# Exercise 1 Report

**Submission Date: 22/12/2024**

**Student Name: Adir Serruya**

**Student ID: 316472455**

## Introduction

In this exercise, I investigate and enhance decision tree models. Specifically, I address the limitation of binary routing in standard decision trees by introducing soft splits. Additionally, I evaluate the performance of these models on multiple datasets using a rigorous evaluation framework.

### Programming Task 1: Soft Splits in Decision Trees

#### Code Implementation

## For the soft split implementation, as requested, only the inference functionality was modified.

## Soft Split Function: I began by implementing a soft\_split function. This function evaluates a sample at a split node and assigns a probability of routing the sample to the "opposite" branch instead of the branch dictated by the splitting condition. This probability is controlled by the alpha parameter:

## With probability, the sample is routed in the opposite direction.

## With probability 1−, it is routed in the expected direction based on the splitting condition.

## predict\_sample\_proba Implementation: I then implemented the predict\_sample\_proba function. This function simulates the soft split inference for a single sample nnn times (where nnn is the number of simulations). During each simulation, the sample is routed probabilistically through the tree, producing a probability vector. These simulations are averaged to generate a final probability vector for the sample.

## Overriding predict\_proba: Finally, I overrode the predict\_proba method of the sklearn decision tree. The new implementation uses predict\_sample\_proba for all samples in the dataset and returns the averaged probability vectors across nnn simulations. This ensures that the final predictions reflect the uncertainty introduced by the soft splits.

## The training process remains unchanged, preserving the original behavior of the sklearn decision tree during the fit process.

## 1.2 Datasets Used - Classification

For evaluating the classifier, I used the following datasets, each of which has more than 1,000 samples and a variety of features and target classes:

1. **Obesity Dataset:**
   * **Description:** A classification dataset used to predict obesity levels based on health-related behaviors.
   * **Samples:** Over 2,000 samples.
   * **Features:** 16 Total, Gender, Agem Height, Weight, Family history etc.
   * **Target Classes:** Multi-class problem with different obesity categories.
2. **Adult Income Dataset:**
   * **Description:** Predicts whether an individual's income exceeds $50K/year based on demographic and economic attributes.
   * **Samples:** Approximately 48,000 samples.
   * **Features:** 14 Total, Includes age, education, work hours, and more.
   * **Target Classes:** Binary classification (above or below $50K income).
3. **Wine Quality Dataset:**
   * **Description:** Predicts the quality of wine based on chemical properties.
   * **Samples:** Over 1600 samples.
   * **Features:** 11 Total, Includes acidity, alcohol content, and other chemical measures.
   * **Target Classes:** Multi-class classification (wine quality scores).
4. **Bank Marketing Dataset:**
   * **Description:** Predicts whether a customer will subscribe to a term deposit based on marketing campaign data.
   * **Samples:** Over 45,000 samples.
   * **Features:** 17 Total, Includes contact duration, previous campaign outcomes, and more.
   * **Target Classes:** Binary classification (subscribed or not).
5. **Student Success Dataset:**
   * **Description:** Predicts student success in education based on demographic and academic performance.
   * **Samples:** Over 4000 samples.
   * **Features:** Includes parental education level, study time, and previous grades.
   * **Target Classes:** Multi-class classification (Enrolled, Graduate, Dropout).

## Datasets - Regression

1. **Garments Worker Productivity Dataset**:

* **Description**: Predicts the productivity of garment workers based on various operational and environmental factors.
* **Samples**: 1,197 samples.
* **Features**: 14 total, including production efficiency, work hours, and departmental indicators.
* **Target Variable**: Worker productivity.

1. **Air Quality Dataset**:

* **Description**: Predicts air quality metrics based on atmospheric and environmental factors.
* **Samples**: 9,471 samples.
* **Features**: 16 total, including CO, NOx levels, temperature, and humidity.
* **Target Variable**: Air quality index or pollutant levels.

1. **Bike Sharing Dataset**:

* **Description**: Predicts the number of bike rentals based on weather conditions, time, and seasonality.
* **Samples**: 17,379 samples.
* **Features**: 16 total, including temperature, humidity, and wind speed.
* **Target Variable**: Number of bike rentals.

1. **Apartments for Rent Dataset**:

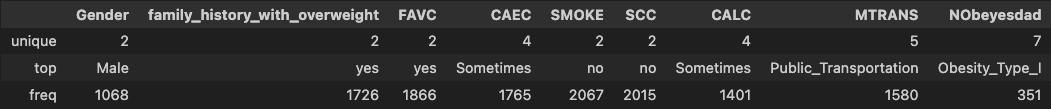
* **Description**: Predicts the rental price of apartments based on various property features and location information.
* **Samples**: 10,000 samples.
* **Features**: 21 total, including apartment size, number of rooms, and location data.
* **Target Variable**: Apartment rental price.

