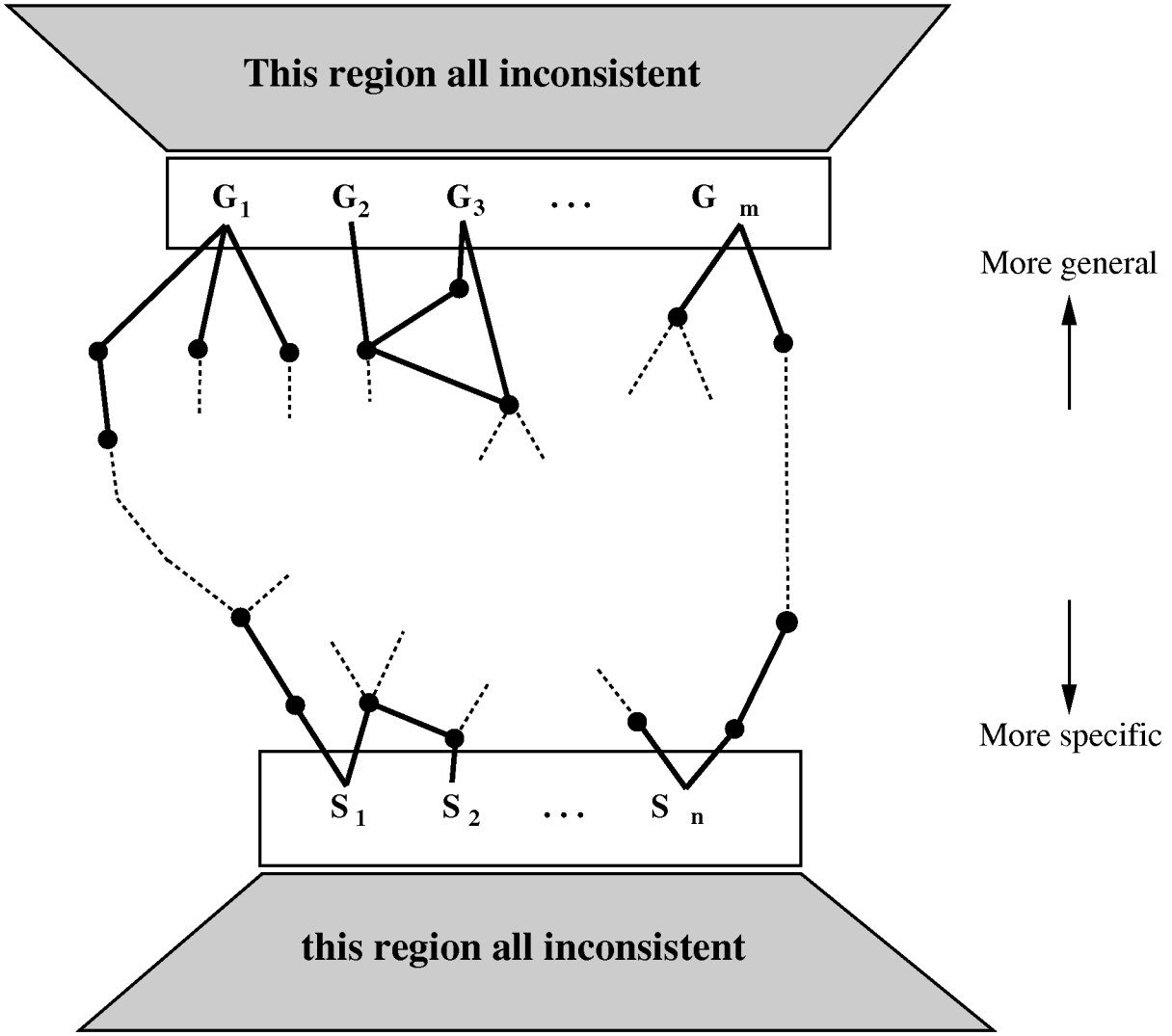
Current-best learning

```
function CURRENT-BEST-LEARNING(examples) returns a hypothesis

   H ← any hypothesis consistent with the first example in examples
   for each remaining example in examples do
       if e is false positive for H then
           H ← choose a specialization of H consistent with examples
       else if e is false negative for H then
           H ← choose a generalization of H consistent with examples
       if no consistent specialization/generalization can be found then fail
   end
   return H
```

