AdaBoost
Analysis & Optimization

FCUP

Machine Learning I

2º. Semester 2023/24

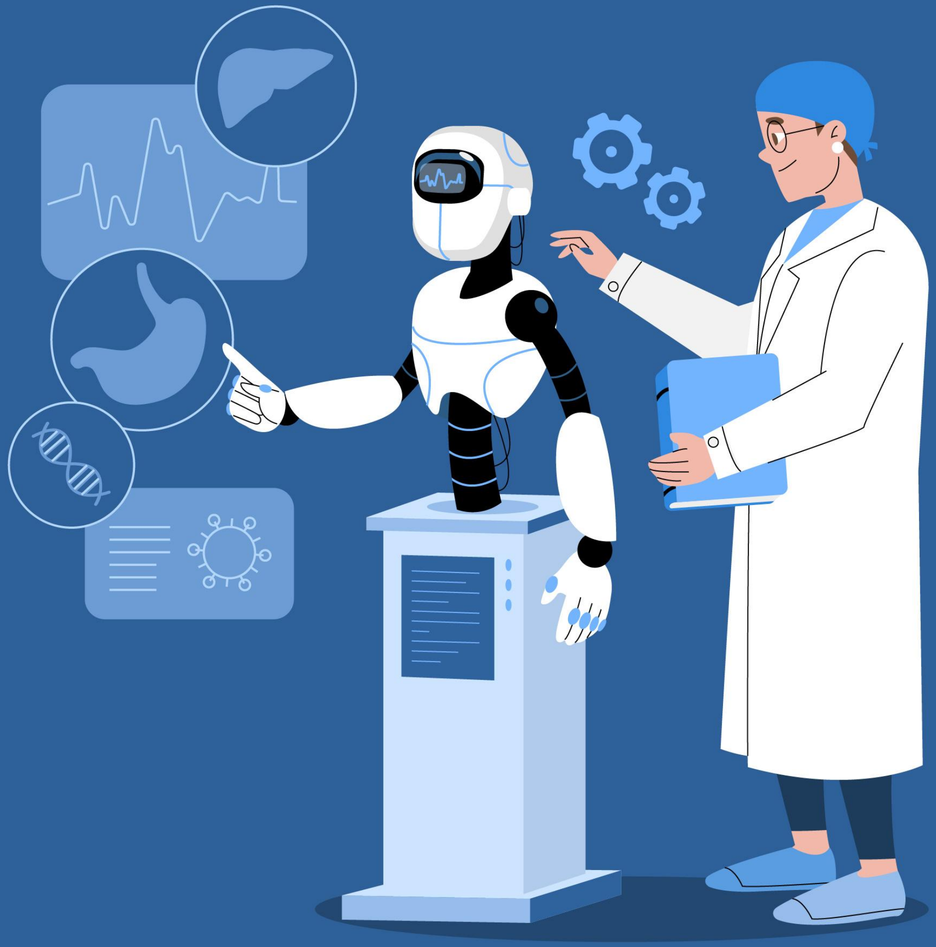
Gonçalo Esteves (up202203947)

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

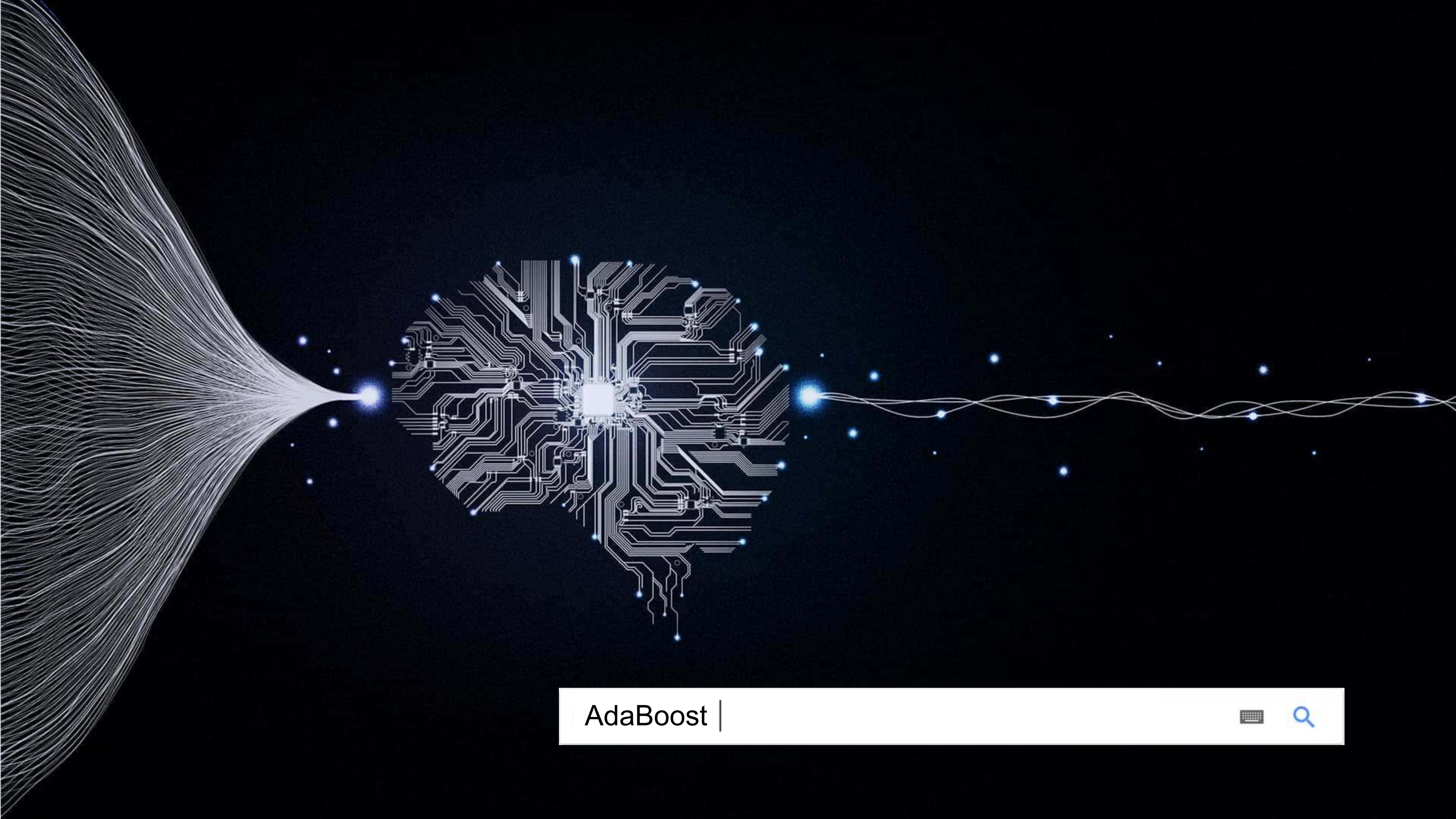
Furthermore, since the choice of algorithm greatly depends on the **specific task** and **data involved**, it's clear that **there is no "Master Algorithm"** (No algorithm can solve every problem).This **Project** focuses on the following topic:

