An introduction to Supervised Machine Learning

Mohamed Siala
https://siala.github.io/

INSA-Toulouse & LAAS-CNRS

March 4, 2022

Context

- An exponential increase of real-life applications of machine learning
- This is an introduction course. Next year you will follow more advanced courses (depending on your orientation)
- You need some basic knowledge regarding linear algebra, algorithms and complexity
- I ask questions very often, so please be interactive
- Feel free to stop me anytime. There is no stupid question!
- The evaluation will be decided later. Most likely it will be based on lecture questions and practical sessions
- The course is articulated around three parts: introduction, interpretable machine learning (myself), and neural networks (Arthur Bit Monnot)

References

T. Hastie, R. Tibshirani, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition.*
Springer Series in Statistics, Springer, 2009.

S. J. Russell and P. Norvig, *Artificial Intelligence - A Modern Approach, Third International Edition.*
Pearson Education, 2010.

Part 1: Introduction

Chapter 1: Context

Figure 1: How to teach a child animal recognition? [1]

[1] Image from https://en.wikipedia.org/wiki/Global_biodiversity

Figure 2: How to predict a player's performance? [2]

[2] Image from `https://en.wikipedia.org/wiki`

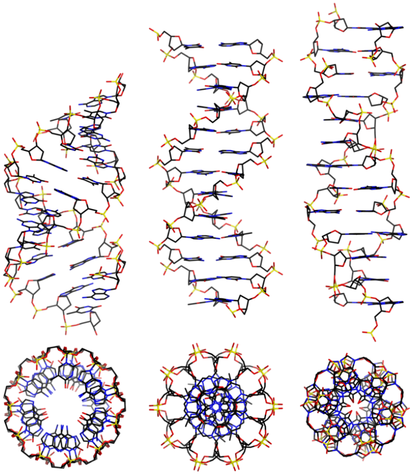


Figure 3: Analysis of evolutionary biology based on DNA patterns [3]

[3] Image from https://en.wikipedia.org/wiki/DNA

Figure 4: How to bike? [4]

[4] Image from https://en.wikipedia.org/wiki/Cycling

- Animal recognition: It does not make sense to show the picture of every animal!

  $\Longrightarrow$ Show some pictures per animal and let the child learn
- Player performance: No straightforward formulae. Keep track of its latest performances and predict accordingly
- DNA clustering: Let the machine discover by itself the different patterns.
- Cycling: It needs a sequence of successful/unsuccessful trials!
- That's pretty much the philosophy of machine learning: feed the computer some date, and let it learn by itself
- Note 1: Some computational problems are simply not solvable in a traditionally way.
- Note 2: But also machine learning is not always the solution! Consider for instance basic arithmetic operations.

Examples of Machine Learning Applications [1]

- Autonomous cars
- Flying drones
- Face recognition
- Computer vision
- Natural language processing
- Music/movie recommendation
- Dating apps (tinder),
- Friends recommendation
- Weather prediction
- Trading
- ...

The Big Picture

It depends on the problem at hand and on the nature of data!

1. **Labelled data**:
   - The task of animal recognition is to predict the animal in a given picture. The data (used to "train" the computer) is a set of pictures and each picture is associated to an animal. In this case, the **label** is a **category**
   - In trading, the task is to predict the price evolution of a given share and the data is simply historical evolution of the share. In this case the data is **labelled** with **float numbers**.
2. **Unlabelled data**: For instance, when using clustering to evaluate the density of a population. The data can simply be a set of coordinates with no labels.
3. **Continuously updated data**: The data is continuously updated according to previous experiences: For instance, a robot that tries to ride a bicycle learns how to bike by a sequence of trials

Supervised/Unsupervised/Reinforcement Learning

- Supervised Learning (Labelled data): Predict a function that associates inputs to outputs based on historical data
  - Categorical labels (discrete values): Classification
  - Non-categorical labels (float numbers): Regression
- Unsupervised Learning: The task is to figure out patterns presented in the data (unlabelled data)
- Reinforcement learning: learning from a series of rewards /punishments
- But also, depending on the problem, data could be both labelled/non labelled, etc.. (semi-supervised learning)