1. **Energy Efficiency Dataset**:

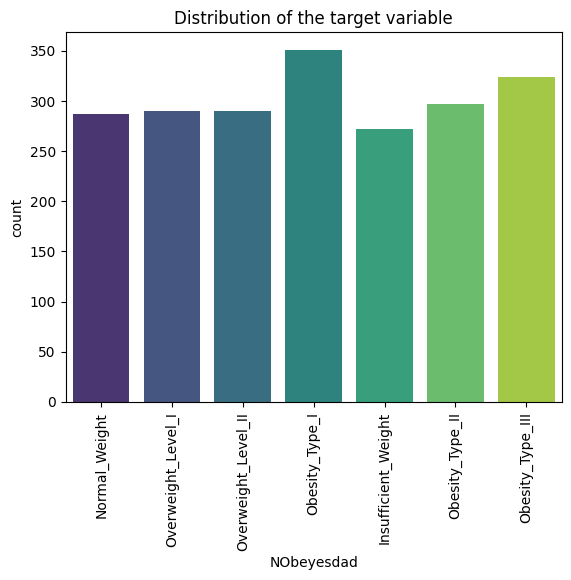
* **Description**: Predicts energy consumption based on household environmental and operational factors.
* **Samples**: 19,735 samples.
* **Features**: 28 total, including temperature, humidity, and equipment usage data.
* **Target Variable**: Energy consumption or efficiency metrics.

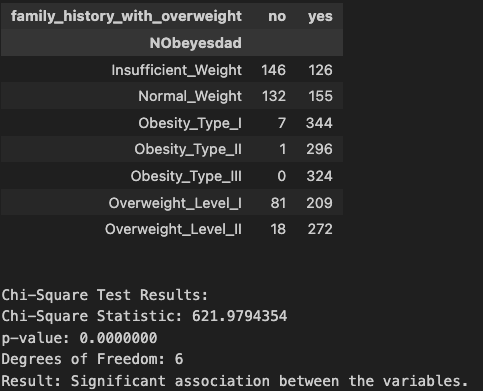
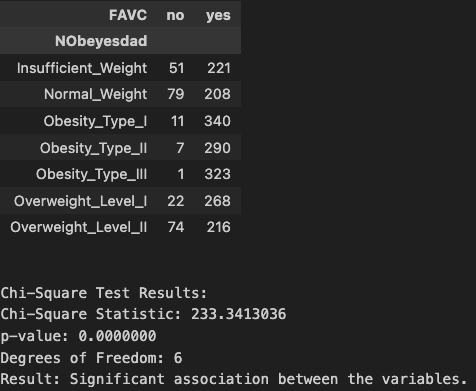
## 1.3 Exploratory Data Analysis – Obesity Dataset

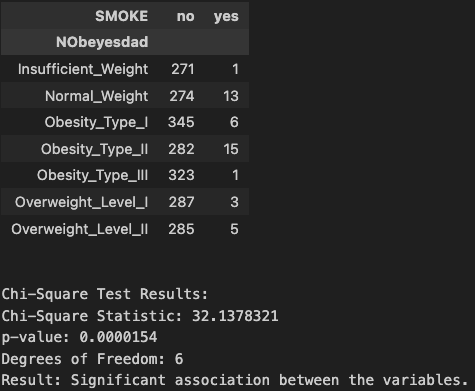
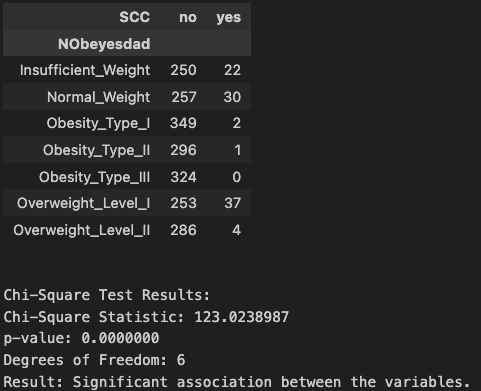
**Description Statistics of the continuous variables:**

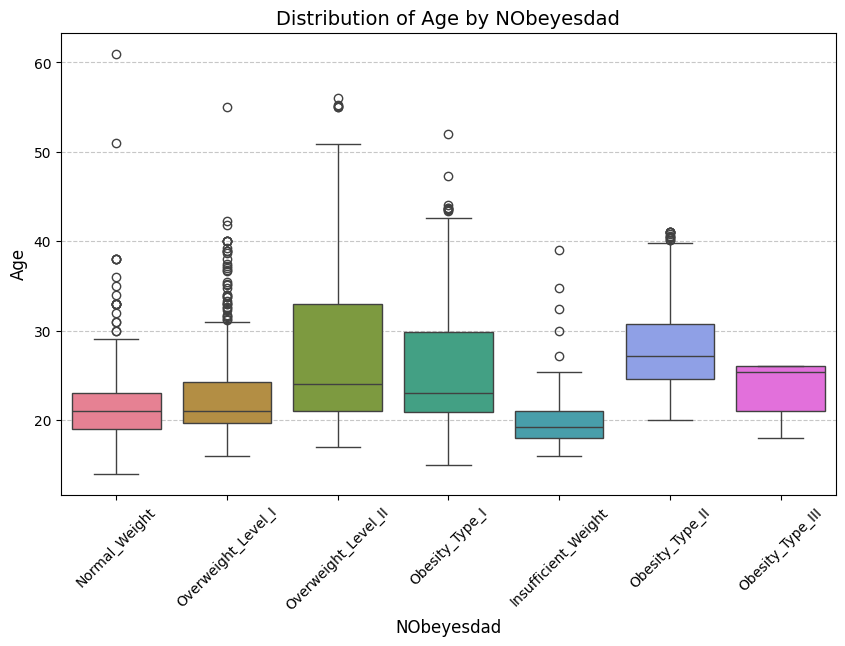
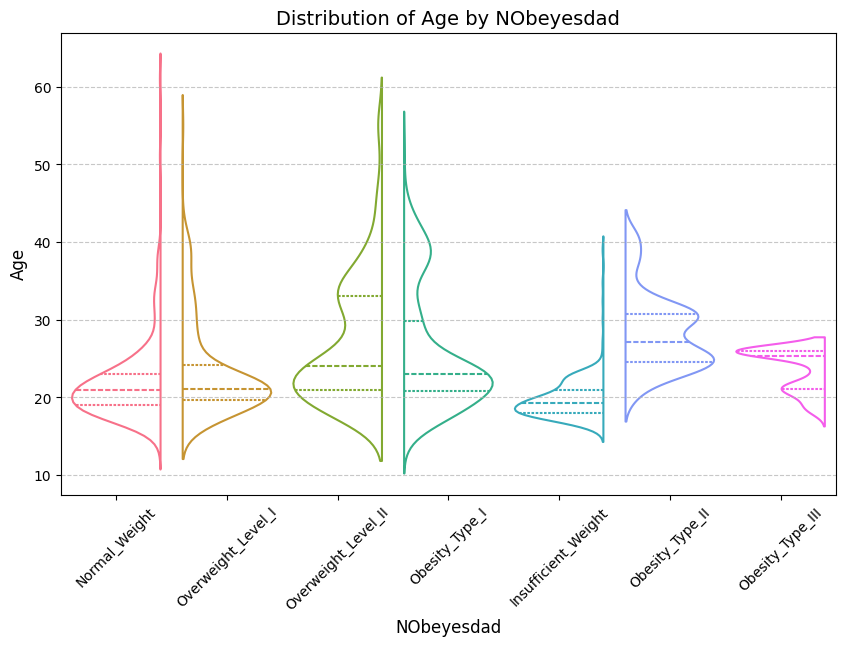
**Description Statistics of the categorical variables:  
**

