**Introduction to Machine Learning Course**

Short HW2: Classification: Introduction

Submitted individually by Sunday, 07.07, at 23:59.

You may answer in Hebrew or English and write on a computer or by hand (but be clear).

Please submit a PDF file named as your ID number, e.g., 123456789.pdf.

Bonus (maximal grade is 100): Writing on a computer (using LyX/LaTeX, Word + Equation tool, etc.) = 2 pts.

## Decision trees

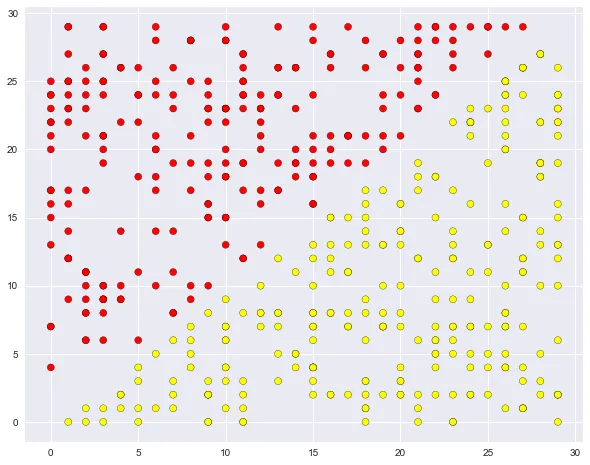
1. This question demonstrates that ID3 does not always yield an optimal tree. Consider the following dataset:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | **Feature A** | **Feature B** | **Feature C** | **Target** |
| 1 | 1 | 1 | 1 | +1 |
| 2 | 1 | 0 | 0 | +1 |
| 3 | 1 | 1 | 0 | -1 |
| 4 | 0 | 0 | 1 | -1 |

* 1. Run ID3 manually up to depth 2. In the case of a leaf with non-zero entropy predict the majority label in that leaf. In case of a tie, predict “-1”. The nodes and edges in the decision tree should be labeled appropriately. You can use your preferred graphics software to draw the tree, or you can hand draw it (neatly). Calculate the training error on the given dataset and state which tuples were misclassified.
  2. Find (manually) a decision tree of depth 2 that fits the training data perfectly.

## Decision trees strike again

1. Given before you a dataset that consists of samples with two features and a label



* 1. Can you identify a linear classifier that achieves a perfect accuracy on the dataset? You may provide a drawing to emphasize your point.
  2. Can you identify a decision tree of depth 1 that achieves a perfect accuracy on the dataset? You may provide a drawing of the decision boundary to emphasize your point.
  3. Can you identify a decision tree of depth 2 that achieves a perfect accuracy on the dataset? You may provide a drawing of the decision boundary to emphasize your point.
  4. Will your answers change if we rotate the dataset by 45 degrees clockwise? Specifically, if each in the dataset is multiplied by a 45 degrees rotation matrix , while the labels remain unchanged?

1. We now wish to prove that the information gain, i.e., , is non-negative for any set of examples and a binary attribute .

Recall our definition of and .

See the following dialogue, where we (really) asked [ChatGPT](https://chat.openai.com/) to prove this (we shortened it a little):



**Question**: could you prove that the information gain in id3 cannot be negative?

**Answer**: Yes, I can prove this using mathematical reasoning.

In the ID3 algorithm, the information gain (IG) of an attribute A with respect to a dataset D is defined as: IG(A) = Entropy(D) - Sum([|Dv| / |D| \* Entropy(Dv)])  
where Entropy(D) is the entropy of the original dataset D, |Dv| is the number of instances in the subset of D that have a specific value v for the attribute A, and Entropy(Dv) is the entropy of that subset.

Entropy is a measure of the randomness or unpredictability of the target variable, and it is calculated as: Entropy(D) = - Sum([p(c) \* log2(p(c))])  
where p(c) is the proportion of instances in D that belong to class c.

Now, let's assume that the IG for a given attribute A is negative, i.e., IG(A) < 0.

Then we have: Entropy(D) - Sum([|Dv| / |D| \* Entropy(Dv)]) < 0  
Rearranging the terms, we get: Entropy(D) < Sum([|Dv| / |D| \* Entropy(Dv)])

Since both Entropy(D) and Entropy(Dv) are non-negative, the right-hand side of the inequality is also non-negative.   
Therefore, we have: Entropy(D) >= 0.

