Inductive Learning

NetID : Drk170130

Name : Dhwaniben R Kaneria

1. Consider the problem of gradient descent that was discussed in class - you would like to predict the number of A grades that a student in the second year of the M.S. program receives (*y*) based on the number of A grades that the student received in the first year of the M.S. program (*x*). You propose ahypothesisoftheform*h*(*x*)=*θ*0+*θ*1*x*,where*θ*0and*θ*1areparametersthatyouwanttofind.The data is presentedbelow:

|  |  |
| --- | --- |
| **x** | **y** |
| 3 | 2 |
| 1 | 2 |
| 0 | 1 |
| 4 | 3 |

You start with an initial choice of parameters as: *θ*0 = 0 and *θ*1 = 1. You can assume that the error function is:

*i*=*m*

*J*= . (*h(x*(*i*)) *y*(*i*))2

1

*θ −*

2*m*

*i*=1

Where*m* is the number of training examples. Run at most 5 rounds of the gradient descent algorithm discussed in class. Does your error go down after 5 rounds? Show all the steps of your calculation.

* After 5 round of the gradient descent algorithm, Error goes down from 0.5 to 0.309.
* Iteration: 1 Error: 0.4057813 Ɵ0:0.0 Ɵ1:0.925

Iteration: 2 Error: 0.36099654 Ɵ0:0.007499999 Ɵ1:0.874375

Iteration: 3 Error: 0.33682162 Ɵ0:0.0196875 Ɵ1:0.8394531

Iteration: 4 Error: 0.32135132 Ɵ0:0.03475782 Ɵ1:0.8146621

Iteration: 5 Error: 0.30966428 Ɵ0:0.051553715 Ɵ1:0.7964211

* Program for calculating Error, Ɵ0and Ɵ1



1. Suppose there is a testing machine for a disease that can identify the disease in 80% of the cases, and also in 90% of the cases it is able to correctly predict those who do not have thedisease.

Identify the values of False Positive and False Negative in percent

If we consider having the disease to be positive and being healthy as negative then,

* 10% is False Positive.
* 20% is False Negative.

1. What are the pros and cons of thefollowing
   1. Selecting the most specific hypothesis (S) based on a trainingdata.
   2. Selecting the most general hypothesis (G) based on a training data.

* If we select most specific hypothesis, then there will be no room for generalization. So some positive examples might get wrong outcome.
* If we select most general hypothesis, then it will be too general. So some negative examples might get positive outcome.

So, it is better to select hypothesis, which lies between S and G.

a) Selecting the most specific hypothesis based on the training data

Pros:

1) The trainer will perform well on the given test data only and will not be able to classify other training examples accurately.

2) In case of starting with most specific hypothesis, we’ll have fewer combinations of attributes and therefore, fewer possible instances

Cons:

1) It can lead to overfitting of data.

2) This is further reflected in the find-S algorithm, which starts with the most specific hypothesis, and is inclined to pick a maximally specific h, that makes it hard to tell if the learner has learned the concept or not, as there may be many more h’s that fit.

b) Selecting the most general hypothesis based on the training data.

Pros:

1) It will perform well on the general training data.

2) Starting with a more general hypothesis will allow more instances to be classified as positive, as fewer restrictions are posed. In fact, all that is classified as positive by a general hypothesis hold true for specific hypothesis as well.

Cons:

1) It may contains noise as it is a general model.

2) If we start with most general hypothesis, the number of possible instances is huge and there are more combinations of attributes.

1. What is a consistent hypothesis and versionspace?

* Consistent hypothesis: it is belongs to version space. The hypothesis is said to be consistent if it is able to classify all the training examples accurately.
* Version Space: set of hypotheses contained in G, even in S, and lie between G and S in the hypothesis space.

1. The most generalhypothesishas don’t care value for eachattribute.
2. Consider the ML task of finding an approximation to the job finding problem for UTD students i.e. thefunction*f*:*X→Y* where*X*isthesetofattributesdefinedbelowand*Y* isabooleanoutput.

*X* = *(x*1*, x*2*, x*3*, x*4*)* such that

*x*1 is a boolean indicating whether GPA *≥* 3*.*5

*x*2 is a boolean indicating whether student has taken CS 6375

*x*3 is a boolean indicating whether student has taken CS 6350

*x*4 is a boolean indicating whether student has taken Years of Work Experience *>*2

For each attribute, there can be three possible choices - 1, 0, or ? (don’t care).

* 1. How many instances i.e. *|X|* are possible?
* Instances = 3 \* 3\* 3\* 3 =81
  1. How many labeling of these instances are possible? (Remember it’s binary classification problem and each labeling represents a possible hypothesis)
* Labeling: 281
  1. If you would like to limit the classifier to a decision tree of depth 2, how many hypotheses are possible?

Hint:First choose 2 attributes out of 4, create decision trees out of them, and find possible ways of labeling

* 4 hypotheses are possible if we are limiting our decision tree to depth of 2. And by choosing 2 attributes out 4 for one tree, then we can take 6 different combinations.
* 24 hypotheses are possible.