Figure 5: How to teach a child animal recognition? [5]

Classification task

[5] Image from https://en.wikipedia.org/wiki/Global_biodiversity

Figure 6: How to predict a player's performance? [6]

Regression task

[6] Image from https://en.wikipedia.org/wiki

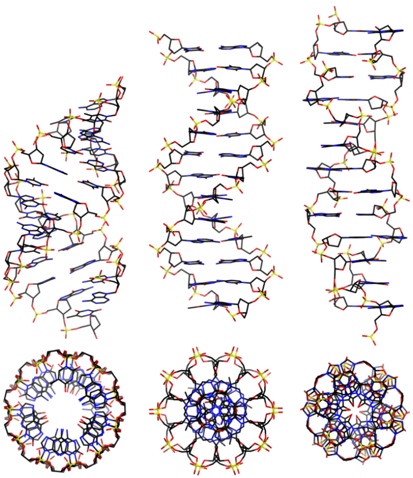


Figure 7: Analysis of evolutionary biology based on DNA patterns [7]

Unsupervised learning (clustering) task

[7] Image from `https://en.wikipedia.org/wiki/DNA`

Figure 8: How to learn cycling? [8]

Reinforcement learning

[8] Image from https://en.wikipedia.org/wiki/Cycling

Chapter 2: Supervised Machine Learning

Supervised Machine Learning

- We focus in this course on Supervised ML
- Examples of applications
  - Tumor detection: The data is a collection of brain scans. Each scan is associated with a label indicating the type of tumor
    $\Longrightarrow$ Classification
  - Credit score: The data is a collection of clients profiles (age, salary, genre (?), job, ...) with a positive or negative feedback
    $\Longrightarrow$ Binary classification
  - Precipitation prediction: (loosely speaking) the data is a collection of sequential weather conditions and the purpose is to predict the Precipitation chance (float value)
    $\Longrightarrow$ Regression

Problem Definition [2]

- Given a historical data (**training set**) in the form of input-output examples: $\{(x_1, y_1), \ldots (x_n, y_n)\}$ where x_i is an input, y_i is the output of x_i drawn from an unknown function f
- Find a function f_h (called a hypothesis or model) that approximates the true function f
- The approximation criterion can be defined in different ways. We can consider it as a function minimizing some error.

Training Phase

- The function f_h is constructed via a **training algorithm**
- The training algorithm depends on the hypothesis space
- Examples of hypothesis space (family of functions) include polynomial functions, trigonometry functions, decision trees, decision lists, neural networks, ...

Testing Phase

- The evaluation of the constructed hypothesis (or model) is done via a set of unseen examples called **testing set**
- The testing set is usually taken randomly from the initial data. However, it shouldn't be part of the training set
- The common way is to split the initial data into a training and testing sets randomly

Model evaluation: Classification

- Training accuracy: percentage of training examples that are well predicted
- Testing accuracy: percentage of testing examples that are well predicted
- The concept of **generalisation** is precisely the quality of testing accuracy

Classification Evaluation: The Confusion Matrix

- Accuracy alone hinders the way predictions are made. Model evaluation needs more refinement
- Consider binary classification (positive/negative classes) with 80% testing accuracy
- 80% seems good, however, a deeper investigation shows that most of negative examples are wrongly predicted! This shows a biased in the model (biased towards positive examples)
- The purpose of the confusion matrix precisely that: to have a refined evaluation of the model's predictions
- In the case on m classes, the matrix is of dimension $m \times m$ where $M[i][j]$ is the number of examples of class i that are predicted to be as the class b
- The model bias can be easily seen: For instance, one can answer statistical queries such as: is the error evenly distributed? to which class the model is likely to predict? ...