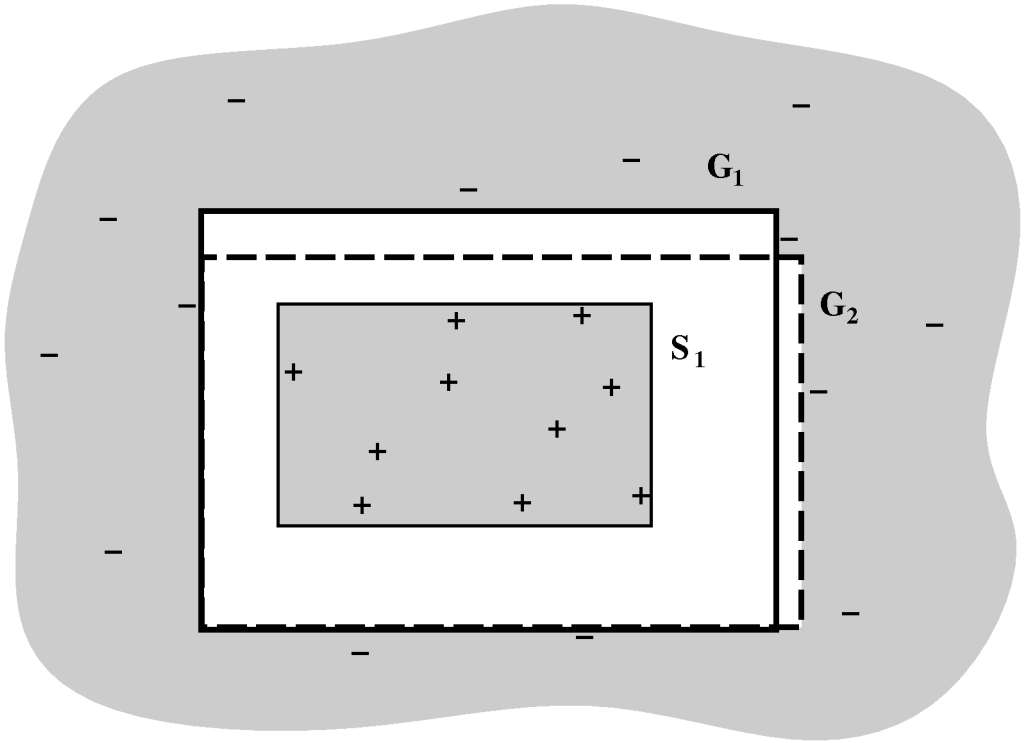
© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

The version space



© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

The boundaries of the version space



© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

General-to-specific search

- Start with most general hypothesis
- Until all examples have been processed
  - If current example is negative
    - Replace all hypotheses that cover the example by their most general specializations that do not cover the example
    - Delete hypotheses that are more specific than other ones
    - Delete hypotheses not consistent with all positive examples
  - If current example is positive
    - Delete all hypotheses not consistent with example

© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

Specific-to-general search

- Start with most specific hypothesis
- Until all examples have been processed
  - If current example is positive
    - Replace all hypotheses that do not cover the example by their most specific generalizations that cover the example
    - Delete hypotheses that are more general than other ones
    - Delete hypotheses covering a negative example
  - If current example is negative
    - Delete all hypotheses covering the example

© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

Comments on the two algorithms

- General-to-specific search needs to maintain
  - The set of most general hypotheses *G*
  - The set of positive examples seen so far
- Specific-to-general search needs to maintain
  - The set of most specific hypotheses *S*
  - The set of negative examples seen so far
- Storing the examples can be avoided by maintaining the set of all consistent hypotheses!

© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

Version space learning

function VERSION-SPACE-LEARNING(*examples*) **returns** a version space
  local variables: V, the version space: the set of all hypotheses

  $V \leftarrow$ the set of all hypotheses
  for each example e in *examples* **do**
    if V is not empty **then** $V \leftarrow$ VERSION-SPACE-UPDATE(V, e)
  end
  return V

---

function VERSION-SPACE-UPDATE(V, e) **returns** an updated version space

  $V \leftarrow \{h \in V : h \text{ is consistent with } e\}$

© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

Maintaining the version space

- Naïve approach: explicitly storing all the consistent hypotheses
- Observation: version space fully specified by boundary sets *S* and *G*
  - *S*: set of most specific hypotheses consistent with the examples seen so far
  - *G*: set of most general hypotheses consistent with the examples seen so far
- We only need to maintain *S* and *G* !

© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

Incorporating a positive example

- Delete G_i that do not cover the example
- Replace all S_i that do not cover the example by their most specific generalizations that cover it
- Delete S_i that are more general than some other member of S
- Delete S_i that are not more specific than (or equal to) at least one member of G

© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

Incorporating a negative example

- Delete S_i that cover the example
- Replace all G_i that cover the example by their most general specializations that do not cover it
- Delete G_i that are less general than some other member of G
- Delete G_i that are not more general than (or equal to) at least one member of S

© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)

Termination of the algorithm

- First possibility: we run out of data
  - Several hypotheses survive
- Second possibility: *S* and *G* converge
  - We have found the only consistent hypothesis
- Third possibility: version space collapses
  - No hypothesis survives
  - This may be due to noise or an inappropriate hypothesis space

© Eibe Frank 2006 (all figures © Stuart Russell and Peter Norvig, 2003)