1. Applythe **Find-Salgorithm**onthefollowingdatasetforUTDstudents.Thereare5attributes

*xGPA* is a boolean indicating whether GPA *>*3*.*5

*xWorkEx* is a boolean indicating whether Years of Work Experience *>*2 *xCS*6375 is a boolean indicating whether student has taken CS 6375 *xCS*6350 is a boolean indicating whether student has taken CS 6350

*xJava* is a boolean indicating whether student has taken advanced Java skills

You are given the following dataset along with the class variable i.e. outcome variable where 1 indicates student got internship and 0 means student didn’t get it. Each data point is in the form

(*(xGPA, xWorkEx, xCS*6375*, xCS*6350*, xJava)*,outcome)

(*(*1, 1, 0, 1,1*)*,1)

(*(*0, 1, 0, 1,1*)*,0)

(*(*1, 1, 1, 1,0*)*,1)

(*(*0, 0, 0, 1,1*)*,0)

(*(*1, 1, 1, 1,1*)*,1)

* H(x) = (Xgpa ^ XworkEx ^ Xcs6350 )

1. Consider the decision tree shown below. There are two splitting attributes GPA and years of work experience.Theclasslabelsareshownbelowtheleafnodes.

Write the final hypothesis shown by this decision tree in the form of Disjunctive Normal Form (DNF)

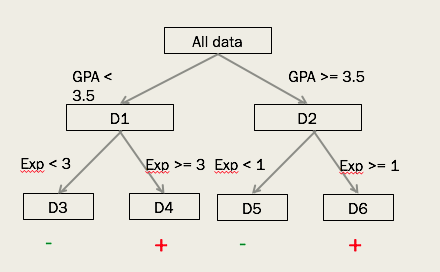


Figure 1: The labels near the leaf nodes represent class attribute i.e. outcome

* DNF = {(GPA<3.5) ^ (Exp>=3)} V {(GPA>=3.5) ^ (Exp>=1)}

1. Solvequestion2.4fromTomMitchell’sbook

a.

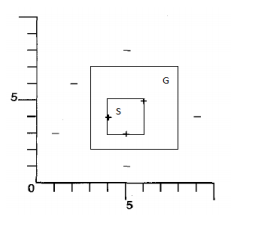
S boundary of version space can be derived from positive examples.

S boundary of this current case: **4<=X<=6, 3<=y<=5**

b.

G boundary of version space can be derived from negative examples.

G boundary of this current case: **3<=X<=8, 2<=y<=7**



C.

If the instance x, y is lies between S and G, means instance not in S set, but in G set, then it is certain that version space size will be reduced. E.g. new instance is (2,6) is negative, then version space will be reduce and G set will be **2<=X<=9, 2<=y<=8.**

**=>** x is negative in target concept, then hypothesis in G set will be specialized. And if x is positive, then set S will be generalized.

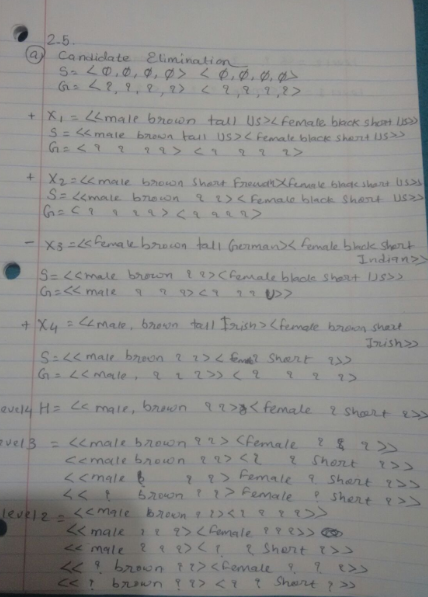
=> If x is not within target space, then version space will not be reduced.

D. The smallest number of possible training examples are 4. They are (3,2),(5,9),(2,1),(6,10). From the above points, (3,2) and (5,9) as a positive examples to learned about S boundary. Where (2,1) and (6,10)

As the negative examples to learned about G boundary.

1. Solvequestion2.5fromTomMitchell’sbook

a)





b) The only two hypothesis will be consistent with the given single positive training example.

Consistent hypotheses: <<male ? ? ? ><? ? ? ? >>

<<male ? ? ? ><Female ? ? ?>>

c)

1) <(?, brown,?, ?), (?, ?, ?, ?)>

2) <( ?, ?, tall, ?), ( ?, ?, ?, ?)>

3) <( ?,? ,? , German), ( ?,? ,? , ?)>

4) <( ?, ?, ?, ?) , (male ,? ,? , ?)>

5) <( ?,? ,? , ?), (? ,black ,? , ?)>

6) <( ?, ?, ?, ?), (? ,? ,short ,?)>

7) <( ?, ?, ?, ?), (? ,? ,? , Portuguese)>

* There are 7 queries in the sequence by which the learner will converge to the single hypothesis. The empty attributes in the above hypothesis denote that there are no more values that particular attribute can take to force S to generalize that attribute.
* The 7 queries are negative examples. The learner classifies only the initial training example as positive. The learner takes conjunction of attributes to classify the data as positive or negative. Even one wrong value will cause it to be classified as negative.
* including the positive training example( that wad fed to learner initially), we have given 8 queries. 28=256.

d)

If we were to enrich the language so that it can express every possible concepts then, instead of checking only 2 values (specific and don’t care) for each attribute in the hypothesis, we can check for every possible combination of values over all the values in the instance space. That means, sex have 2 values, hair have 3 possible values, height have 3 possible values and the nationality have 7 different values. So, there will be (2\*3\*3\*7)\*(2\*3\*3\*7)= 15,876 combinations possible.