AdaBoost
Analysis & Optimization

FCUP

Machine Learning I

2º. Semester 2023/24

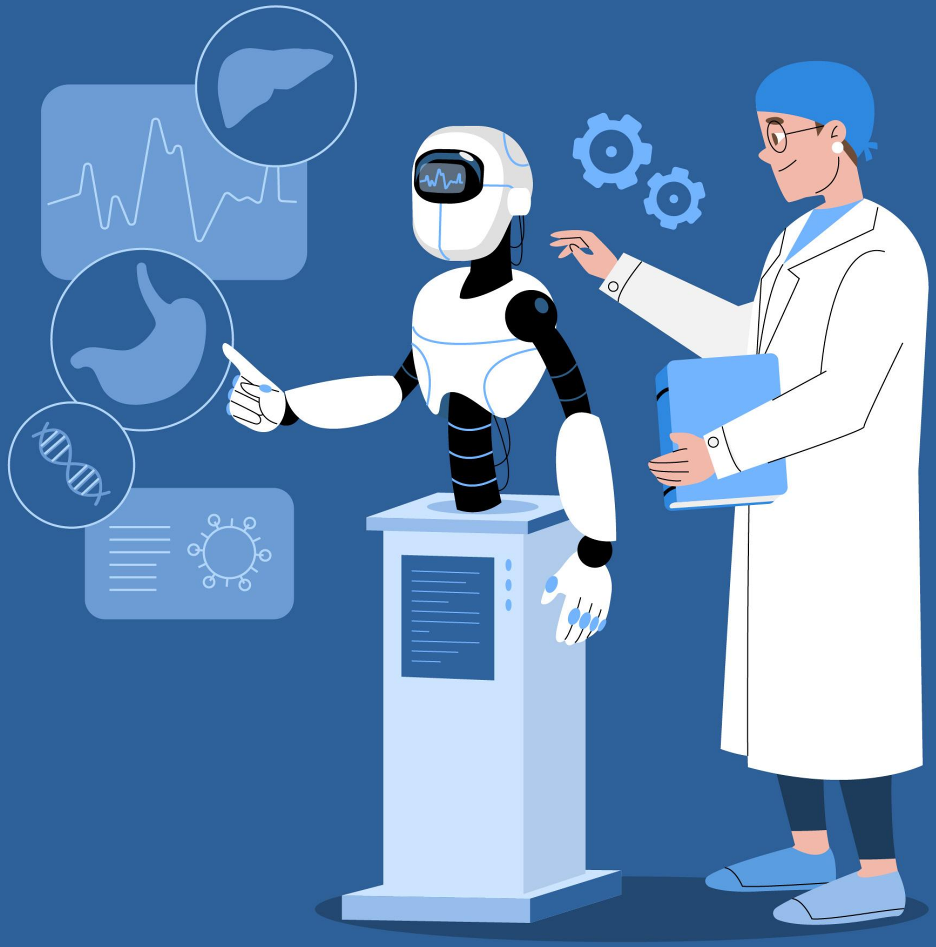
Gonçalo Esteves (up202203947)

Nuno Gomes (up202206195)

Lesson Plan

Course	Machine Learning I
Topic	AdaBoost: Analysis & Optimization
Destination	Machine Learning Enthusiasts
Methodology	*PowerPoint* Presentation

Estimated Time: 10-15 minutes



Introduction

In recent years, the rise of **Artificial Intelligence** and the widespread use of **Machine Learning** have revolutionized the way we tackle complex real-world challenges. However, due to the **diverse nature of data involved**, choosing the right algorithm is crucial to achieve **efficient and effective solutions**. Therefore, understanding the strengths and weaknesses behind different Machine Learning algorithms, and **knowing how to adapt them** to meet specific challenges, can become a fulcral skill to develop.