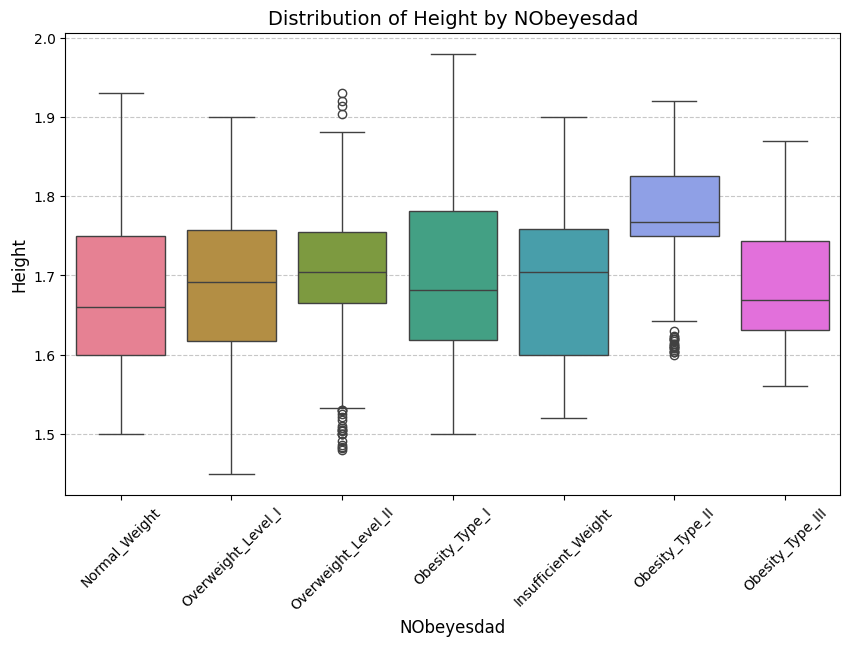
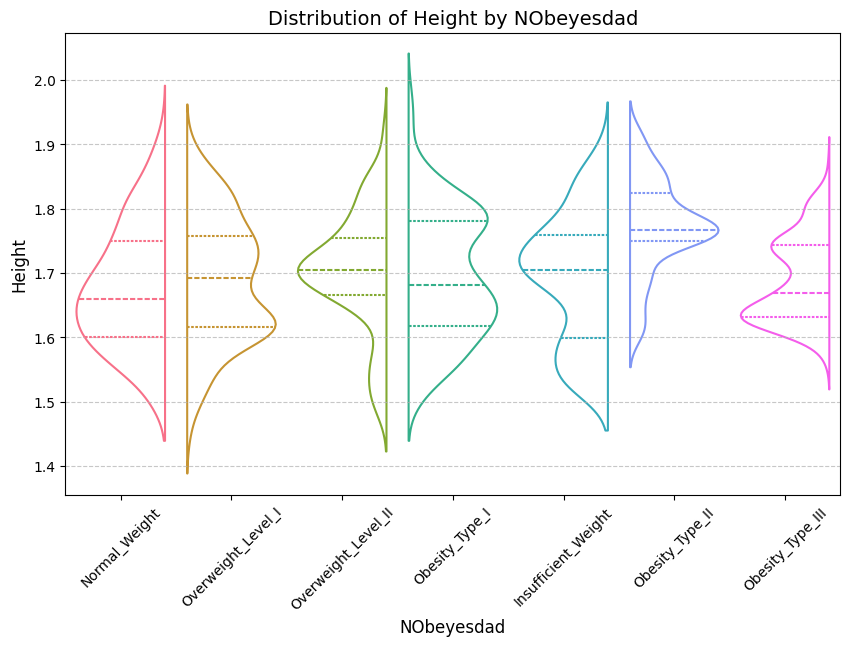
**Distribution of Target Variable:**

