Machine Learning: Exercise Sheet 2

Manuel Blum
AG Maschinelles Lernen und Natürlichsprachliche Systeme
Albert-Ludwigs-Universität Freiburg

mblum@informatik.uni-freiburg.de

Exercise 1: Version Spaces
Task (a)

What are the elements of the version space?

- hypotheses (descriptions of concepts)
- $VS_{H,D} \subseteq H$ with respect to the hypothesis space H contains those hypotheses that are consistent with the training data D

How are they ordered?

- arranged in a general-to-specific ordering
- partial order: $\leq_g$, $<_g$

Exercise 1: Version Spaces
Task (a)

What can be said about the meaning and sizes of G and S?

- They are sets containing the most general and most specific hypotheses consistent with the training data. Thus, they depict the general and specific boundary of the VS.
- For conjunctive hypotheses (which we consider here) it always holds $|S| = 1$, assuming consistent training data. G attains its maximal size, if negative patterns with maximal hamming distance have been presented. Thus, in the case of binary constraints, it holds $|G| \leq n(n-1)$ where n denotes the number of constraints per hypothesis.

Exercise 1: Version Spaces

Task (b)

In the following, it is desired to describe whether a person is *ill*. We use a representation based on conjunctive constraints (three per subject) to describe individual person. These constraints are "running nose", "coughing", and "reddened skin", each of which can take the value true ('+') or false ('–'). We say that somebody is ill, if he is coughing and has a reddened nose — each single symptom individually does not mean that the person is ill.

- Specify the space of hypotheses that is being managed by the version space approach. To do so, arrange all hypotheses in a graph structure using the more-specific-than relation.
  - hypotheses are vectors of constraints, denoted by $\langle N, C, R \rangle$
  - with $N, C, R = \{-, +, \emptyset, *\}$

Exercise 1: Version Spaces
Task (c)

Apply the candidate elimination (CE) algorithm to the sequence of training examples specified in the table and name the contents of the sets S and G after each step.

Training Example	N (running nose)	C (coughing)	R (reddened skin)	**Classification**
d_1	$+$	$+$	$+$	positive (ill)
d_2	$+$	$+$	$-$	positive (ill)
d_3	$+$	$-$	$+$	negative (healthy)
d_4	$-$	$+$	$+$	negative (healthy)
d_5	$-$	$-$	$+$	negative (healthy)
d_6	$-$	$-$	$-$	negative (healthy)

Exercise 1: Version Spaces
Task (c)

- Start (init): $G = \{\langle * * * \rangle\}$, $S = \{\langle \emptyset \emptyset \emptyset \rangle\}$
- **foreach** $d \in D$ **do**
  - $d_1 = [\langle + + + \rangle, pos] \Rightarrow G = \{\langle * * * \rangle\}$, $S = \{\langle + + + \rangle\}$
  - $d_2 = [\langle + + - \rangle, pos] \Rightarrow G = \{\langle * * * \rangle\}$, $S = \{\langle + + * \rangle\}$
  - $d_3 = [\langle + - + \rangle, neg]$
    - no change to S: $S = \{\langle + + * \rangle\}$
    - specializations of G: $G = \{\langle - * * \rangle, \langle * + * \rangle, \langle * * - \rangle\}$
    - there is no element in S that is more specific than the first and third element of G
      $\rightarrow$ remove them from $G \Rightarrow G = \{\langle * + * \rangle\}$

Exercise 1: Version Spaces

Task (c)

- **foreach** $d \in D$ **do**
  - *loop continued ...*
  - so far we have $S = \{\langle + + *\rangle\}$ and $G = \{\langle * + *\rangle\}$
  - $d_4 = [\langle - + +\rangle, neg]$
    - no change to S: $S = \{\langle + + *\rangle\}$
    - specializations of G: $G = \{\langle + + *\rangle, \langle * + -\rangle\}$
    - there is no element in S that is more specific than the second element of G
      $\rightarrow$ remove it from $G \Rightarrow G = \{\langle + + *\rangle\}$
  - Note:
    - At this point, the algorithm might be stopped, since $S = G$ and no further changes to S and G are to be expected.
    - However, by continuing we might detect inconsistencies in the training data.