Introduction | Project Overview

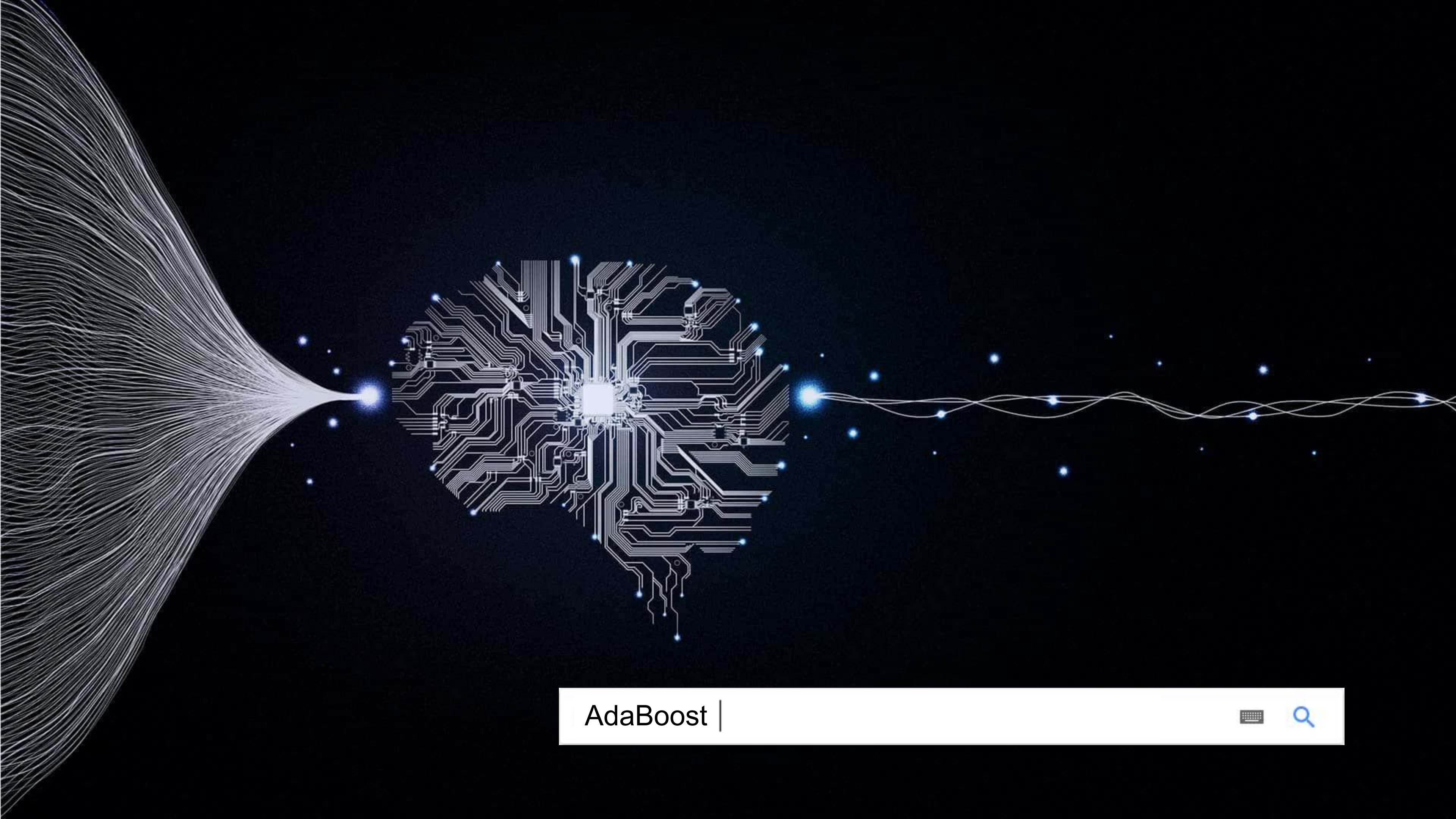
Furthermore, since the choice of algorithm greatly depends on the **specific task** and **data involved**, it's clear that **there is no "Master Algorithm"** (No algorithm can solve every problem).This **Project** focuses on the following topic:

With no Master Algorithm, is it possible to improve a existing Machine Learning Algorithm in characteristics it struggles the most?

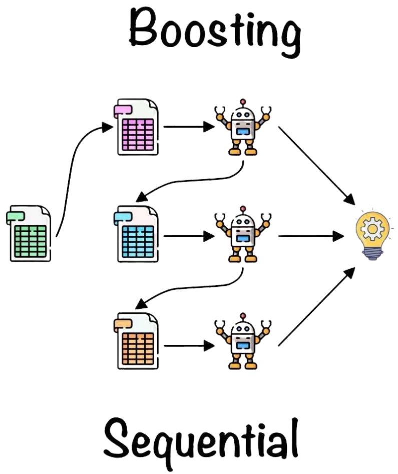
Introduction | Project Overview

Therefore, after choosing a **Machine Learning algorithm** and gaining a thorough understanding of its theoretical and empirical aspects, we aim to **refine it**, specifically targeting its weaknesses in solving **classification problems**.

Classification Problems


AdaBoost

AdaBoost | Description



AdaBoost (Adaptive Boosting) is a type of ensemble learning technique. It consists on:

→ Train a **series of weak classifiers** on the dataset - **increases the focus** on data points that were previously **predicted incorrectly**.

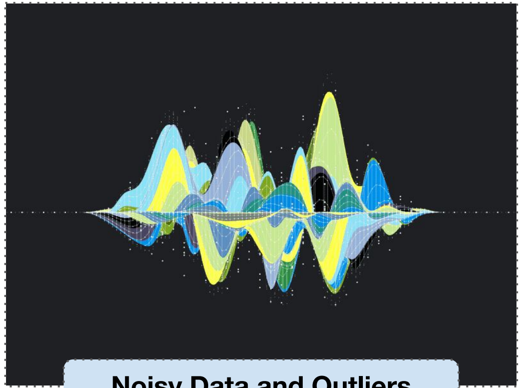
→ Weights all the individual weak classifiers **based on their performance** (classifiers with **higher predictive accuracy contribute more to the final decision**).

AdaBoost | Limitations

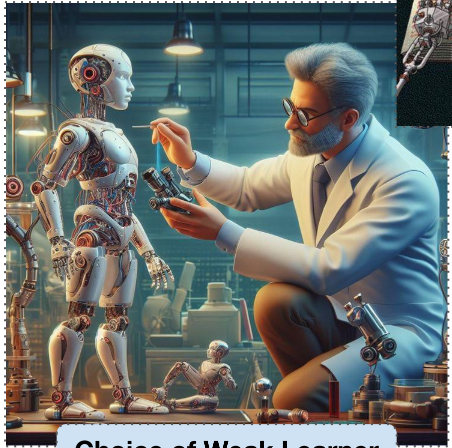
Despite the Algorithm’s powerful techniques, it yet remains limited by a few factors:

→ **Noisy Data and Outliers**

→ **Choice of Weak Learner**

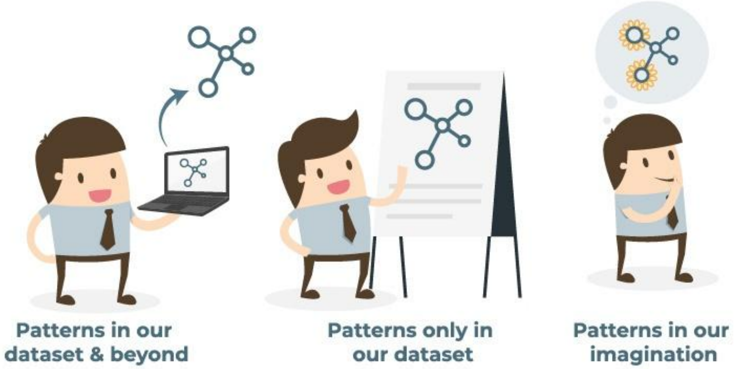


Noisy Data and Outliers



Choice of Weak Learner

AdaBoost | Limitations | Noisy Data and Outliers



Noisy Data

→ it may focus too much on **correcting previous misclassifications**, leading to overfitting.

Outliers

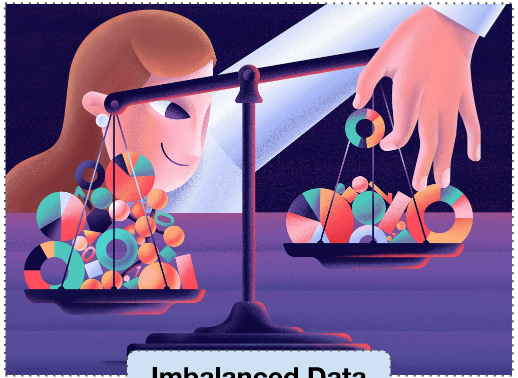
→ Impact performance, as AdaBoost tries to fit each example perfectly, potentially **overlooking actual patterns** in the data.

AdaBoost | Limitations | Choice of Weak Learner

Furthermore, AdaBoost can be quite **sensitive** to the Weak Learners used. The best weak models are often described as **simple**, **fast** and with **low variance** and **high bias**. Therefore, they do not memorize the training data but rather **learn to generalize** different aspects within the data.


Proposal

AdaBoost | Proposal



Imbalanced Data

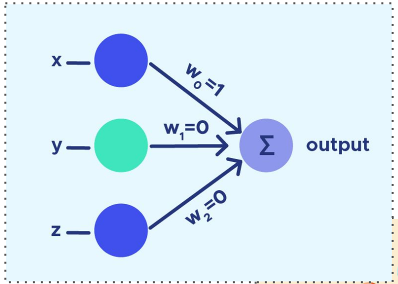
We addressed AdaBoost's performance issues with **imbalanced data**. Due to its iterative nature, AdaBoost can overlook the **minority class**, including noise and outliers, leading to overfitting. We investigated the **impact of weak learners** on its effectiveness in **binary classification**, aiming to find modifications that improve handling class imbalance and enhance overall **accuracy and robustness**.

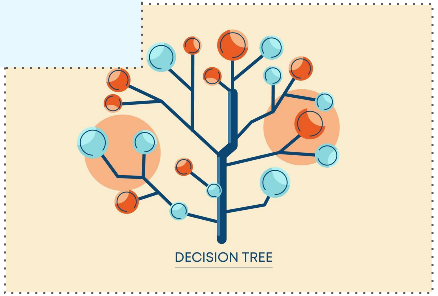
AdaBoost | Proposal | Choice of Weak Learners

Weak learners should not only be capable of achieving performance levels that are **slightly better than random guessing** but also remain simple enough so that they **do not require high computational costs**. Therefore, we propose using:

→ **Decision Trees with Increased Depth**

→ **Perceptrons**





AdaBoost | Proposal | Choice of Weak Learners

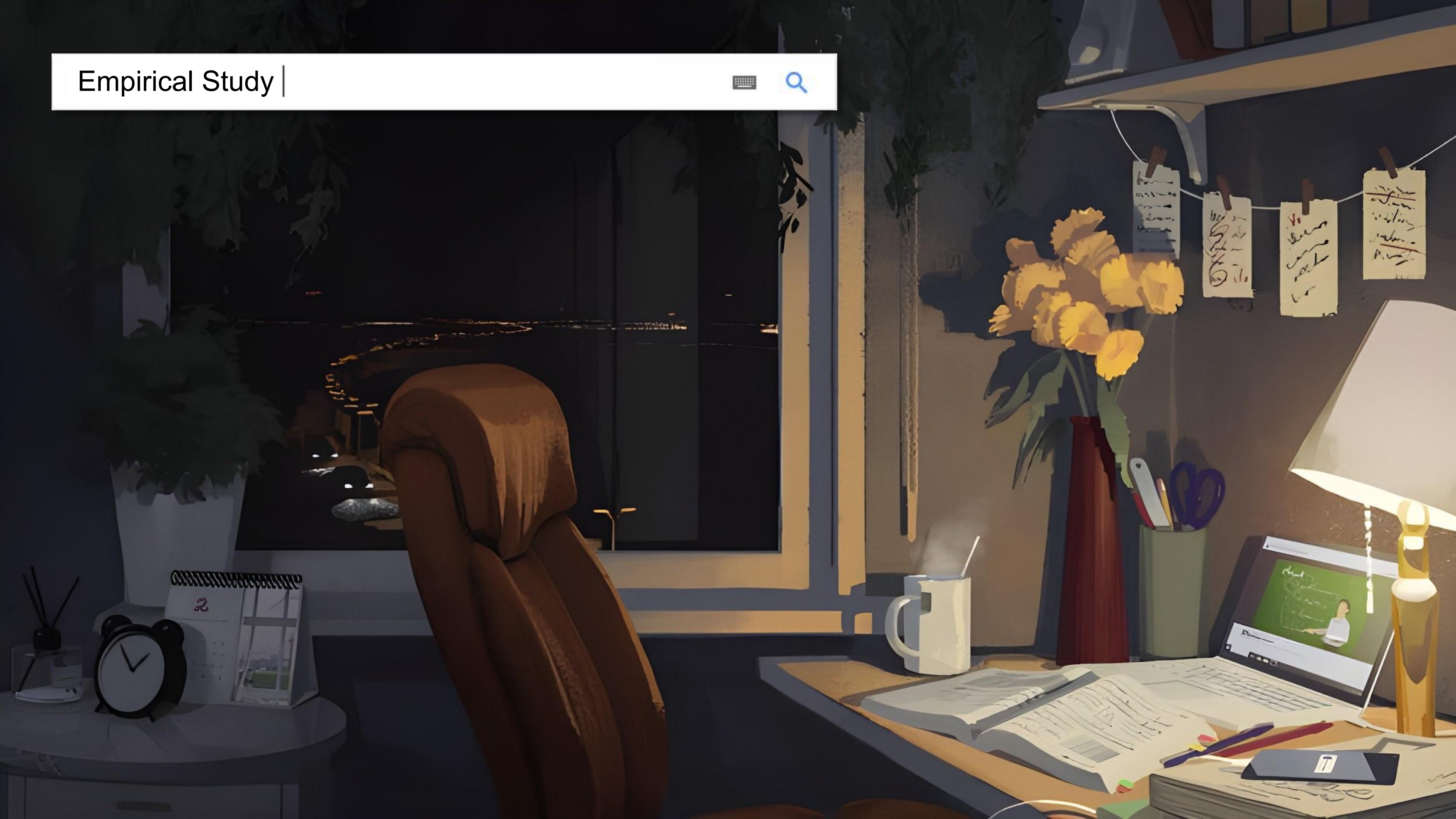
Decision Trees with Increased Depth

→ In AdaBoost, **decision stumps** are commonly used since they are less prone to overfit. However, **slightly deeper trees** (Max Depth of 3) might be more beneficial to **capture more complex patterns** without causing significant overfitting.

Perceptrons

→ Perceptrons are **simple linear classifiers**, thereby for datasets with **linearly separable data**, they lead to **faster training times** compared to decision trees.

→ Its simplicity can be **advantageous in imbalanced datasets** where complex models might **overfit by learning too much from the minority class** examples (**Noise and Outliers**).


Empirical Study

We selected the datasets inside the **OpenML-CC18 Curated Classification** that focus on **Binary Classification Problems**.

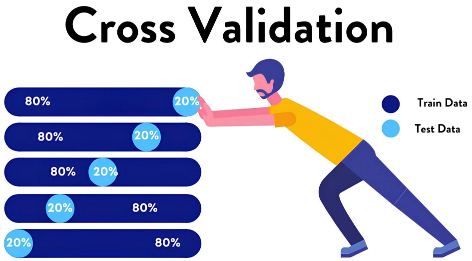
The Algorithms are mainly affected by **2 Hyperparameters**:

→ **Number of Boosting Rounds** (M)

→ **Weak Learners Used** and the respective **hyperparameters**

To Evaluate the models, we performed:

→ **K-Fold Cross Validation** (5 Folds).



AdaBoost | Empirical Study | Results

Results show that using **deeper Decision Trees** in AdaBoost performs slightly better than the original version, but **not significantly** so. However, using **Perceptrons** as weak learners results in slightly **worse performance** compared to the standard AdaBoost.

Dataset	Positive Class (%)	Negative Class (%)	Majority Class (%)	AdaBoost [Base] (%)	AdaBoost [DT - MaxDepth 3] (%)	AdaBoost [Perceptron] (%)
kr-vs-kp	47.78	52.22	52.22	94.09	98.72	96.62
breast-w	34.99	65.01	65.01	96.34	96.78	96.34
credit-approval	54.67	45.33	54.67	87.29	87.59	68.60
credit-g	30.00	70.00	70.00	75.80	76.90	70.00
diabetes	34.90	65.10	65.10	76.96	75.91	66.41

Sample of the Results

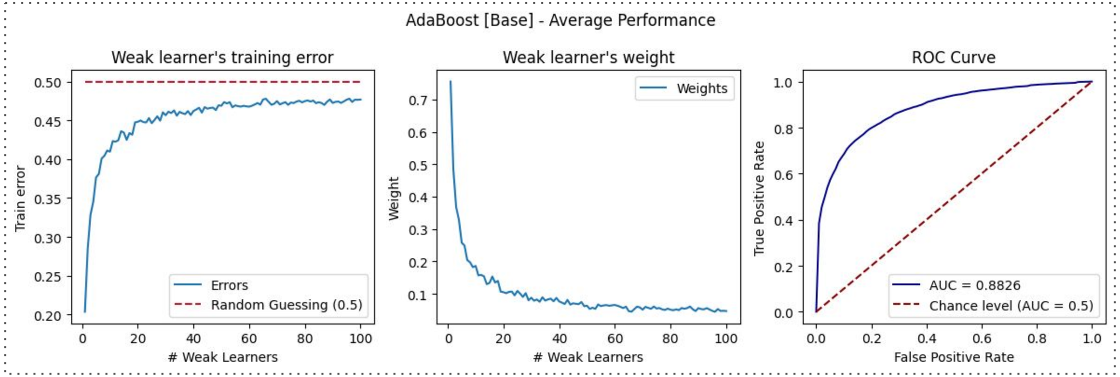
Average Results

AdaBoost [Base] (%)	AdaBoost [DT - MaxDepth 3] (%)	AdaBoost [Perceptron] (%)
83.978529	85.828529	80.553235