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

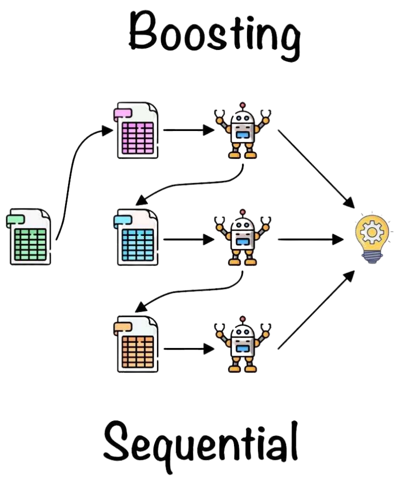
Introduction | Project Overview

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 used in machine learning to solve both classification and regression problems. It consists on training a **series of weak classifiers** on the dataset. Therefore, with each iteration, the algorithm **increases the focus** on data points that were previously **predicted incorrectly**.

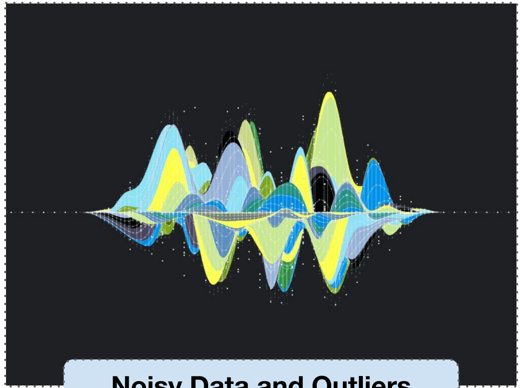
As a result, the AdaBoost algorithm builds a model by considering all the individual weak classifiers which are **weighted based on their performance**. Consequently, classifiers with **higher predictive accuracy contribute more to the final decision** which reduces the influence of less accurate ones in the final prediction.

AdaBoost | Limitations

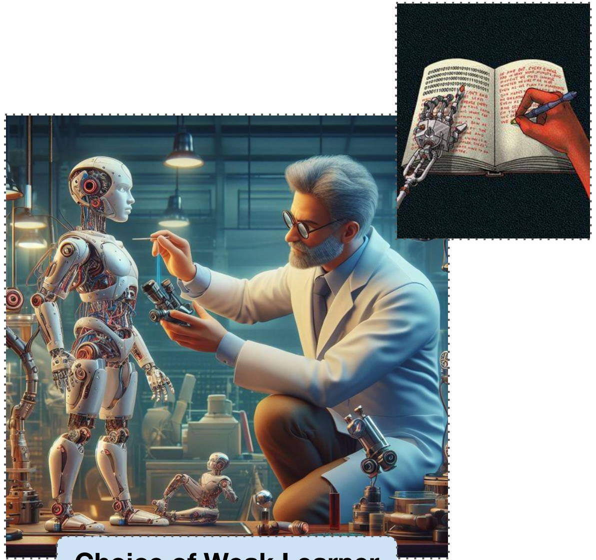
Despite the Algorithm’s powerful techniques, it yet remains limited by a few factors:

→ **Noisy Data and Outliers**

→ **Choice of Weak Learner**

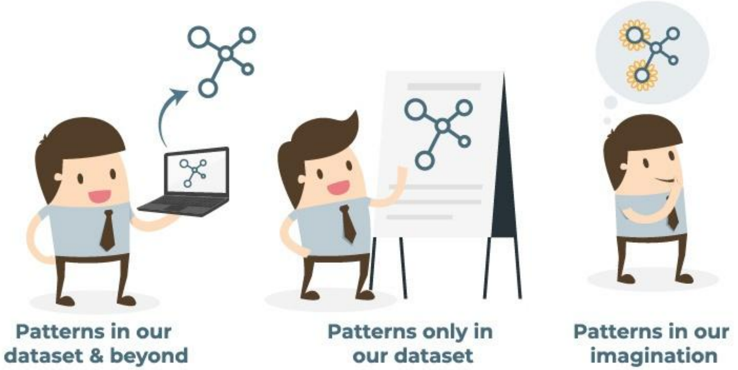


Noisy Data and Outliers



Choice of Weak Learner

AdaBoost | Limitations | Noisy Data and Outliers



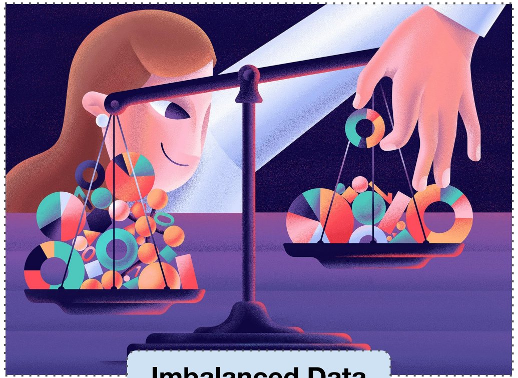
For instance, if the AdaBoost algorithm is used upon **noisy data**, it can focus too much on correcting previous misclassifications despite their actual trends, which can lead to **overfitting**.