Exercise 1: Version Spaces
Task (c)

- Start (init): $G = \{\langle * * * \rangle\}$, $S = \{\langle \emptyset \emptyset \emptyset \rangle\}$
- **foreach** $d \in D$ **do**
  - *loop continued ...*
  - $d_5 = [\langle - - + \rangle, neg] \Rightarrow$ Both, $G = \{\langle + + * \rangle\}$ and $S = \{\langle + + * \rangle\}$ are consistent with d_5.
  - $d_6 = [\langle - - - \rangle, neg] \Rightarrow$ Both, $G = \{\langle + + * \rangle\}$ and $S = \{\langle + + * \rangle\}$ are consistent with d_6.
- **return** S and G

Exercise 1: Version Spaces
Task (d)

Does the order of presentation of the training examples to the learner affect the finally learned hypothesis?

- No, but it may influence the algorithm's running time.

Exercise 1: Version Spaces
Task (e)

Assume a domain with two attributes, i.e. any instance is described by two constraints. How many positive and negative training examples are *minimally* required by the candidate elimination algorithm in order to learn an arbitrary concept?

- By learning an arbitrary concept, of course, we mean that the algorithm arrives at $S = G$.
- The algorithm is started with $S = \{\langle \emptyset, \emptyset \rangle\}$ and $G = \{\langle *, * \rangle\}$.
- We just consider the best cases, i.e. situations in where the training instances given to the CE algorithm allow for adapting S or G.

Exercise 1: Version Spaces
Task (e)

Clearly, three appropriately chosen examples are sufficient.

- Negative Examples: Change G from $\langle *, * \rangle$ to $\langle v, * \rangle$ or $\langle *, w \rangle$. Or they change G from $\langle v, * \rangle$ or $\langle *, w \rangle$ to $\langle v, w \rangle$.
- Positive Examples: Change S from $\langle \emptyset, \emptyset \rangle$, $\langle v, w \rangle$. Or they change S from $\langle v, w \rangle$ to $\langle v, * \rangle$ or $\langle *, w \rangle$. Or from $\langle v, * \rangle$ or $\langle *, w \rangle$ to $\langle *, * \rangle$.
- At least one positive example is required (otherwise S remains $\langle \emptyset, \emptyset \rangle$).
- Special case: Two positive patterns $\langle d_1, d_2 \rangle$, $\langle e_1, e_2 \rangle$ are sufficient, if it holds $d_1 \neq e_1$ and $d_2 \neq e_2$.
  $\Rightarrow S = \langle \emptyset, \emptyset \rangle \rightarrow \langle d_1, d_2 \rangle \rightarrow \langle *, * \rangle$

Exercise 1: Version Spaces

Task (f)

We are now extending the number of constraints used for describing training instances by one additional constraint named "fever". We say that somebody is ill, if he has a running nose and is coughing (as we did before), or if he has fever.

Training Example	N (running nose)	C (coughing)	R (reddened skin)	F (fever)	**Classification**
d_1	$+$	$+$	$+$	$-$	positive (ill)
d_2	$+$	$+$	$-$	$-$	positive (ill)
d_3	$-$	$-$	$+$	$+$	positive (ill)
d_4	$+$	$-$	$-$	$-$	negative (healthy)
d_5	$-$	$-$	$-$	$-$	negative (healthy)
d_6	$-$	$+$	$+$	$-$	negative (healthy)

Exercise 1: Version Spaces

Task (f)

How does the version space approach using the CE algorithm perform now, given the training examples specified on the previous slide?

