BAUHAUS UNIVERSITÄT WEIMAR

INTRODUCTION TO MACHINE LEARNING

SUBMITTED BY

Anjana Muraleedharan Nair (125512)

Isabel Maria Binu (125514)

Vishal Sanjay Shivam (125353)

Sharat Anand (125404)

Exercise 1: Machine Learning (general)
(a) Define the terms “supervised learning”, “unsupervised learning”, and “reinforcement learning”.

Supervised learning - Supervised learning is a type of machine learning where an algorithm learns from labelled training data. It is a function from a set of input-output-pairs. The algorithm will be provided with a dataset where each example is paired with a corresponding target or output.
Examples: Classification (assigning data points to predefined categories).
Regression (predicting a numerical value).
Optical character recognition.

Unsupervised learning - Unsupervised learning is a type of machine learning where an algorithm is given unlabeled data and tasked with finding hidden patterns or structure within the data. Unlike supervised learning, there are no target outputs provided.
Examples : Clustering.
Dimensionality reduction.
Density estimation.
Intrusion detection in a network data stream.

Reinforcement learning - Reinforcement learning is a type of machine learning that focuses on training agents to make a sequence of decisions in an environment to maximize a cumulative reward. Learn, adapt, or optimize a behavior strategy in order to maximize the own benefit by interpreting feedback that is provided by the environment.
Examples : Development of behavior strategies in a hostile environment.
Robotics, game playing, and autonomous systems.

(b) Determine the learning paradigm (supervised, unsupervised, reinforcement) for the following tasks:
(b1) Sentiment analysis (determine if a text has positive or negative sentiment) - Supervised learning.

(b2) Data compression - Unsupervised learning.

(b3) Self-driving cars - Reinforcement learning.

(b4) Personalized content recommendation - Supervised Learning.

(b5) Spam filtering - Supervised Learning.

(b6) Sorting fruits in a basket by type - Supervised Learning.

Exercise 2 : Specification of Learning Tasks

(a) A pile of Mushrooms. - O

(b) A table with the columns “size”, “weight”, and “color”, as well as one row for each mushroom, and the respective measurements in the cells. - X

(c) A human mushroom expert who can tell whether any mushroom you show them is poisonous or edible. -$\gamma(o)$

(d) A device that measures the size, weight, and color of a mushroom. - $\alpha(o)$

(e) The set {Poisonous, Edible} -C

(f) The machine learning system that you are trying to build. - y(x)

Exercise 3: Data Annotation and Feature Engineering (3+1+0=4 Points)

(b) Look back to the exercise 2. Which symbol in the picture corresponds to the role that you are playing? Which symbol corresponds to the functions that you implement in this exercise?

$\gamma(o)$ is the symbol used. This is the ideal target function which classifies the object from the real world

Exercise 5: Rule-Based Learning (Practise)

	Weekday	Mother-in-the-car	Mood	Time of day	run-a-red-light
1	Monday	no	easygoing	evening	yes
2	Monday	no	annoyed	evening	no
3	Saturday	yes	easygoing	lunchtime	no
4	Monday	no	easygoing	morning	yes

(a) Apply the Find-S algorithm for the example sequence 1, 2, 3, 4.

$h_0 = (\perp, \perp, \perp, \perp)$

x_1 = (Monday, no, easygoing, evening) h_1 = (Monday, no, easygoing, evening)

x_2 and x_3 are ignored – since it is negative. h_2 = (Monday, no, easygoing, evening) h_3 = (Monday, no, easygoing, evening)

x_4 = (Monday, no, easygoing, morning) h_4 = (Monday, no, easygoing, ?)

(b) Apply the Candidate-Elimination algorithm for the example sequence 1, 2, 3, 4, and identify the boundary sets H_S and H_G.

$S_0 = (\perp, \perp, \perp, \perp)$

S_1 = (Monday, no, easygoing, evening) = S_2=S_3

S_4 = (Monday, no, easygoing, ?)

(Monday, no, easygoing, ?)

G_4 = (Monday, ?, easygoing, ?) (?, no, easygoing, ?)

G_3 = (Monday, ?, easygoing, ?) (?, no, easygoing, ?) (?, ?, easygoing, evening)

G_2 = (?, ?, easygoing, ?)

G_0 = (?, ?, ?, ?) = G_1

(c) What is the version space HD for this example?

H_D = (Monday, no, easygoing, ?)

Exercise 6: Rule-Based Learning (Background)

(a) Can a version space H_D contain hypotheses that are neither in the set H_S nor in the set H_G? If so, how?
Yes. A version space H_D contains hypotheses that are neither in the set H_S nor in the set H_G. The version space H_D contains all the hypotheses from the hypothesis space that are consistent. When there is uncertainty and ambiguity in the training data, hypotheses tend to shift data from specific to general between H_S and H_G in rule based learning.

(b) For any two hypotheses y1(), y2(), $y1() \neq y2()$, from the set H_S of a version space H_D holds (check all that apply):

- ○ $(y2() \geq_g y1()) \vee (y1() \geq_g y2())$
- ○ $(y2() \not\geq_g y1()) \vee (y1() \geq_g y2())$

(c) Which of the two algorithms Find-S and Candidate-Elimination has a stronger inductive bias? Explain your answer.

Inductive bias refers to a set of assumptions or preferences that a learning algorithm uses when generalising the training data to make predictions on unknown data. The find-S Algorithm has a stronger inductive bias.
Find-S Algorithm generalises the hypothesis as it begins with the most specific hypothesis and revises it when it discovers a positive case. Candidate Elimination Algorithm has a weaker inductive bias as it shifts from most specific (Hs) to general (HG) hypotheses for positive values and goes from General to specific hypotheses for negative values in version space.