Moreover, **outliers** can also impact the algorithm's performance significantly. Since AdaBoost **tries to perfectly fit each example**, it can overlook on rare instances in the data rather than **finding the actual patterns** within it.

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

Following the challenges faced by the AdaBoost Algorithm we chose to address its **performance issues** with **Imbalanced Data**. Due to its iterative nature, AdaBoost tends to **overlook data points that were previously misclassified**. This can cause the model to excessively focus on the **minority class**, which may include noisy data and outliers, thereby increasing the **risk of overfitting**.

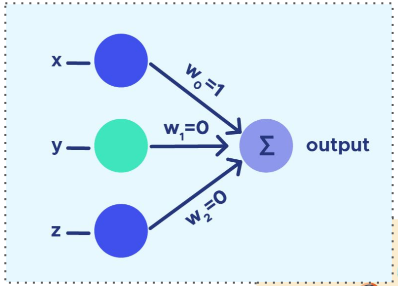
Consequently, we have embarked on an investigation to explore the **impact of weak learners** on the algorithm's effectiveness in these Binary Classification scenarios. This study aims to identify modifications in weak learners that might **improve** how AdaBoost **handles class imbalance**, enhancing its overall **predictive accuracy and robustness**.

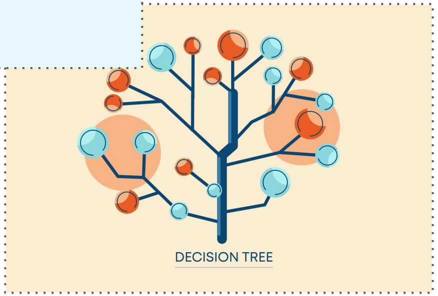
AdaBoost | Proposal | Choice of Weak Learners

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

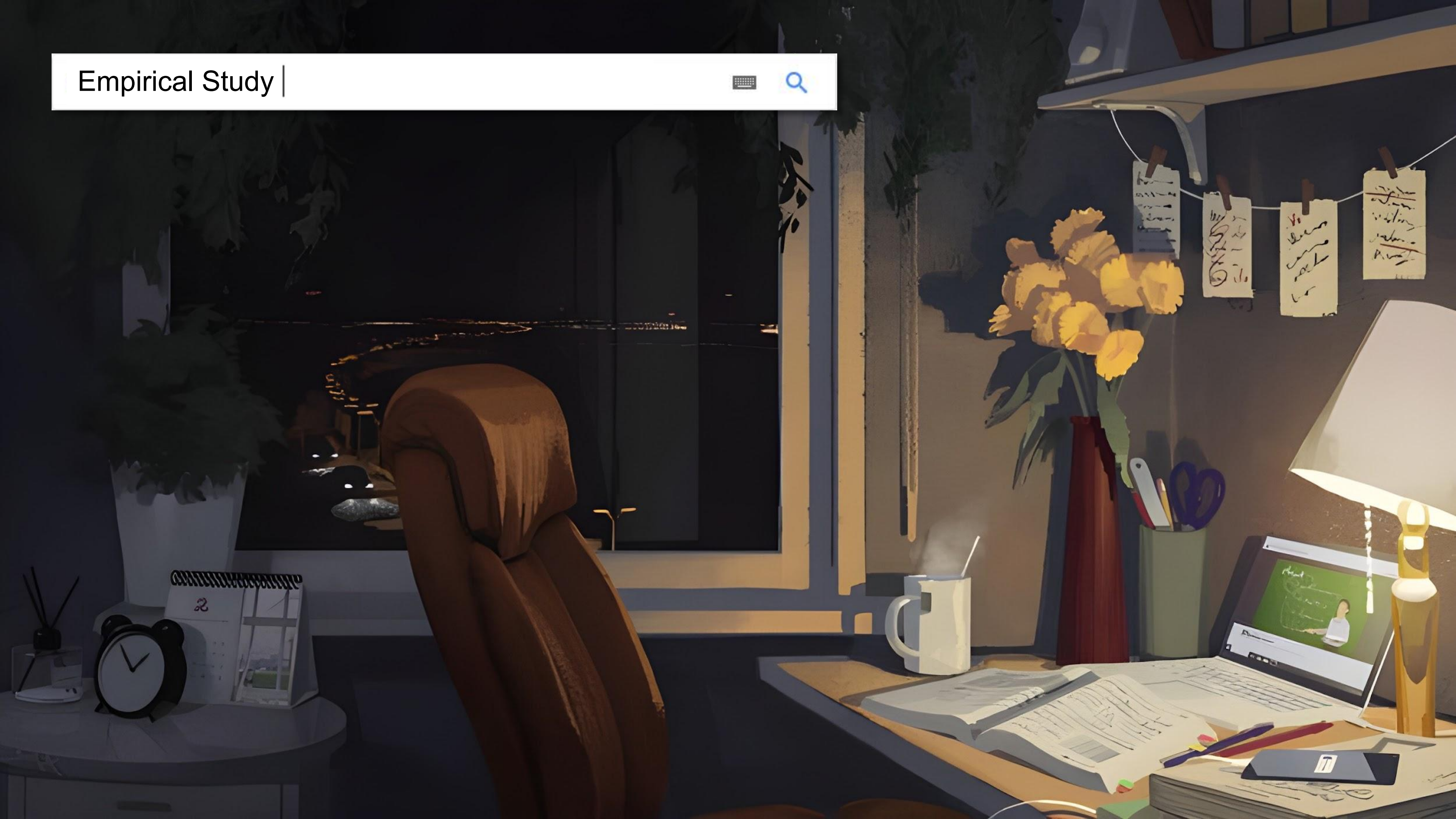
→ In AdaBoost, **decision stumps** are commonly used since they are less prone to overfit. However, **slightly deeper trees** (Max Depth of 3 as used in the Empirical Study) might be more beneficial to **capture more complex patterns** without causing significant overfitting.