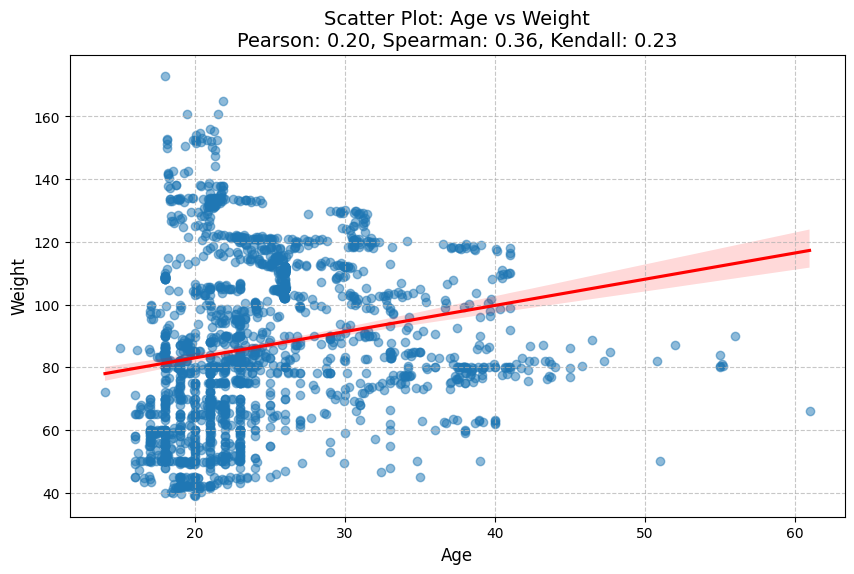


**Contingency Tables between Categoricals + Chi Square Test for Independence:**

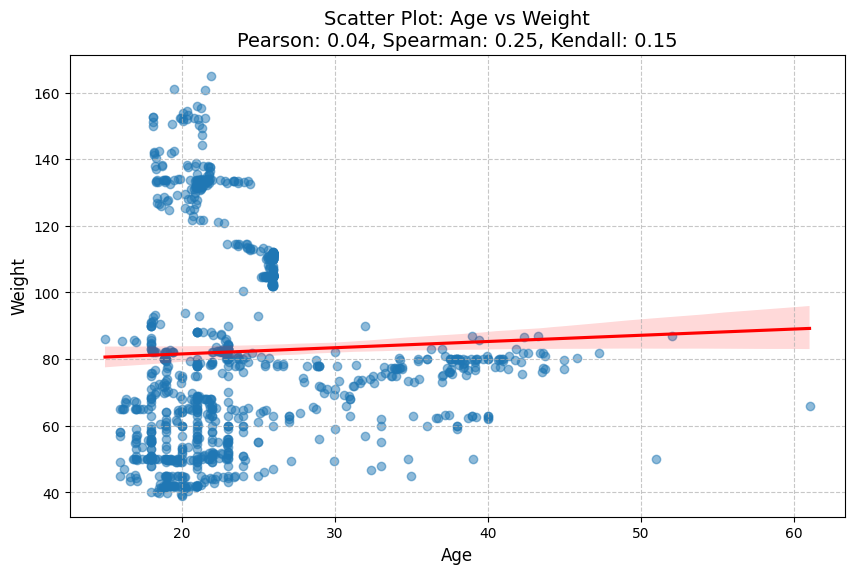
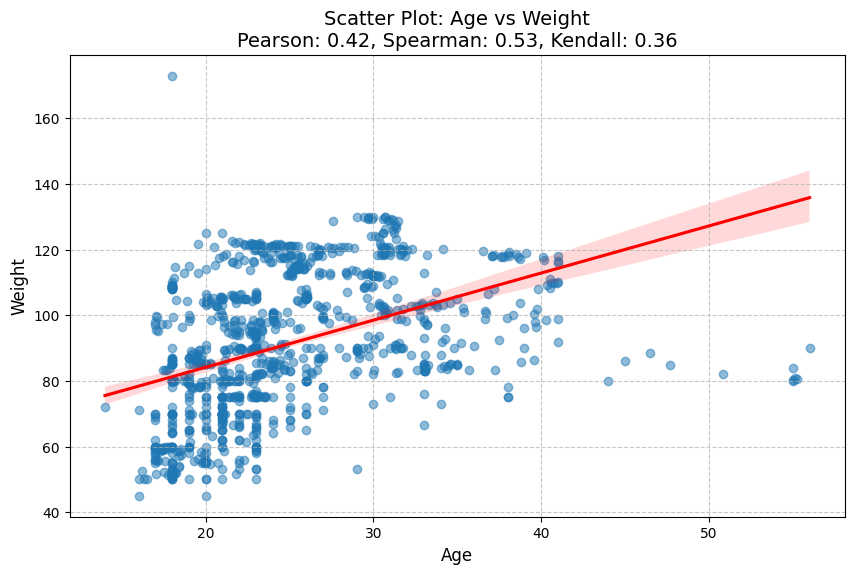


