Lab Class ML:VI

If you have any questions regarding the exercises, feel free to ask them in the Moodle forum.

Until Wednesday, Jan. 15th, 2025, 11:59 pm CET, solutions to the following exercises must be submitted as one zip-file named ml23-ex4-group<your-group-number>.zip via Moodle: 1, 2a–c, 3a, 4a, 5a–b, and 6.

Exercise 1 : Decision Trees (1+1+1+1+1+1=6 Points)

(a) Name these concepts:

(a1) $\mathbf{x}|_A$

(a2) T

(a3) t

(a4) $X(t)$

(a5) $D(t)$

(a6) $\Delta\iota$

(b) Name this expression: $X = \{\mathbf{x} \in X : \mathbf{x}|_A \in B\} \cup \{\mathbf{x} \in X : \mathbf{x}|_A \notin B\}$

(c) What are the three requirements of an impurity function?

(d) What is the hypothesis space of decision trees?

(e) What is the search space of the ID3 algorithm?

(f) What is the difference between the inductive bias of the candidate elimination algorithm and that of the ID3 algorithm? Hint: search bias and restriction bias.

Exercise 2 : Decision Trees (1+1+1+0=3 Points)

Construct by hand decision trees corresponding to each of the following Boolean formulas. The examples $(\mathbf{x}, c) \in D$ consist of a feature vector $\mathbf{x}$ where each component corresponds to one of the Boolean variables $(A, B, \ldots)$ used in the formula, and each example corresponds to one interpretation (i.e. assignment of 0/1 to the Boolean variables). The target concept c is the truth value of the formula given that interpretation. Assume the set D contains examples with all possible combinations of attribute values.

Hint: It may be helpful to write out the set D for each formula as a truth table.

(a) $A \wedge \neg B$

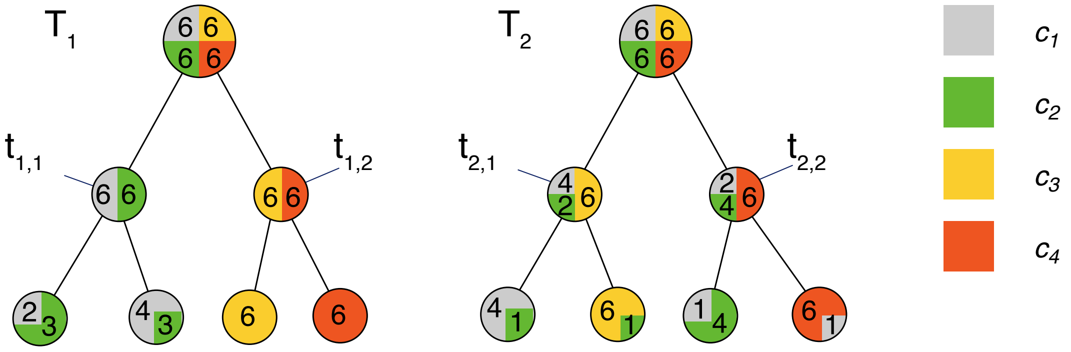
(b) $A \textit{ XOR } B$

(c) $A \vee (B \wedge C)$

(d) $(A \wedge B) \vee (C \wedge D)$

Exercise 3 : Impurity Functions (3+0+0=3 Points)

Let D be a set of examples over a feature space $\mathbf{X}$ and a set of classes $C = \{c_1, c_2, c_3, c_4\}$, with $|D| = 24$. Consider the following illustration of two possible decision trees, T_1 and T_2 – the colors represent the classes present in each subset $D(t_i)$ represented by node $t_{i,j}$ of T_i; the numbers denote how many examples of each class are present.



(a) First, consider only the first split that each of the two trees makes: compute $\Delta\iota(D, \{D(t_{1,1}), D(t_{1,2})\})$ and $\Delta\iota(D, \{D(t_{2,1}), D(t_{2,2})\})$ with (1) the misclassification rate $\iota_{misclass}$ and (2) the entropy criterion $\iota_{entropy}$ as splitting criterion.

Interpret the results: which of $\{D(t_{1,1}), D(t_{1,2})\}$ or $\{D(t_{2,1}), D(t_{2,2})\}$ is the better first split?