Perceptrons

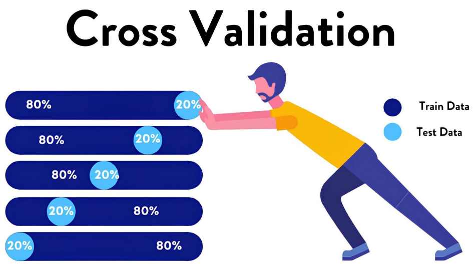
→ Perceptrons are **simple linear classifiers**, thereby for datasets with **linearly separable data**, they often converge quickly to a good solution which leads to **faster training times** compared to decision trees.

→ Due to their simplicity, Perceptrons can prevent the Model from becoming too complex. This 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



AdaBoost | Empirical Study | Experimental Setup



Since the developed AdaBoost Algorithm focuses on **Binary Classification Problems**, we decided to filter all the datasets inside the **OpenML-CC18 Curated Classification** study by selecting those with only **2 possible target values**.

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

→ **Number of Boosting Rounds** (M)

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

Therefore, to properly evaluate the models, we performed a **K-Fold Cross Validation** (5 Folds) in order to gain insights on how each model performs against the selected datasets.

AdaBoost | Empirical Study | Results

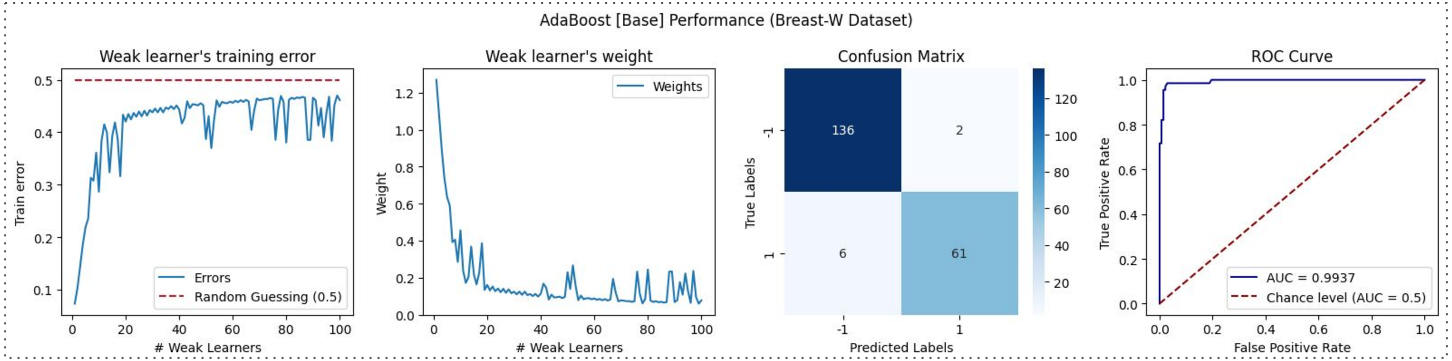
As we can observe from a **small snippet of the results**, the modified version of AdaBoost that use Decision Trees with increased depth appears to **perform slightly better** than the original version despite not being able to significantly outperform it. However the same does not occur when using Perceptrons as weak learners. In fact, the **use of Perceptrons makes the Algorithm perform slightly worse** than the standard version of AdaBoost.

Dataset	Positive Class (%)	Negative Class (%)	Majority Class (%)	AdaBoost [Base]	AdaBoost [DT - MaxDepth 3]	AdaBoost [Perceptron]
kr-vs-kp	0.477785	0.522215	0.522215	0.940859	0.987169	0.966204
breast-w	0.349927	0.650073	0.650073	0.963418	0.969257	0.963386
credit-approval	0.546708	0.453292	0.546708	0.872860	0.874422	0.685966
credit-g	0.300000	0.700000	0.700000	0.758000	0.769000	0.700000
diabetes	0.348958	0.651042	0.651042	0.769612	0.759121	0.664112
spambase	0.394045	0.605955	0.605955	0.933927	0.945226	0.855681
tic-tac-toe	0.653445	0.346555	0.653445	0.732755	0.986420	0.670152

AdaBoost | Empirical Study | Results

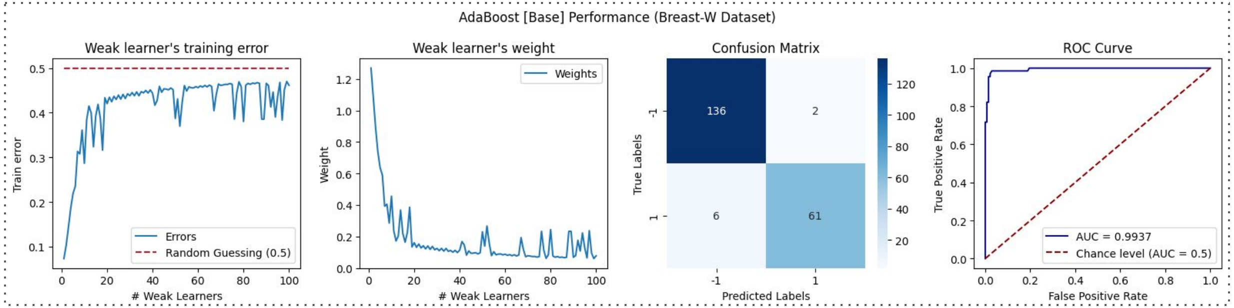
For instance, let's analyse how they **behave** against a given dataset → **Breast-W Dataset**

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**.