**Distributions of Continuous variables per category:**



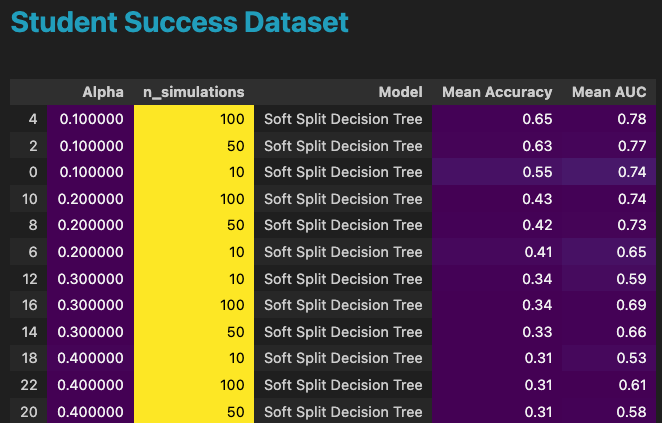
**Correlations between continuous variables**

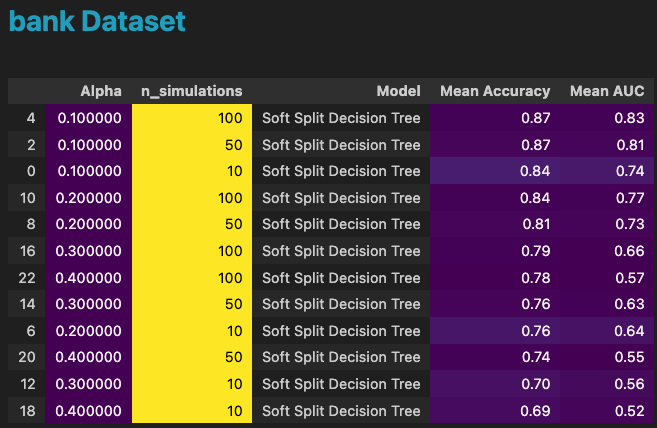
The above didn’t look very informative so I looked at it separately for females and males…

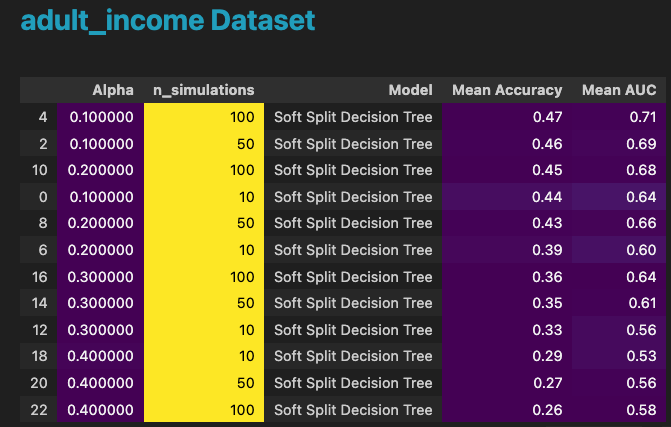
Females and Males Accordingly :

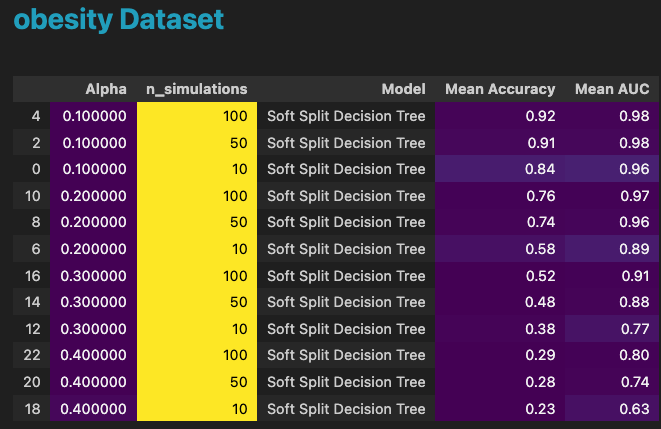
We can see that for males age is much more correlated with the weight, if the goal of the project was to get the best results possible I would probably try to create interaction features here.

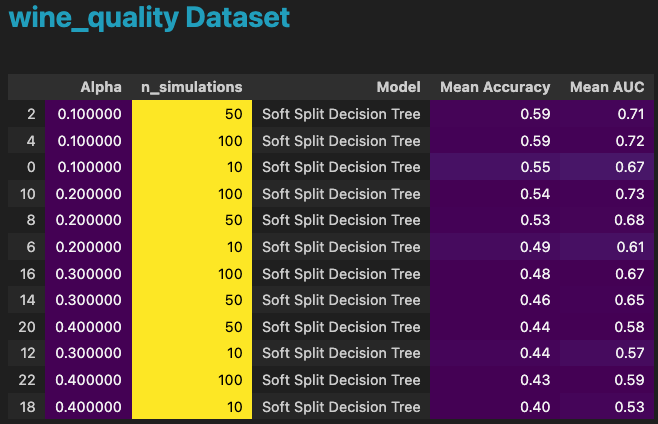
## Sensitivity Analysis for Hyperparameters – Task 1



