Zhuang Jianning T13
A0214561M

BL1

Prove that h is consistent with D iff every positive training instance satisfies h and every negative instance does not satisfy h

1. A hypothesis h is consistent with a set of training examples D iff h(x) = c(x) for all <x, c(x)> ∈ D. (definition of consistent hypothesis in slide 8)
2. A hypothesis h is consistent with D iff h(x) = 1 for all <x, 1> ∈ D and h(x) = 0 for all <x, 0> ∈ D. (target concept c maps X to {0, 1})
3. A hypothesis h is consistent with D iff every positive training instance satisfies h and every negative instance does not satisfy h (<x,1> is a positive example and definition of x satisfying h is h(x) = 1 in slide 7. <x,0> is a negative example and definition of x not satisfying h is h(x) = 0)

TM2.1

a) New input attribute WaterCurrent = {Light, Moderate, Strong} takes on 3 possible values.

   Input instances X do not have don't care or null attribute values
   |X| = number of possible input instances = 3x2x2x2x2x2x3 = 288

   Hypotheses containing 1 or more null values all classify as negative
   |H| = number of semantically distinct hypotheses = 4x3x3x3x3x3x4 + 1 = 3889

b) New input attribute that takes on k possible values.

   |X'| = k|X|
   |H'| = (|H|-1)(k+1) + 1 = k|H| + |H| - k

TM2.2

a) X1 = <Sunny, Warm, High, Strong, Cool, Change> +ve
   X2 = <Rainy, Cold, High, Strong, Warm, Change> -ve
   X3 = <Sunny, Warm, High, Strong, Warm, Same> +ve
   X4 = <Sunny, Warm, Normal, Strong, Warm, Same> +ve

S0 = {<null, null, null, null, null, null>}
S1 = {<Sunny, Warm, High, Strong, Cool, Change>}
S2 = {<Sunny, Warm, High, Strong, Cool, Change>}
S3 = {<Sunny, Warm, High, Strong, ?, ?>}
S4 = {<Sunny, Warm, ?, Strong, ?, ?>}

G4 = {<Sunny, ?, ?, ?, ?, ?>, <?, Warm, ?, ?, ?, ?>}
G3 = {<Sunny, ?, ?, ?, ?, ?>, <?, Warm, ?, ?, ?, ?>}
G2 = {<Sunny, ?, ?, ?, ?, ?>, <?, Warm, ?, ?, ?, ?>, <?, ?, ?, ?, Cool, ?>}
G1 = {<?, ?, ?, ?, ?, ?>}
G0 = {<?, ?, ?, ?, ?, ?>}

b) Starting from the most general G and specific S hypotheses, Candidate-Elimination Algorithm minimally generalizes S and specializes G for each training example to remove inconsistent hypotheses from the version space. The sequence of training examples does not matter as the algorithm will always output all the maximally general and specific hypotheses based on the training examples D. This only depends on the examples in D and not the sequence.

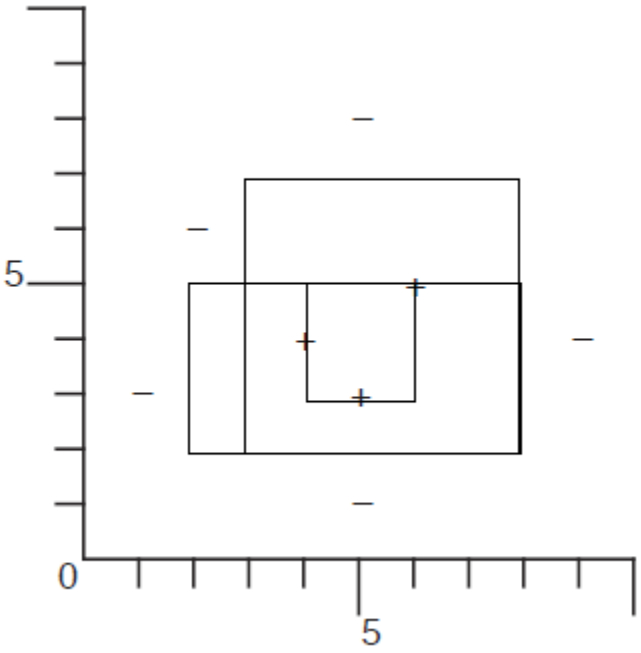
To minimize the sizes of intermediate S and G sets, we could minimally generalize S first with all the positive training examples, then minimally specialize G with all the rest of the negative examples. Starting with the most general G consisting of all ? (don't care), all positive examples will satisfy G, hence are consistent and will not be removed. Meanwhile, set S always maintains 1 maximally specific hypothesis because each inconsistent data removes the current s from S and adds a minimal generalization which usually just changes the attributes which did not meet the constraints. Since all the attributes need to be changed, we only have to add 1 new s with the changed attributes. After all the +ve examples, we get a maximally specific hypothesis which limits the minimal specializations of g as they each have to be more general than S.

TM2.4

a) All the points within the rectangle satisfy the hypothesis, hence should only contain + points.

   S = {<4 <= x <= 6, 3 <= y <= 5>}

b) All the - points should be just outside the rectangles
   Find maximum rectangle for both axis (vertically and horizontally)
   G = {<3 <= x <= 8, 2 <= y <= 7>, <2 <= x <= 8, 2 <= y <= 5>}



c) A query guaranteed to reduce the size of VS is an instance outside of S and within G. If it is positive, S minimally grows to be consistent with the new positive instance. If it is negative, G minimally shrinks to be consistent with the new negative instance.

   A query guaranteed to not reduce the size of VS is an instance within S or outside G. When it is within S, If it is positive, S and G are still consistent so there is no change to size of VS. If it is negative, there must be an error/noise since S has to shrink to be consistent with the negative example, but it is already the minimum size to be consistent with the other positive examples.

   When it is outside G, If it is negative, S and G are still consistent so there is no change to size of VS. If it is positive, there must be an error/noise since G has to grow to be consistent with the positive example, but it is already the maximum size to be consistent with the other negative examples.

d) To learn a particular target concept c, boundaries S must coincide with G so VS only contains c

Defining the minimum boundaries for S only requires the 2 positive examples in 2 opposite corners at the boundaries for the target concept. {(3,2), (5,9)}

Defining the boundaries for G requires to limit the 4 sides of G with negative examples. {(2,5), (6, 5), (4,1), (4,10)}

Total of 6 points