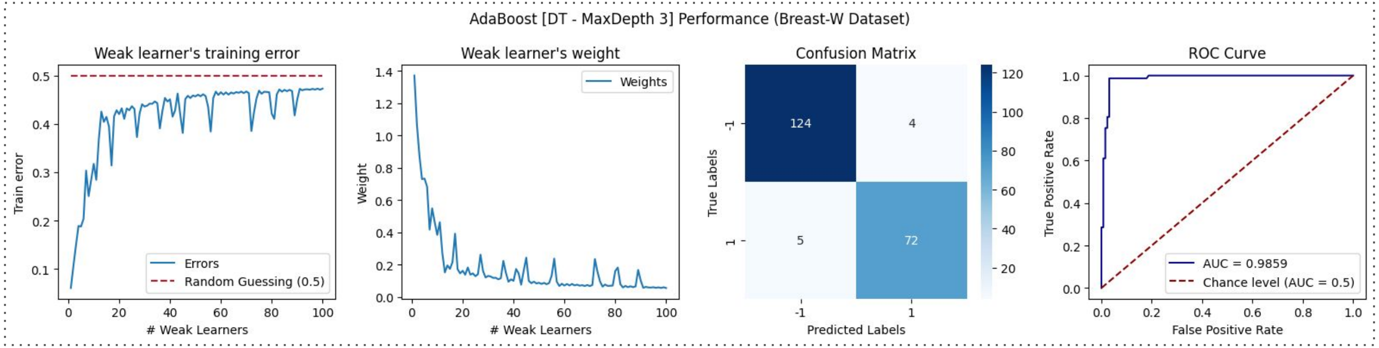
AdaBoost | Empirical Study | Results

As **boosting progresses**, the data used to train these weak learners begin to concentrate on previously misclassified examples (a characteristic of the AdaBoost algorithm → **reinforcing learning from past errors**). However, if these data points are actually **outliers**, the algorithm **risks overfitting** by continuously trying to **correct misclassifications** that **do not enhance pattern recognition** within the overall dataset. The algorithm demonstrates **strong performance** on the Breast-W dataset, which is evident from the Confusion Matrix (accurately classifies most of the samples) and the ROC Curve. With an **AUC of 0.9856**, we can confidently infer that the model **excels** in handling the provided dataset.



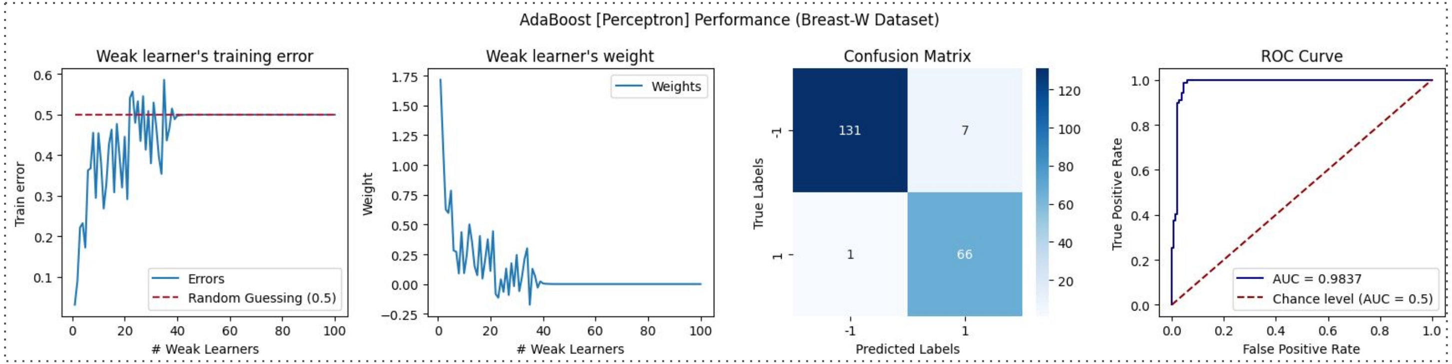
AdaBoost | Empirical Study | Results

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.



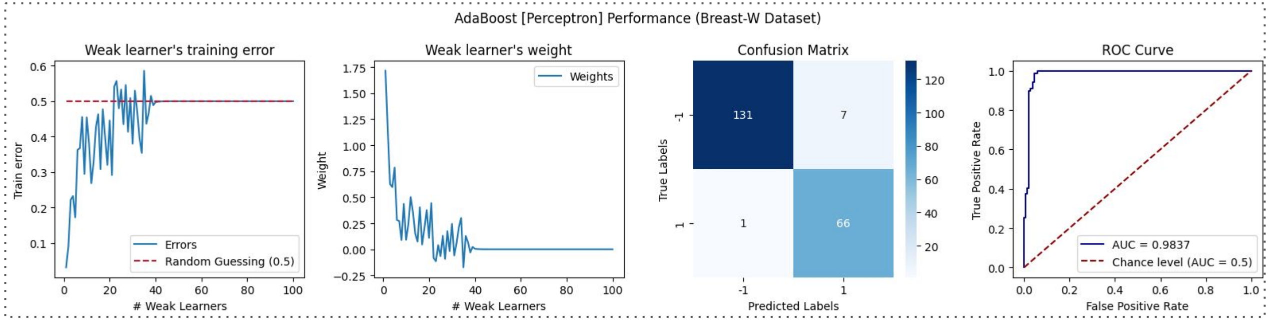
AdaBoost | Empirical Study | Results

The AdaBoost model using **Perceptrons** as weak learners shows **promising classification results** on the Breast-W dataset, but aspects of the performance closely **resemble random guessing**. This behaviour **raises concerns** about whether these learners can **outperform random guessing**, a **key expectation** in the AdaBoost iterative strategy effectively.