AdaBoost | Empirical Study | Results

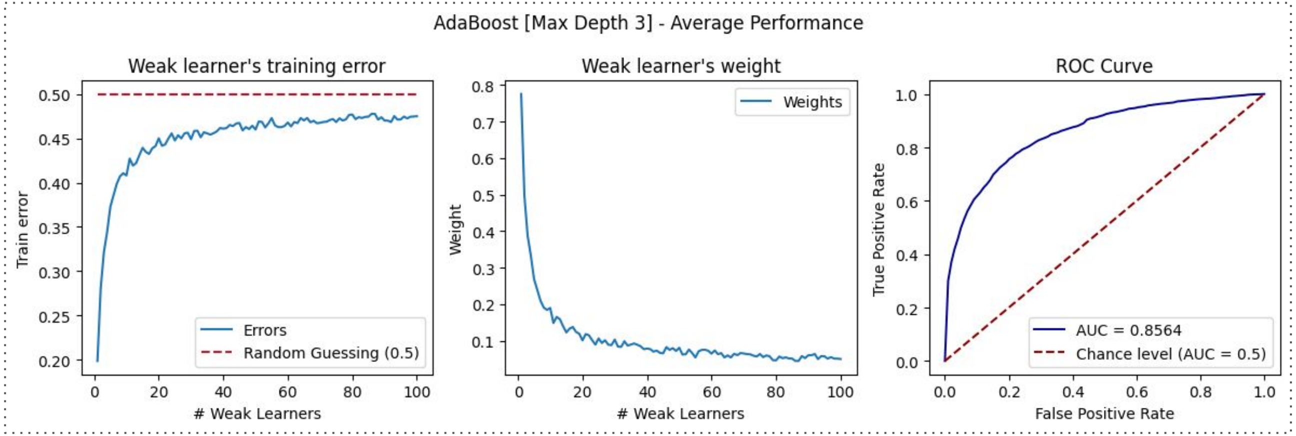
For instance, let's analyse how they **behave** against a given dataset → **Breast-W Dataset**

As we train **new weak learners** with new subsets of data in each **boosting round**, we observe their **influence** in the final decision **decrease** while their **training error increases**. This trend occurs because the initial weak learners process more **"new" data** and are therefore better positioned to **identify effective classification boundaries**, thus having a **greater impact on the final prediction**.



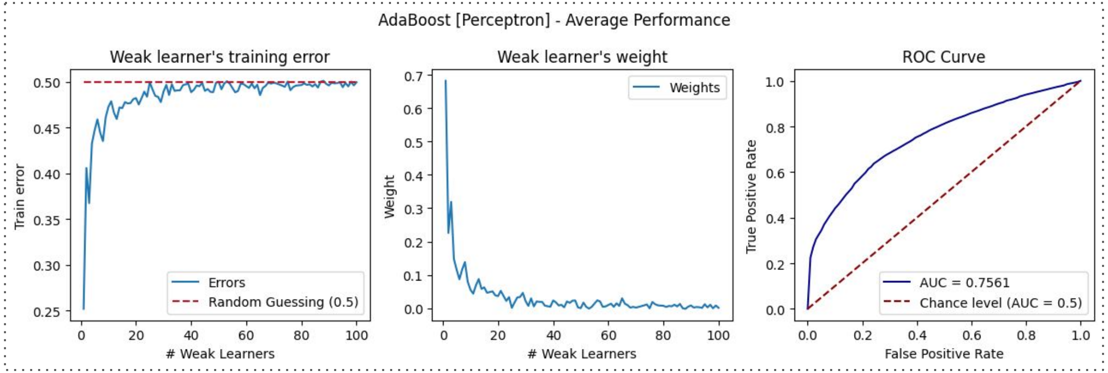
AdaBoost | Empirical Study | Results

Similarly to the results obtained with the **base algorithm**, the AdaBoost with **increased depth** on the weak learners seems to correctly classify most cases which corroborates with the **good performance** shown in the ROC Curve diagram. In addition, the algorithm **evolves similarly** to the original version throughout each **boosting round** which makes it converge to a good solution.