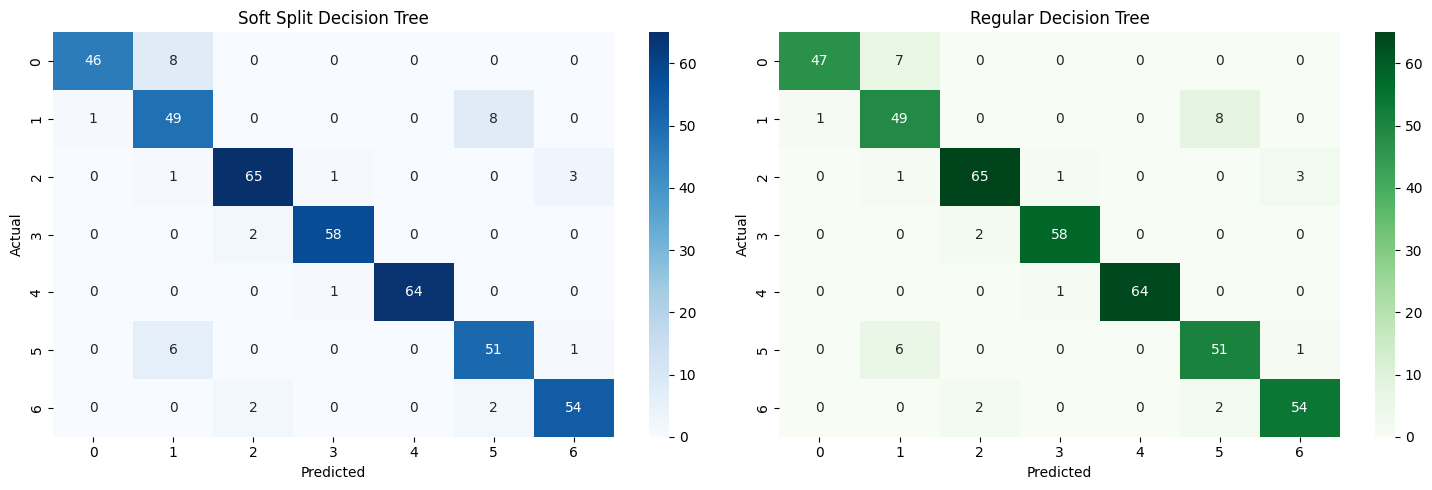


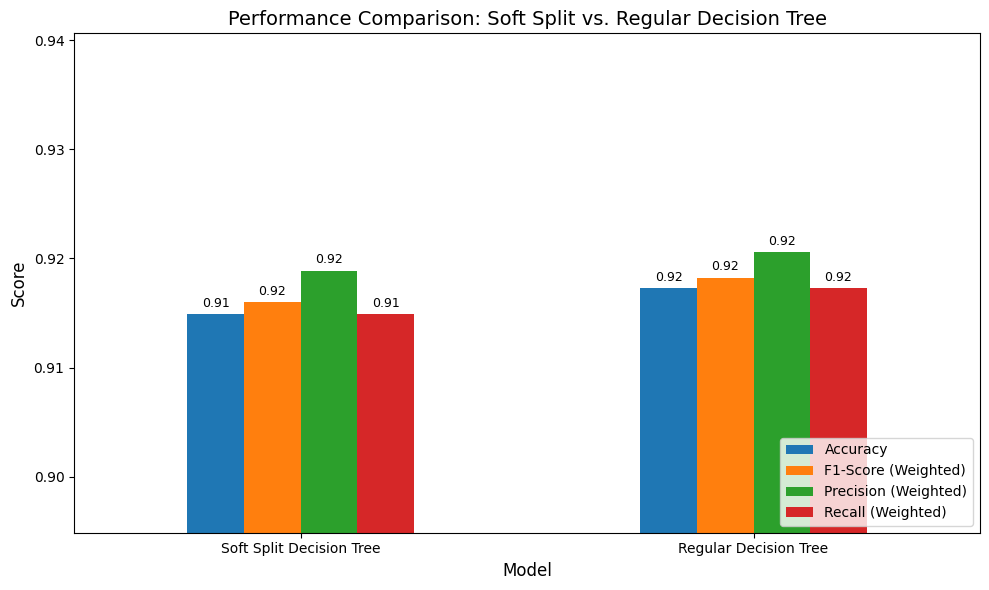






## Obesity Results Comparison





## Results Table – Task 1

* Note – for each dataset, I showed here the best result we got (per alpha / n\_simulations).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | Accuracy | AUC |
| Obesity | Original DT | - | - | 0.92 | 0.99 |
| Obesity | Soft DT | 0.1 | 100 | 0.92 | 0.98 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | Accuracy | AUC |
| Adult Income | Original DT |  |  | 0.89 | 0.92 |
| Adult Income | Soft DT | 0.1 | 100 | 0.47 | 0.71 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | Accuracy | AUC |
| Wine Quality | Original DT |  |  | 0.6 | 0.61 |
| Wine Quality | Soft DT | 0.1 | 100 | 0.59 | 0.72 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | Accuracy | AUC |
| Bank Campaign | Original DT |  |  | 0.87 | 0.7 |
| Bank Campaign | Soft DT | 0.1 | 100 | 0.87 | 0.83 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | Accuracy | AUC |
| Student Success | Original DT |  |  | 0.94 | 0.95 |
| Student Success | Soft DT | 0.1 | 100 | 0.65 | 0.78 |

# Programming Task 2: Regression with Soft Splits

## 2.1 Description of the Adaptation from Classification to Regression

## The transition from the SoftSplitDecisionTreeClassifier to the SoftSplitDecisionTreeRegressor involves the following key adaptations:

## 1. Prediction Objective

## Classification: The predict\_proba method returns class probabilities. Each leaf node contains class counts, which are normalized to compute probabilities for each class.

