SGN-13006 Introduction to Pattern Recognition and Machine Learning (5 cr)

Concept Learning

Joni-Kristian Kämäräinen

November 2017

Laboratory of Signal Processing
Tampere University of Technology

Material

- Lecturer's slides and blackboard notes
- T.M. Mitchell. *Machine Learning*. McGraw-Hill, 1997: Chapter 2

Contents

General and Specific Concepts

Concepts

Positive and negative examples: *a training set*

Sky	Temp	Humid	Wind	Water	Forecst	EnjoySpt
Sunny	Warm	Normal	Strong	Warm	Same	Yes
Sunny	Warm	High	Strong	Warm	Same	Yes
Rainy	Cold	High	Strong	Warm	Change	No
Sunny	Warm	High	Strong	Cool	Change	Yes

The inductive learning hypothesis: Any hypothesis found to approximate the target function well over a sufficiently large set of training examples will also approximate the target function well over other unobserved examples.

Representing Hypotheses

- Many possible representations
- Here, h is conjunction of constraints on attributes
- Each constraint can be
  - a specfic value (e.g., $Water = Warm$)
  - don't care (e.g., "$Water =?$")
  - no value allowed (e.g.,"Water=$\emptyset$")

For example,

Sky	AirTemp	Humid	Wind	Water	Forecst
$\langle Sunny$	?	?	$Strong$	?	$Same\rangle$

FIND-S Algorithm

Find-S algorithm

```
Initialize h to the most specific hypothesis in H
for For each positive training instance x do
    for For each attribute constraint a_i in h do
        if the constraint a_i in h is satisfied by x then
            do nothing
        else
            replace a_i in h by the next more general constraint that
            is satisfied by x
        end if
    end for
end for
Output hypothesis h
```

Complaints about FIND-S

1. Can't tell whether it has learned concept
2. Can't tell when training data inconsistent
3. Picks a maximally specific h (why?)
4. Depending on H, there might be several!

Candidate Elimination Algorithm

Candidate Elimination Algorithm

Version spaces

Version Spaces

A hypothesis h is **consistent** with a set of training examples D of target concept c if and only if $h(x) = c(x)$ for each training example $\langle x, c(x) \rangle$ in D.

$$Consistent(h, D) \equiv (\forall \langle x, c(x) \rangle \in D)\ h(x) = c(x)$$

The **version space**, $VS_{H,D}$, with respect to hypothesis space H and training examples D, is the subset of hypotheses from H consistent with all training examples in D.

$$VS_{H,D} \equiv \{h \in H | Consistent(h, D)\}$$

The List-Then-Eliminate Algorithm

1: *VersionSpace* $\leftarrow$ a list containing every hypothesis in H
2: For each training example, $\langle x, c(x) \rangle$
3: remove from *VersionSpace* any hypothesis h for which $h(x) \neq c(x)$
4: Output the list of hypotheses in *VersionSpace*

Representing Version Spaces

- The **General boundary**, G, of version space $VS_{H,D}$ is the set of its maximally general members
- The **Specific boundary**, S, of version space $VS_{H,D}$ is the set of its maximally specific members
- Every member of the version space lies between these boundaries

$$VS_{H,D} = \{h \in H | (\exists s \in S)(\exists g \in G)(g \geq h \geq s)\}$$

where $x \geq y$ means x is more general or equal to y

Candidate Elimination Algorithm

```
G ← maximally general hypotheses in H
S ← maximally specific hypotheses in H
for each training example d do
    if d is a positive example then
        Remove from G any hypothesis inconsistent with d
        for each hypothesis s in S that is not consistent with d do
            Remove s from S
            Add to S all minimal generalizations h of s such that h is consistent with d, and some member
            of G is more general than h
            Remove from S any hypothesis that is more general than another hypothesis in S
        end for
    end if
    if d is a negative example then
        Remove from S any hypothesis inconsistent with d
        for each hypothesis g in G that is not consistent with d do
            Remove g from G
            Add to G all minimal specializations h of g such that h is consistent with d, and some member
            of S is more specific than h
            Remove from G any hypothesis that is less general than another hypothesis in G
        end for
    end if

end for
```

Summary

Summary

1. A concept: Definition and representation

Summary

1. A concept: Definition and representation
2. Concept learning as search

Summary

1. A concept: Definition and representation
2. Concept learning as search
3. General and specific hypotheses

Summary

1. A concept: Definition and representation
2. Concept learning as search
3. General and specific hypotheses
4. FIND-S algorithm: Finding maximally specific hypothesis

Summary

1. A concept: Definition and representation
2. Concept learning as search
3. General and specific hypotheses
4. FIND-S algorithm: Finding maximally specific hypothesis
5. The CANDIDATE-ELIMINATION algorithm: Version spaces