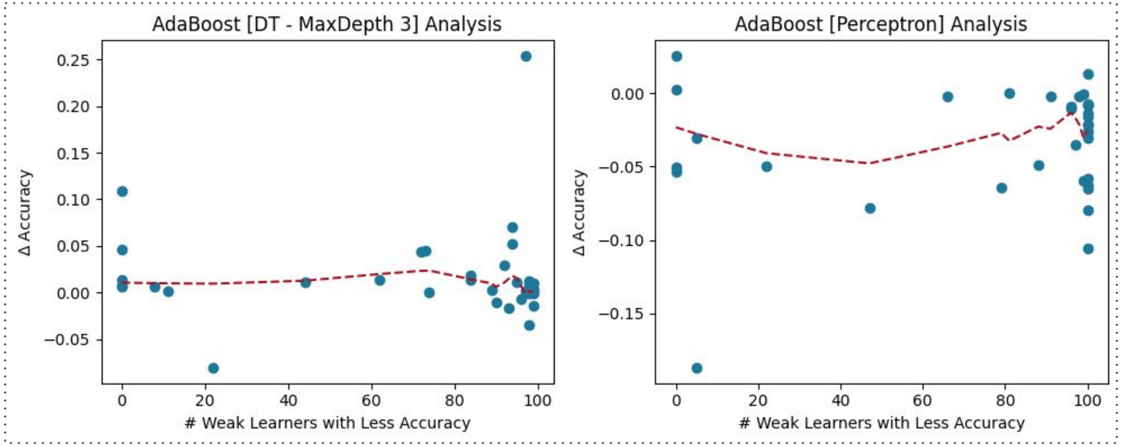
AdaBoost | Empirical Study | Results

The AdaBoost model using **Perceptrons** shows promising results, but with some performance aspects resemble **random guessing**. Error rates stabilize near the random guessing threshold over 100 rounds, indicating **minimal learning progress**. This suggests Perceptrons may not fully utilize AdaBoost's strategy of refining decision boundaries through targeted training. Additionally, their influence on the ensemble's decisions quickly **decreases**, indicating early learners may not provide a **solid foundation** for subsequent ones.



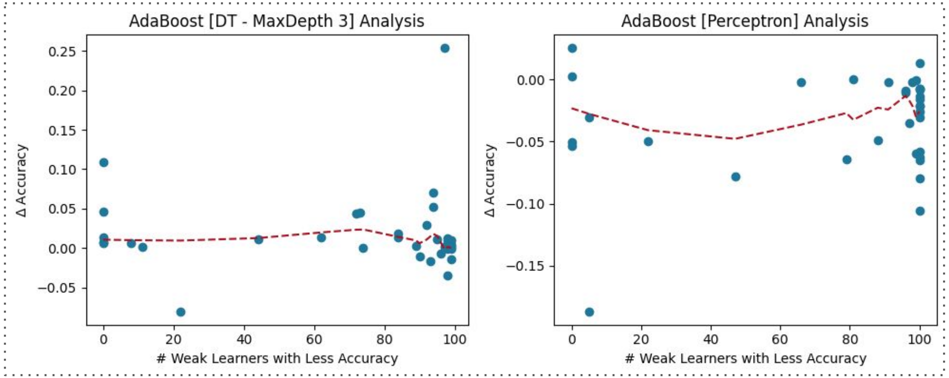
AdaBoost | Empirical Study | Results

Our following analysis will aim to determine **how many weak learners perform at or below random guessing** level and assess how it **affects the model's overall accuracy**.



AdaBoost | Empirical Study | Results

The **Δ Accuracy** for AdaBoost with **decision trees of Max Depth 3** fluctuates around zero with occasional improvements, indicating it **performs similarly** to the base model. Increasing tree depth doesn't offer a clear advantage over simpler stumps. In contrast, AdaBoost with **Perceptrons** shows consistently **negative Δ Accuracy**, failing to handle non-linear data effectively, highlighting Perceptrons' **limitations** in capturing complex relationships.



Statistical Inference

AdaBoost | Statistical Inference | Friedman Test

To assess whether there are **significant differences** in the results, we chose to conduct a **Friedman Test** since it is especially effective for comparing the performance of different models across **multiple datasets**. Therefore, it focuses in determining whether any model outperforms the others or if the **differences in their performances are statistically significant**.

AdaBoost | Statistical Inference | Hypothesis Formulation

→ **Null Hypothesis (H0)** : All the Models have the same performance and therefore there are **no significant differences between them**.

$$\mu_1 = \mu_2 = \mu_3$$

→ **Alternative Hypothesis (H1)** : At least one of the models performs differently compared to **at least one** of the other ones.

$$\forall j \ \exists i, i, j \in \{1, 2, 3\} \land (i \neq j) : \mu_j \neq \mu_i$$

→ If ***P-Value* > α**, **H0 is not rejected**

→ If ***P-Value* ≤ α**, **H0 is rejected**

For a **significance level of 0.05** ($\alpha = 0.05$):

→ ***P-Value*** ≈ 1.048×10^{-9}

Therefore we **reject the Null Hypothesis (H0)** and consequently conclude that **there are significant differences** between the studied models.

AdaBoost | Statistical Inference | Post-Hoc Test

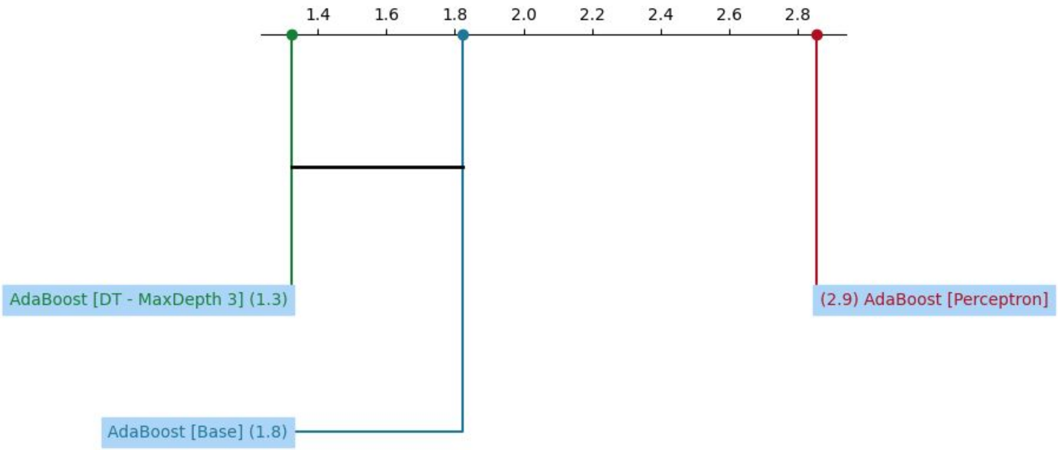
However, since the FriedMan Test does not explicitly convey **what specific models** are significantly different from each other, we need to conduct a **Post-Hoc Test**:

→ **Nemenyi Test** (**Compares all pairs of groups**)

Consequently, we aim to **plot the results** within a

Critical Difference Diagram

AdaBoost | Statistical Inference | Critical Difference Diagram

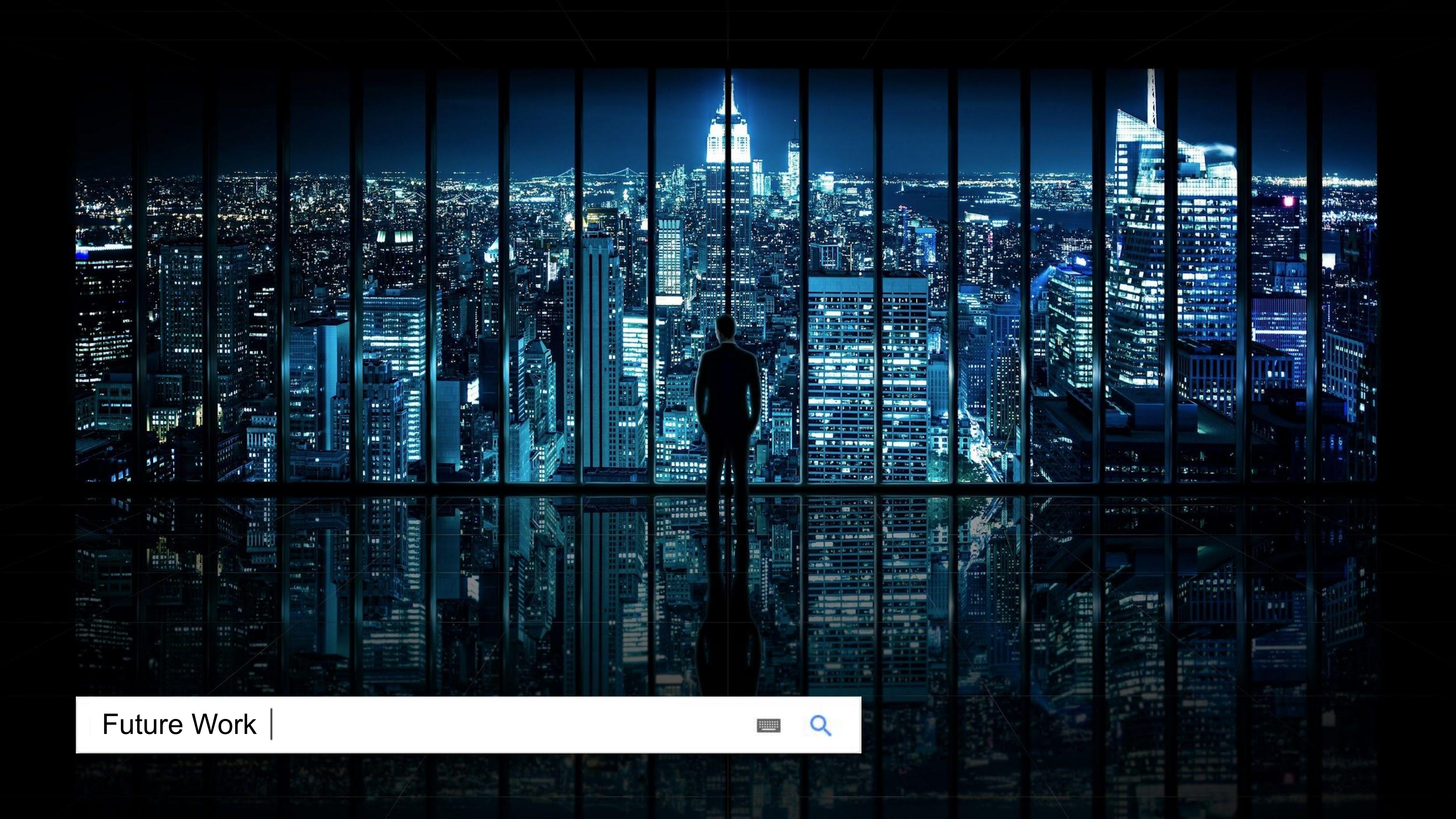


The AdaBoost with a **Decision Tree of Max Depth 3** and the **Standard Version** of AdaBoost model **perform similarly** and significantly **better** than the one that uses Perceptrons as weak learners.