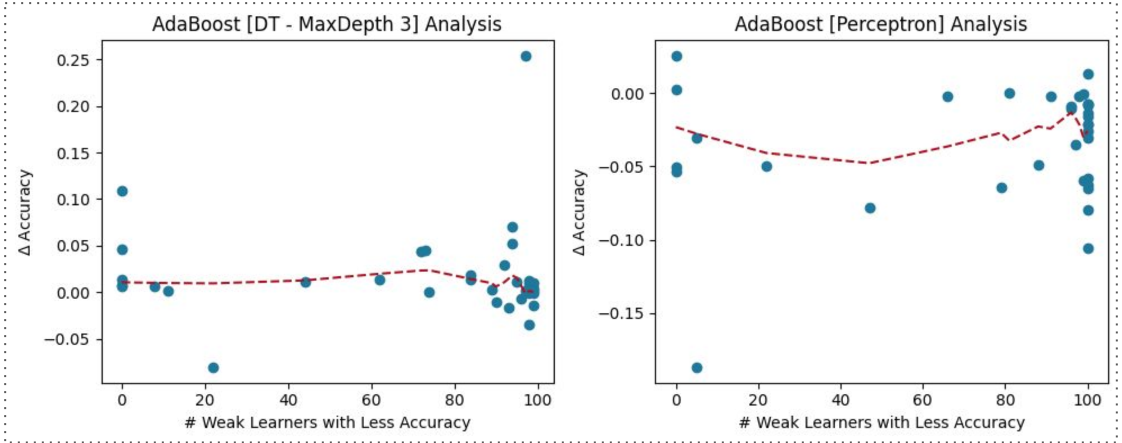
AdaBoost | Empirical Study | Results

A analysis of the **weak learner's weights** and training errors over 100 boosting rounds reveal that the **error rates** of individual weak learners **stabilize near the random guessing threshold** which suggests **minimal learning progress**. This pattern shows potential **limitations of Perceptrons** since AdaBoost's main concept (refine decision boundaries through targeted training on misclassified cases) may not be fully utilized. Moreover, the **influence of these weak learners** on the ensemble's decisions quickly decreases, indicating that **early learners might not provide a solid foundation** for subsequent ones.



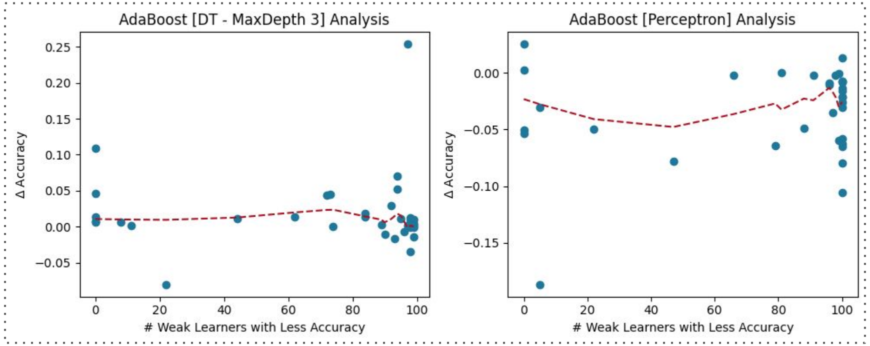
AdaBoost | Empirical Study | Results

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

On one hand, the **Δ Accuracy** for the AdaBoost model employing decision trees with Max Depth 3 **fluctuates around zero with a few outliers** that demonstrate **accuracy improvements**. This trend suggests that while this configuration performs **similarly to the Base AdaBoost** Model, it occasionally achieves superior results - when the **decision boundaries are more complex** but still manageable by the weak learner. In addition, this relatively stable performance indicates that **increasing the depth of the decision tree does not offer a clear advantage over simpler stumps**.

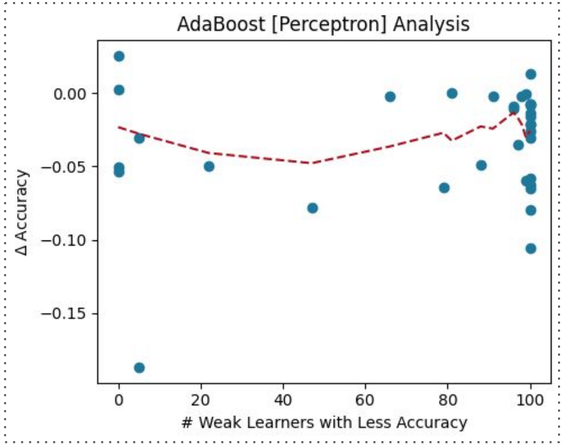


AdaBoost | Empirical Study | Results

On the other hand, the AdaBoost model utilizing Perceptrons shows a distinct **trend of negative ΔAccuracy**, mostly remaining below the zero mark.

In this version of AdaBoost, each **Perceptron** in the sequence is meant to **correct the misclassifications** of its predecessors. However, if the data **requires non-linear separation**, the linear boundaries created by each Perceptron **cannot effectively tackle the misclassified examples** unless those errors happen to align with a **linearly separable pattern** which is usually not the case with complex datasets.

In Suma, this consistent **negative ΔAccuracy highlights** the inability of Perceptrons to **capture non-linear relationships** in the data.



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.

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

AdaBoost | Statistical Inference | Decision Making

→ If ***P-Value* > α**, **H0 is not rejected** and therefore there is **not enough evidence** to conclude that there is any **difference between the model's performances**.

→ If ***P-Value* ≤ α**, **H0 is rejected** and therefore **not all the models perform equally well**.