This contradicts the assumption that IG(A) < 0, because   
Entropy(D) - Sum([|Dv| / |D| \* Entropy(Dv)]) is the value of IG(A).

Thus, we can conclude that the information gain in the ID3 algorithm is always non-negative and cannot be negative.

* 1. Attached is a quote from “ארץ נהדרת” that best describes the answer above.

Briefly explain what the problem with the above “proof” is.

* 1. Let us really prove this.

Let be an arbitrary set of examples and be a binary attribute (feature).

Assume the classes are also binary.

The entropy is thus: .

(Notice that like in the tutorial, we use with a set or with a probability interchangeably.)

**Prove:** .

You can use the following property without proving it.

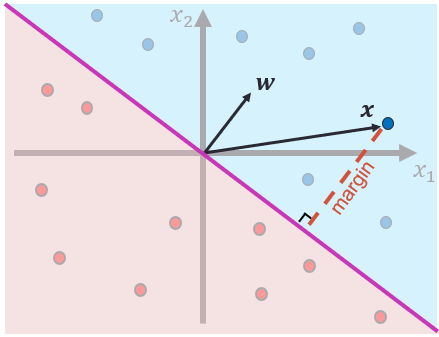
**Property:** It holds that: .

## A diagram of a graph Description automatically generatedLinear classification

In tutorial 02, we illustrated algebraic and geometric properties of linear classifiers. Specifically, we explained that a homogeneous linear classifier is parameterized by a normal vector , such that:

.

Notice how the decision rule is invariant to the scale of , and only considers its direction.



Furthermore, we showed that the geometric (signed) margin of is:

1. A nonhomogeneous linear classifier is parameterized by a vector and a bias , such that:

.

Notice how can almost be interpreted as the minimal margin required for a positive prediction (we say “almost” since is not the margin, but rather is).

* 1. In the homogeneous case, we had a degree of freedom while choosing (due to the scale invariance).  
     What similar degree of freedom do we have while choosing in the non-homogeneous case?  
     That is, how can be changed without changing the classifier itself? Briefly explain your answer.
  2. Following are 4 figures of decision boundaries induced by the same with different biases .

The biases are ordered as: .

Which bias corresponds to which figure?

Write your answer in the following format (answers are made up): A is , B is , C is , D is .

No explanations are required.

**(A) (B) (C) (D)**

A group of black and pink lines

Description automatically generatedA group of black and pink lines

Description automatically generatedA group of black and pink lines

Description automatically generatedA group of black and pink lines

Description automatically generated

## Nearest Neighbors

1. Following are 2 training sets in the feature space with 2 classes (blue/red).

Assume no dataset has two points in the exact same coordinates.

|  |  |
| --- | --- |
| **(A)** | **(B)** |
|  |  |

Following are 3 models.

1. kNN with (a training point is not considered a neighbor of itself) w.r.t.
2. kNN with (a training point is not considered a neighbor of itself) w.r.t.
3. kNN with (a training point is not considered a neighbor of itself) w.r.t.
   1. For each model above, write which datasets this model can perfectly fit (i.e., with 0 training error) and which datasets it cannot. Write your answers in a table, like in the example below. When you say a model cannot perfectly fit a certain dataset, explain why in 1-2 sentences (without drawings).

Example for a table (answers are random):

|  |  |  |
| --- | --- | --- |
| **Model / dataset** | **(A)** | **(B)** |
| **i.** | Yes | Yes |
| **ii.** | No.  For unknown reasons. | Yes. |
| **iii.** | Yes | No.  Since is positive. |

* 1. Now assume that the data points from all datasets above are rotated by the same unknown angle (around the origin). That is, each 2-dimensional data point is mapped into .

Without knowing the exact angle , answer for each of the models above:

* Might your answers for that model change?
* If not, briefly explain why.
* Otherwise, the answers for which datasets might change? Briefly explain why.

Answer for example:

* + 1. Answers unchanged because this is the best model ever.
    2. Answers unchanged since this was once revealed to me in a dream.
    3. Answers on dataset (A) might change because of this and that.