(b) If we compare T_1 and T_2 in terms of their misclassification rate on D, which one is the better decision tree?

(c) Assuming the splits shown are the only possibilities, which of T_1 or T_2 would the ID3 algorithm construct, and why?

Exercise 4 : Decision Trees (5+0=5 Points)

Given is the following dataset to classifiy whether a dog is dangerous or well-behaved in character:

Color	Fur	Size	Character (C)
brown	ragged	small	well-behaved
black	ragged	big	dangerous
black	smooth	big	dangerous
black	curly	small	well-behaved
white	curly	small	well-behaved
white	smooth	small	dangerous
red	ragged	big	well-behaved

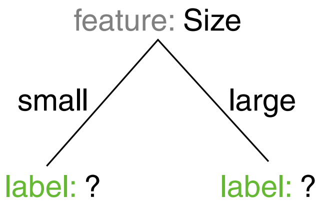
(a) Use the ID3 algorithm with $\iota_{entropy}$ as the impurity function to determine the tree T.

(b) Classify the new example (Color=black, Fur=ragged, Size=small) using T.

Exercise 5 : Cost functions (1+1+0=2 Points)

Consider the set of training examples describing mushrooms, and the simple one-level decision tree given below:

	Color	Size	Points	Edibility
1	red	small	yes	toxic
2	brown	small	no	edible
3	brown	large	yes	edible
4	green	small	no	edible
5	red	large	no	edible



(a) Determine the labels of all nodes using the cost function $\mathit{cost}(c', c)$ (cf. ML:VI-36):

$$\mathit{cost}(c', c) = \begin{cases} 1 & \text{if } c' \neq c, c \in C \\ 0 & \text{otherwise} \end{cases}$$

(b) Devise a new cost function that ensures that, for the same tree structure, none of the poisonous mushrooms in the training set are classified as edible.

(c) Compute the misclassification costs of the tree for both cost functions.

Exercise 6 : P Classification with CART Decision Trees (1+1+1+1+1+1+1+1=8 Points)

In this exercise, you will implement the CART algorithm for constructing decision trees for predicting whether a given text was written by a human or generated by a language model. Submit the file with your predictions for the test set along with your other solutions.

Download and use these files from Moodle (the `tsv` files are the same as in the last sheet):

- `features-train.tsv`: Feature vectors for each example in the training set.
- `features-test.tsv`: Feature vectors for each example in the test set.
- `labels-train.tsv`: Quality scores for each example in the training set.
- `programming_exercise_decision_trees.py`: Template for the programming exercise. It contains function stubs for each function mentioned below, as well as functions implemented in the previous exercises. Use the following command to run the program:

  ```
  python3 programming_exercise_decision_trees.py
  ```

  Note: The program will read the above-mentioned `tsv` files from the `data` folder that should be in the same directory as the program.
- `requirements.txt`: Requirements file for the template; can be used to install dependencies.

(a) Implement a function `most_common_class` to find the most common class in the dataset.

(b) Implement a function `gini_impurity` that computes the Gini index for the given set of example classes C (slide ML:VI-79).

(c) Implement a function `gini_impurity_reduction` that computes the Gini impurity reduction of a binary split (slide ML:VI-50).

(d) Implement a function `possible_thresholds` that returns all possible thresholds for splitting the example set X along the given feature. Pick thresholds as the mid-point between all pairs of distinct, consecutive values in ascending order.

(e) Implement a function `find_best_split` that finds the best split based on the Gini impurity reduction for the given set of examples X and the given set of classes C.

(f) Implement the `id3_cart` function to construct a CART decision tree with the modified ID3 algorithm (slides ML:VI-109, ML:VI-22). The function should return the root node of the tree.

(g) Implement a function `train_and_predict`. This function should train the model on the training set and return the predictions for the test set. What is the misclassification rate on the training set?

(h) Run the `plot_misclassification_rates` function to plot the misclassification rate on the training set for different depths of the decision tree. What phenomenon do you observe in the plot with increasing depth and why does it happen?

If you would like to improve your model, here are some hints of what you could try:

- Implement a stopping criterion from slide ML:VI-125 to avoid overfitting.
- Implement the pruning algorithm from slide ML:VI-126 to avoid overfitting.