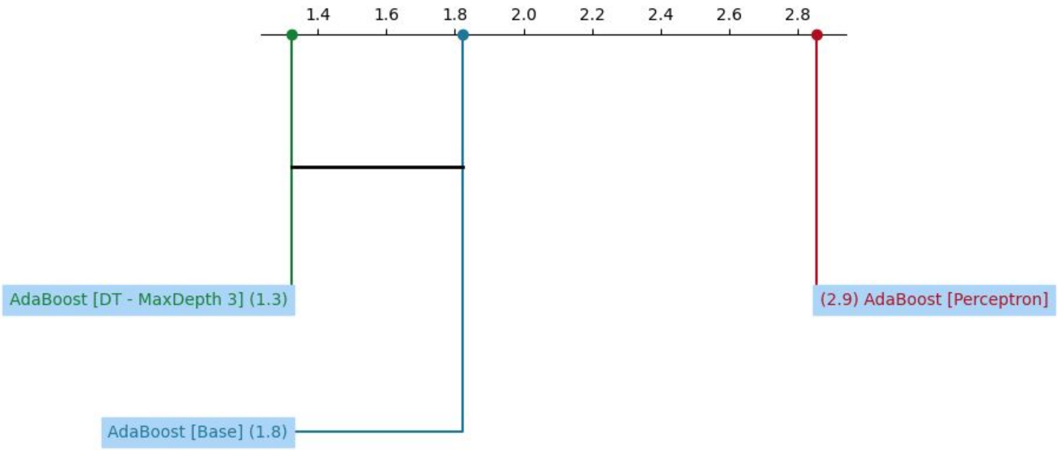
For a **significance level of 0.05** ($\alpha = 0.05$), the *P-Value* $\approx 1.048 \times 10^{-9}$ and therefore we **reject the Null Hypothesis (H0)** and consequently conclude that **there are significant differences** between the studied models.

AdaBoost | Statistical Inference | Post-Hoc Test

Since we **rejected the Null Hypothesis**, we concluded that there were significant differences between the model's performances. However, since the FriedMan Test alone does not explicitly convey **what specific models** are significantly different from each other, we need to conduct a **Post-Hoc Test** to determine which model pairs are significantly different.

After careful consideration, we chose to perform a **Nemenyi Test** (**Compares all pairs of groups** to determine which ones differ significantly) and plot the results within a **Critical Difference Diagram**.

AdaBoost | Statistical Inference | Critical Difference Diagram



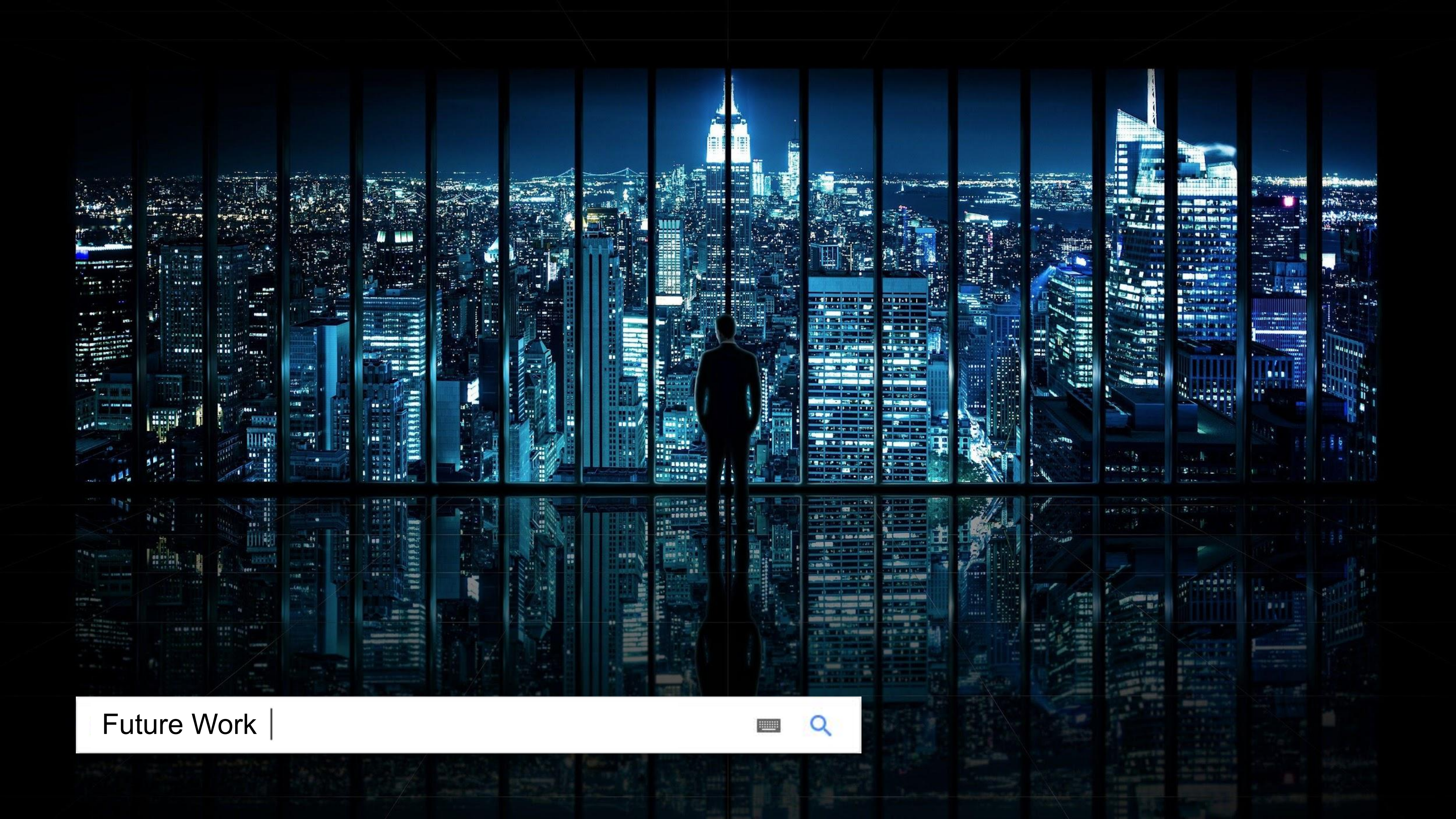
The AdaBoost with a **Decision Tree of Max Depth 3** and the **Standard Version** of AdaBoost model **perform similarly** and significantly **better** than the one that uses Perceptrons as weak learners.