Conclusions

AdaBoost | Conclusions

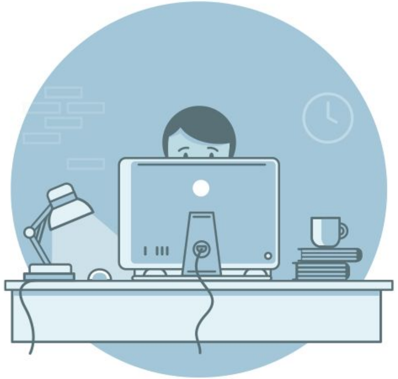
Despite our efforts, we **couldn't improve** AdaBoost for imbalanced data. However, our investigation provided valuable insights into AdaBoost's dynamics and the **impact of weak learners**. We found that **Decision Trees are more effective than Perceptrons** as weak learners, emphasizing the importance of **selecting appropriate weak learners** based on the dataset's characteristics.


Future Work

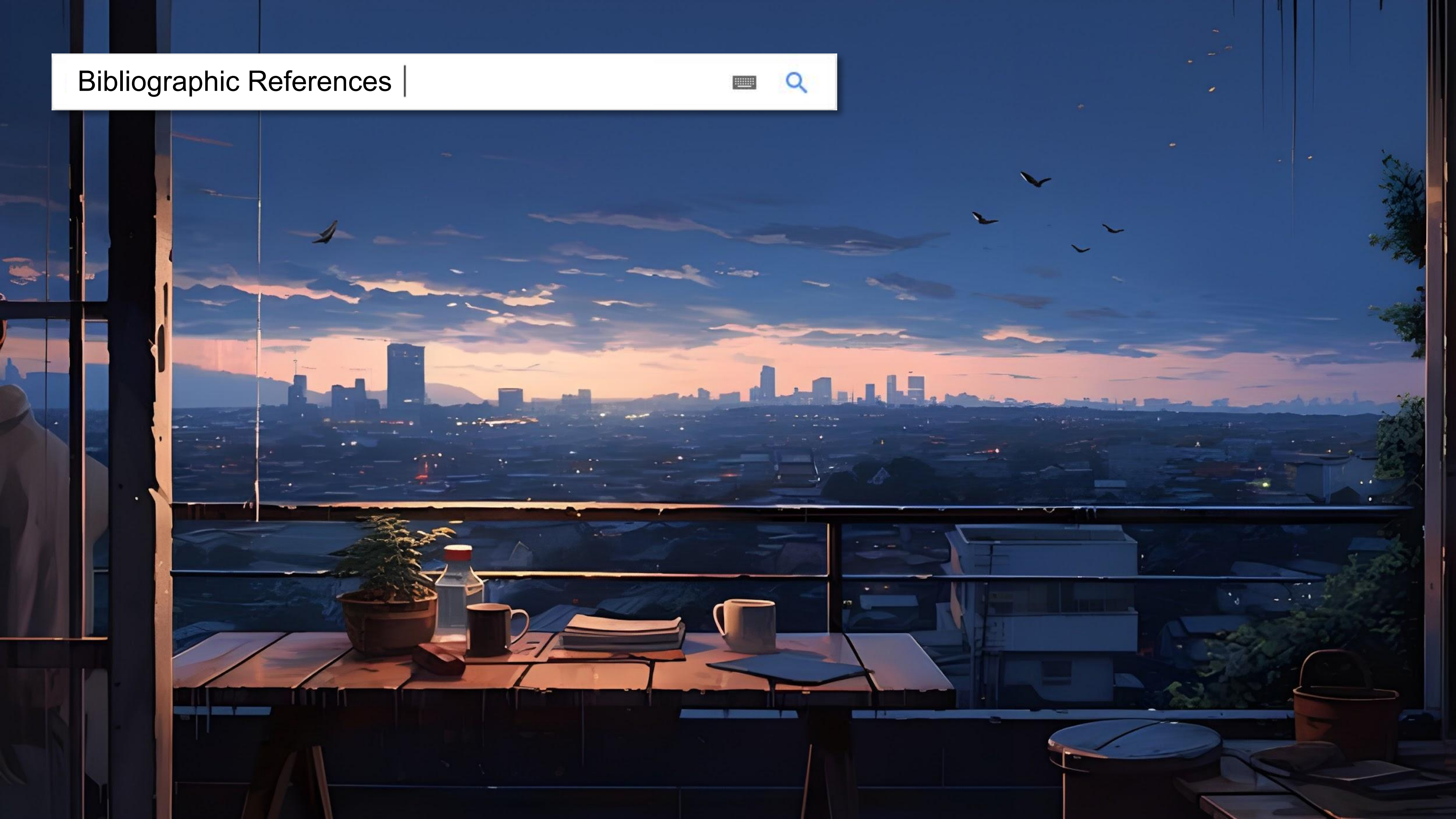
AdaBoost | Future Work

In the future, we considered using **other weak learners**, such as **Multi-Layered Perceptrons** (1 Hidden Layer), Logistic Regression or even **Support Vector Machines**. Analyse their performances and compare them with our previous work.

On another note, we could also tackle the algorithm's tendency towards overfitting. We intend to enhance the **error computation methods**, optimize the **update mechanisms for the weights** and **adjust the weighting strategy** for training data. These modifications aim to regulate the algorithm's focus on misclassified examples and **reducing the influence of noise and outliers**.



Bibliographic References



Bibliographic References | APA 6.th Ed.

→ Cano. Alvaro (Towards Data Science, 2021). *AdaBoost from Scratch*. Available **here**.

→ Geeks For Geeks (2023). *Boosting in Machine Learning | Boosting and AdaBoost*. Available **here**.

→ Geeks For Geeks (2024). *Implementing the AdaBoost Algorithm from Scratch*. Available **here**.

Thanks for the Attention!