- Initially: $S = \{\langle \emptyset\emptyset\emptyset\emptyset \rangle\}$, $G = \{\langle * * ** \rangle\}$
- $d_1 = [\langle + + + - \rangle, pos] \Rightarrow S = \{\langle + + + - \rangle\}$, $G = \{\langle * * ** \rangle\}$
- $d_2 = [\langle + + - - \rangle, pos] \Rightarrow S = \{\langle + + * - \rangle\}$, $G = \{\langle * * ** \rangle\}$
- $d_3 = [\langle - - + + \rangle, pos] \Rightarrow S = \{\langle * * ** \rangle\}$, $G = \{\langle * * ** \rangle\}$
  $\rightarrow$ We already arrive at $S = G$.
- $d_4 = [\langle + - - - \rangle, neg] \Rightarrow S = \{\langle * * ** \rangle\}$, $G = \{\langle * * ** \rangle\}$
  - Now, S becomes empty since $\langle * * ** \rangle$ is inconsistent with d_4 and is removed from S.
  - G would be specialized to $\{\langle - * ** \rangle, \langle * + ** \rangle, \langle * * + * \rangle, \langle * * * + \rangle\}$. But it is required that at least one element from S must be more specific than any element from G.
    $\rightarrow$ This requirement cannot be fulfilled since $S = \emptyset$. $\Rightarrow G = \emptyset$

Exercise 1: Version Spaces
Task (f)

What happens, if the order of presentation of the training examples is altered?

- Even a change in the order of presentation does not result in yielding a learning success (i.e. in $S = G \neq \emptyset$).
- When applying the CE algorithm, S and G become empty independent of the presentation order.
- Reason: The informally specified target concept of an "ill person" represents a disjunctive concept.
- The target concept is not an element of the hypothesis space H (which is made of conjunctive hypotheses).

Exercise 2: Decision Tree Learning with ID3
Task (a)

Apply the ID3 algorithm to the training data in the table.

Training	fever	vomiting	diarrhea	shivering	**Classification**
d_1	no	no	no	no	healthy (H)
d_2	average	no	no	no	influenza (I)
d_3	high	no	no	yes	influenza (I)
d_4	high	yes	yes	no	salmonella poisoning (S)
d_5	average	no	yes	no	salmonella poisoning (S)
d_6	no	yes	yes	no	bowel inflammation (B)
d_7	average	yes	yes	no	bowel inflammation (B)

Exercise 2: Decision Tree Learning with ID3

Task (a)

Exemplary calculation for the first (root) node.

- entropy of the given data set S: $Entropy(S)$
  $= -\frac{1}{7}\log_2(\frac{1}{7}) - \frac{2}{7}\log_2(\frac{2}{7}) - \frac{2}{7}\log_2(\frac{2}{7}) - \frac{2}{7}\log_2(\frac{2}{7}) = 1.950$
- consider attribute x= "Fever"

Values	H	I	S	B	$Entropy(S_i)$
S_1 (no)	*			*	$[\frac{1}{2}, 0, 0, \frac{1}{2}] \rightarrow 1$
S_2 (average)		*	*	*	$[0, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}] \rightarrow 1.585$
S_3 (high)		*	*		$[0, \frac{1}{2}, \frac{1}{2}, 0] \rightarrow 1$

$\Rightarrow Entropy(S|Fever) = \frac{2}{7} \cdot 1 + \frac{3}{7} \cdot 1.585 + \frac{2}{7} \cdot 1 = 1.251$

Exercise 3: Decision Tree Learning with ID3

Task (a)

- consider attribute x= "Vomiting"

Values	H	I	S	B	$Entropy(S_i)$
S_1 (yes)			*	**	$[0, 0, \frac{1}{3}, \frac{2}{3}] \rightarrow 0.918$
S_2 (no)	*	**	*		$[\frac{1}{4}, \frac{2}{4}, \frac{1}{4}, 0] \rightarrow 1.5$

$\Rightarrow Entropy(S|Vomiting) = \frac{3}{7} \cdot 0.918 + \frac{4}{7} \cdot 1.5 = 1.251$

- consider attribute x= "Diarrhea"

Values	H	I	S	B	$Entropy(S_i)$
S_1 (yes)			**	**	$[0, 0, \frac{2}{4}, \frac{2}{4}] \rightarrow 1$
S_2 (no)	*	**			$[\frac{1}{3}, \frac{2}{3}, 0, 0] \rightarrow 0.918$

$\Rightarrow Entropy(S|Diarrhea) = \frac{4}{7} \cdot 1 + \frac{3}{7} \cdot 0.918 = 0.965$