Conclusions

AdaBoost | Conclusions

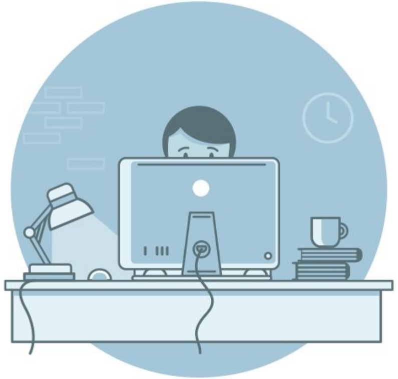
Despite our dedicated efforts to enhance the AdaBoost algorithm for imbalanced data, we were **unable to overcome the challenges** presented and thus did **not achieve an improvement over the original algorithm**.

Nonetheless, our investigation provided **valuable insights** into the dynamics of AdaBoost and the significant **impact** that the choice of weak learners has on its performance across different types of datasets. Most notably, we conclusively determined that **Decision Trees are more effective than Perceptrons** when used as weak learners within the AdaBoost framework. This understanding reveals the importance of selecting **appropriate weak learners** based on the specific characteristics and demands of the dataset at hand.

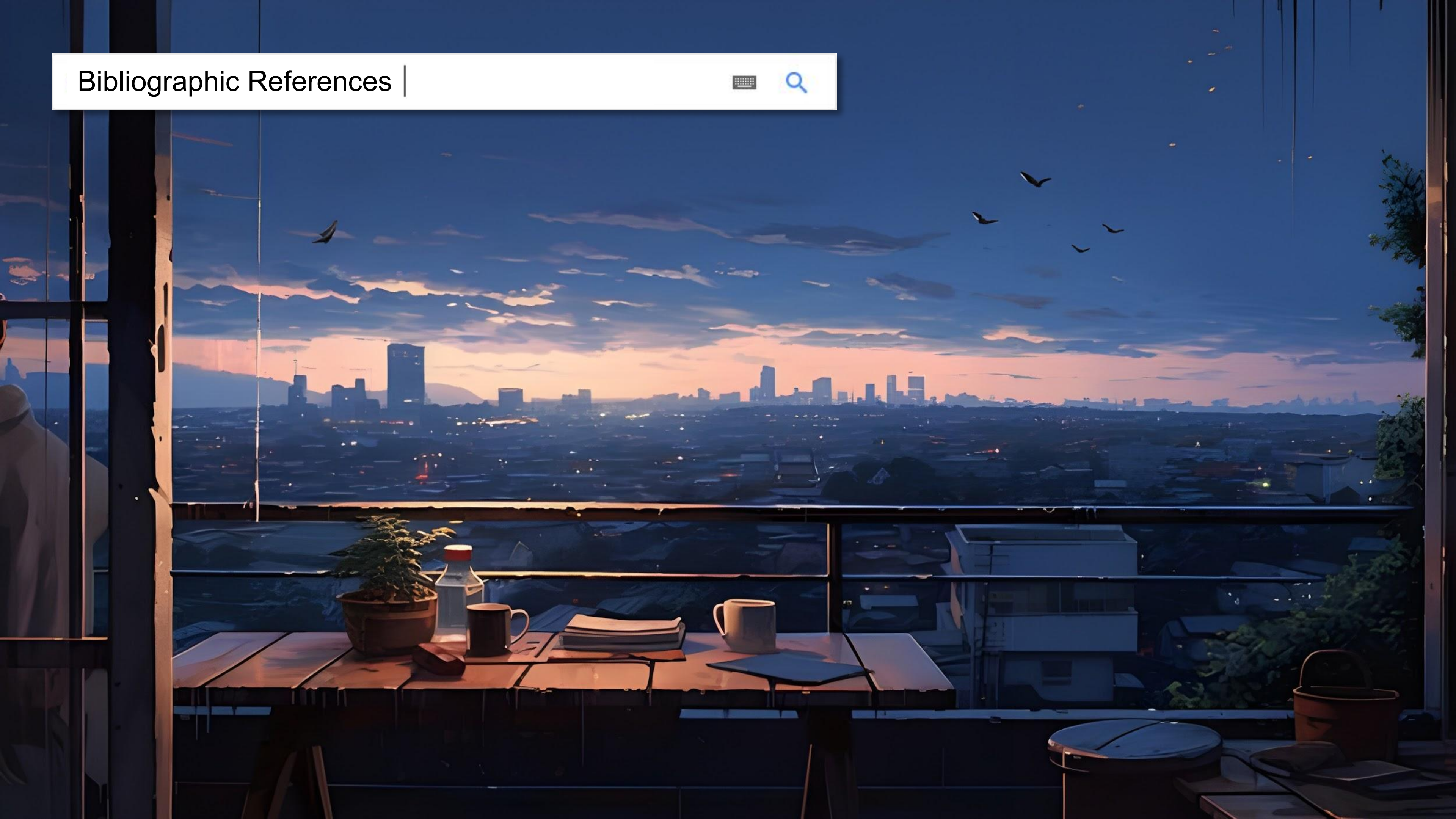

Future Work

AdaBoost | Future Work

In our **Future Work** on the AdaBoost Algorithm, we plan to tackle its tendency toward **overfitting** by refining crucial elements of its architecture. We intend to enhance the **error computation methods**, optimize the **update mechanisms for the weights** of each weak learner, and **adjust the weighting strategy** for training data in subsequent boosting rounds. These modifications aim to **regulate the algorithm's excessive focus on previously misclassified examples**, reducing the undue **influence of noisy data and outliers**. By making these adjustments, we anticipate improving the model’s generalization capabilities and achieving more **stable and accurate predictive performances**.



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!