Machine Learning - Exercise 1  
Theoretical Part

|  |  |
| --- | --- |
| Name: | Tomer Gill (תומר גיל) |
| ID: | 318459450 |
| U2 Username: | gilltom |
| Date: | 19/03/2018 |

# Part 1 - General definition of the problem

1. For
2. ((1,0,1,1),0) **CAN** exist in the training set because does equal 0. It means there are 4 literals () but the value of doesn’t matter to the result (y).
3. **Yes**, both ((0,1,1,0),1) and ((0,1,1,1),1) can exist in the training set, because 's value doesn't matter and they both fulfils the conjunction: .

# Part 2 - The Consistency Algorithm

1. **Yes**, the algorithm implements the ERM principle, because it's "learning" is done using empirical data – it uses many examples' results to try to "learn" the distribution of the outputs.
2. First, I'll prove by induction that and then I'll prove the bound is :
   1. **Base**: For , the algorithm makes mistakes.  
      **Step**: Let's assume that for an arbitrary d, the algorithm makes mistakes, and we'll prove that for , the algorithm makes :  
      We have d+1 literals (), so our initial hypothesis is the *all-negative hypothesis*: . The algorithm will go through our examples. The maximal number of mistakes that involve the literal is 2: a mistake that will remove and another mistake that will remove . In the "worst case", those will be removed from our hypothesis, and so we will be left with only d literals. From the induction's assumption we get that for d literals .  
      So in total for d+1 literals, .
   2. If the real conjunction is the all negative hypothesis, then the algorithm will never make mistakes because it is right from the start. So, for all conjunctions that are not the all-negative, we have n literals or their negations – minimum 0, maximum of d (). The first time a mistake is made, then out of the literals in the all-negative, at least d literals are removed (for each literal there is a binary value in the example, and if a mistake is made then the appropriate literal/negation is removed) to a total of d [so far 1 mistake].  
      In the worst case, the real conjunction consists of only 1 literal and each mistake will only remove one literal from our hypothesis. So the maximal mistakes that can be made in the worst case is mistakes.  
      For conclusion, we proved that maximal number of mistakes in the first mistake and then d mistakes, so .
3. In the worst case, an iteration is the first mistake – the algorithm goes over all values of the example and update the hypothesis. Every other iteration goes over maximum of d literals.  
   So in total, an iteration's time is .