The Confusion Matrix in the Case of Binary Classification

- A positive class and a negative class
- The confusion matrix is of dimension 2×2
- True Positive Rate (TP rate): the likelihood that a positive example is well classified
- False Positive Rate (FP rate): the likelihood that a positive example is wrongly classified
- True Negative Rate (TN rate): the likelihood that a negative example is well classified
- False Negative Rate (FN rate): the likelihood that a negative example is wrongly classified

The Covid Example

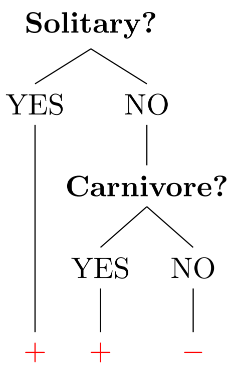
- Assume that we have a prediction model with 85% accuracy
- 100 individuals: 70 positives and 30 negatives
- The confusion matrix:

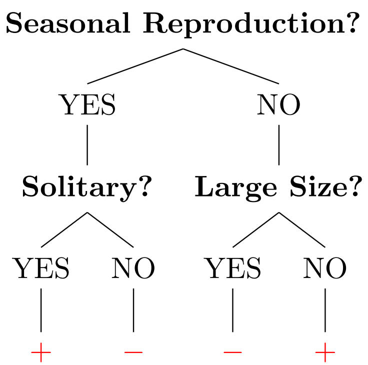
	Positive	Negative
Positive	65	5
Negative	10	20

- TP rate is $65/70 = 92,85\%$; FP rate is $5/70 = 7,15\%$
- TN rate is $20/30 = 66,66\%$; FN rate is $10/30 = 33,33\%$
- A positive individual that tested positive, the test is $92,85\%$ accurate
- Is an individual is negative and tested positive, the test is $33,33\%$ true

Toy Example: DTs to Predict The Likelihood of Animal Extinction

Big Size	Carnivore	Seasonal Reproduction	Solitary	Extinct
0	1	0	1	yes
1	0	0	1	yes
0	0	0	1	no
1	1	1	0	no
0	0	1	0	yes





- Tabular data
- Hypothesis space: Decision trees
- Left tree: accuracy 2/5
- Right tree: accuracy 2/5

Ockham's Razor Principle

- In the previous example, we have two different trees with the same accuracy
- Which tree to choose?
- Hard to answer without specific requirements
- Ockham's Razor Principle[9]: pick the simplest!
- Simplicity is also hard to define
- In decision trees, simplicity could be the depth, the number of features, a combination of both, etc
- When using polynomials (as a hypothesis space), lower degrees seems to be simpler
- In other cases it is very hard to define simplicity

[9] Philosopher `https://en.wikipedia.org/wiki/William_of_Ockham`

Training Algorithms Evaluation

- Suppose that we have a number of training algorithms for a given hypothesis space
- How to choose the best?
- Generalisation should be evaluated a number of random splits
- A common way is to use the $k-$fold cross validation:
  1. Split the data into k folds
  2. Perform the training k times. At each iteration, a different fold is chosed as a testing set and the rest is used for training.

Questions & Exercises

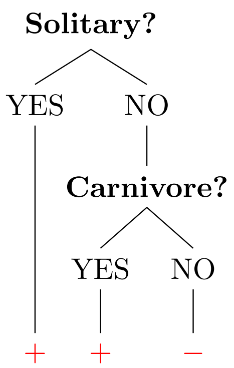
Out of these three questions, which one is the hardest and which one is the easiest (computationally)?

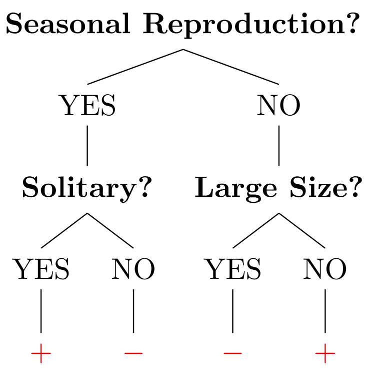
1. Building a perfect decision tree 100% accuracy ?
2. Building the best decision tree within a height h ?
3. Find a perfect decision tree with a minimum height ?

Question 3 is the hardest (we need a proof) and question 1 is the easiest (we can keep splitting until a perfect classification)

Questions & Exercises

Big Size	Carnivore	Seasonal Reproduction	Solitary	Extinct
0	1	0	1	yes
1	0	0	1	yes
0	0	0	1	no
1	1	1	0	no
0	0	1	0	yes





- Find a perfect tree
- Find a perfect tree with a minimum weight
- Find a best tree with height 2