- consider attribute x= "Shivering"

Values	H	I	S	B	$Entropy(S_i)$
S_1 (yes)		*			$[0, 0, 1, 0] \rightarrow 0$
S_2 (no)	*	*	**	**	$[\frac{1}{6}, \frac{1}{6}, \frac{2}{6}, \frac{2}{6}] \rightarrow 1.918$

$\Rightarrow Entropy(S|Shivering) = \frac{1}{7} \cdot 0 + \frac{6}{7} \cdot 1.918 = 1.644$

Exercise 3: Decision Tree Learning with ID3

Task (a)

choose the attribute that maximizes the information gain

- Fever: $Gain(S) = Ent(S) - Ent(S|Fever) = 1.95 - 1.251 = 0.699$
- Vomiting: $Gain(S) = Ent(S) - Ent(S|Vomit) = 1.95 - 1.251 = 0.699$
- Diarrhea: $Gain(S) = Ent(S) - Ent(S|Diarrh) = 1.95 - 0.965 = 0.985$
- Shivering: $Gain(S) = Ent(S) - Ent(S|Shiver) = 1.95 - 1.644 = 0.306$

$\Rightarrow$ Attribute “Diarrhea” is the most effective one, maximizing the information gain.

Exercise 3: Decision Tree Learning with ID3

Task (b)

Does the resulting decision tree provide a disjoint definition of the classes?

- Yes, the resulting decision tree provides disjoint class definitions.

Exercise 3: Decision Tree Learning with ID3

Task (c)

Consider the use of real-valued attributes, when learning decision trees, as described in the lecture.
The data in the table below shows the relationship between the body height and the gender of a group of persons (the records have been sorted with respect to the value of *height* in cm).

Height	161	164	169	175	176	179	180	184	185
Gender	F	F	M	M	F	F	M	M	F

- Calculate the information gain for the potential splitting thresholds (recall that cut points must always lie at class boundaries) and determine the best one.
- Potential cut points must lie in the intervals $(164, 169)$, $(175, 176)$, $(179, 180)$, or $(184, 185)$.

Exercise 3: Decision Tree Learning with ID3

Task (c)

- Calculate the information gain for the potential splitting thresholds (ctd.).
- $C_1 \in (164, 169)$
  - resulting class distribution: if $x < C_1$ then $2 - 0$ else $3 - 4$
  - conditional entropy: if $x < C_1$ then $E = 0$ else $E = -\frac{3}{7}\log_2\frac{3}{7} - \frac{4}{7}\log_2\frac{4}{7} = 0.985$
  - entropy: $E(C_1|S) = \frac{2}{9} \cdot 0 + \frac{7}{9} \cdot 0.985 = 0.766$
- $C_2 \in (175, 176)$
  - resulting class distribution: if $x < C_2$ then $2 - 2$ else $3 - 2$
  - entropy: $E(C_2|S) = \frac{4}{9} \cdot 1 + \frac{5}{9} \cdot 0.971 = 0.984$
- $C_3 \in (179, 180)$
  - resulting class distribution: if $x < C_3$ then $4 - 2$ else $1 - 2$
  - entropy: $E(C_3|S) = \frac{6}{9} \cdot 0.918 + \frac{3}{9} \cdot 0.918 = 0.918$

Exercise 3: Decision Tree Learning with ID3

Task (c)

- Calculate the information gain for the potential splitting thresholds (ctd.).
- $C_4 \in (184, 185)$
  - resulting class distribution: if $x < C_4$ then $4 - 4$ else $1 - 0$
  - entropy: $E(C_4|S) = \frac{8}{9} \cdot 1 + \frac{1}{9} \cdot 0 = 0.889$
- Prior entropy of S is $-\frac{5}{9} \log_2 \frac{5}{9} - \frac{4}{9} \log_2 \frac{4}{9} = 0.991$.
- Information gain is $Gain(S, C_1) = 0.225$, $Gain(S, C_2) = 0.007$, $Gain(S, C_3) = 0.073$, and $Gain(S, C_4) = 0.102$
  $\rightarrow$ First splitting point (C_1) is the best one.