## Regression: The predict method returns a continuous numerical value. Each leaf node contains the average of the target values for the samples that fall into that leaf.

## Change Made:

## Replaced the \_predict\_sample\_proba method, which computed class probabilities, with \_predict\_sample, which computes the regression value (the mean of the target values in the leaf).

## 2. Leaf Node Representation

## Classification: Leaf nodes store the count of samples per class.

## Regression: Leaf nodes store the mean target value.

## Change Made:

## In \_predict\_sample, the returned value is directly accessed as the first value of the leaf node (tree.value[node][0, 0]), representing the mean target value for regression.

## 3. Soft Split Logic

## The logic of soft splits remains identical for both classification and regression. A sample is probabilistically routed to the left or right child node based on the alpha parameter.

## No Changes Needed:

## The \_soft\_split method is reused without modifications.

## 4. Aggregation of Predictions

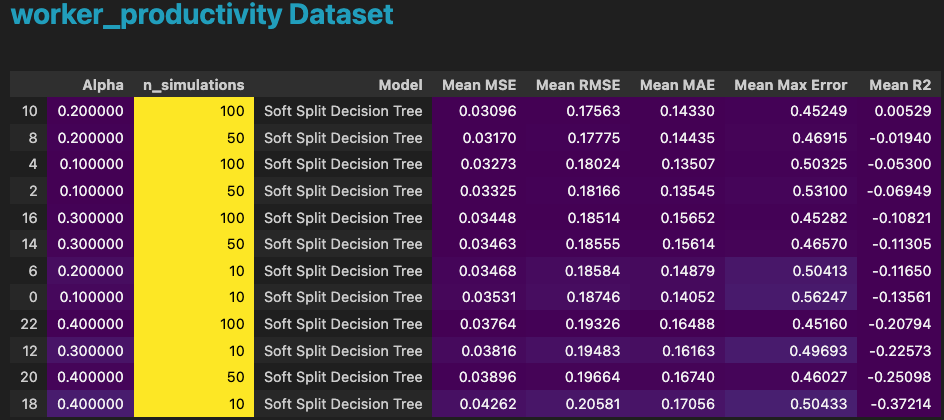
## Classification: Aggregates multiple simulations to compute the average class probabilities.

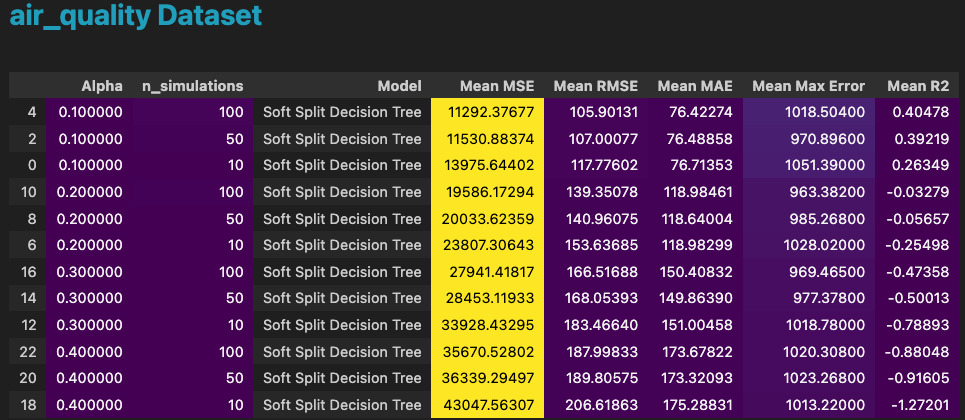
## Regression: Aggregates multiple simulations to compute the mean predicted value for regression.

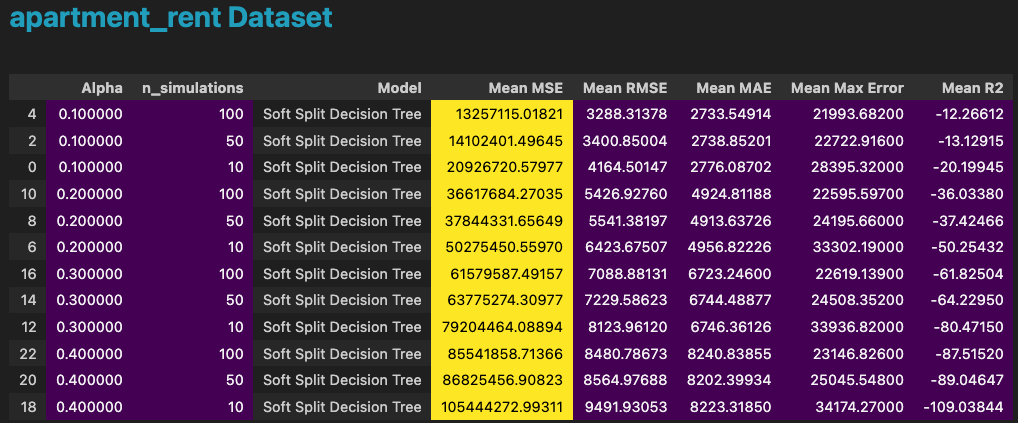
## Change Made:

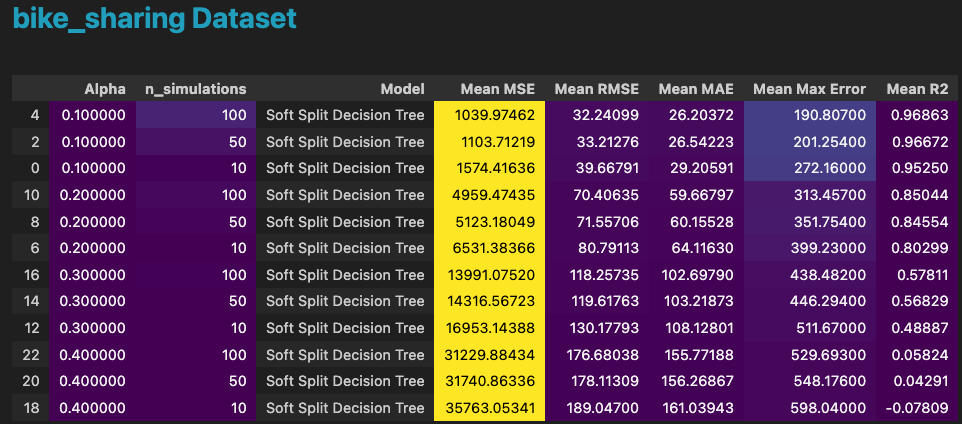
## In the predict method, predictions from multiple simulations are averaged to return the final regression value.

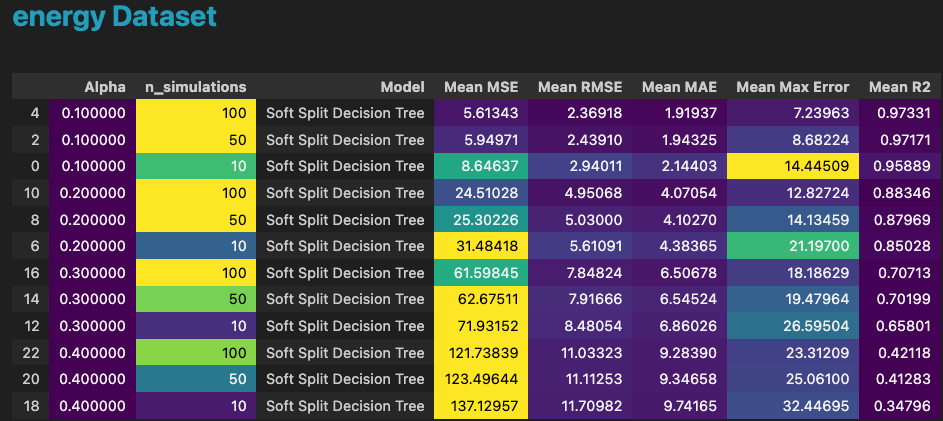
## 2.2 Results and Analysis











## Results Table – Task 2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | MSE | RMSE | MAE | Max Error | R2 |
| Air Quality | Original DT |  |  | 8917 | 94 | 19.51 | 1235 | 0.53 |
| Air Quality | Soft DT | 0.1 | 100 | 11292 | 105.9 | 76 | 1018 | 0.4 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | MSE | RMSE | MAE | Max Error | R2 |
| Apartment | Original DT |  |  | 740781 | 682 | 37.15 | 21697 | 0.55 |
| Apartment | Soft DT | 0.1 | 100 | 13,255,115 | 3288 | 2733 | 21993 | -12.27 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | MSE | RMSE | MAE | Max Error | R2 |
| Bike Sharing | Original DT |  |  | 33.62 | 5.78 | 2.67 | 99.7 | 0.99 |
| Bike Sharing | Soft DT | 0.1 | 100 | 1039 | 32.24 | 26.2 | 190.8 | 0.97 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | MSE | RMSE | MAE | Max Error | R2 |
| Energy | Original DT |  |  | 0.0008 | 0.0091 | 0.0065 | 0.06186 | 0.99 |
| Energy | Soft DT | 0.1 | 100 | 5.61 | 2.369 | 1.9 | 7.23 | 0.973 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | MSE | RMSE | MAE | Max Error | R2 |
| Worker Productivity | Original DT |  |  | 0.03 | 0.175 | 0.143 | 0.70 | 0.0017 |
| Worker Productivity | Soft DT | 0.2 | 100 | 0.03 | 0.175 | 0.143 | 0.45 | 0.0053 |

# Programming Task 3: Weighted Prediction

## Proposed Method: Improved Soft Splits Using Distance from Uniform Distribution

## Description

## In this alternative method, I propose weighting the decision tree leaves during prediction based on their distance from a uniform class distribution. The key idea is to adjust the randomness of routing decisions within the tree (via soft splits) by incorporating a measure of uncertainty—specifically, how far the class distribution in a node is from being uniform.

## Nodes with a high distance from a uniform distribution are considered more certain (i.e., dominated by a specific class), while nodes closer to uniformity indicate greater uncertainty. This additional information is used to dynamically adjust the split probabilities, promoting smoother decision boundaries and reducing overfitting.

## Theoretical Justification

## Soft Splits and Uncertainty:

## Soft splits introduce stochasticity in routing decisions, which helps to avoid overfitting by reducing deterministic biases in individual splits.

## Incorporating the distance from uniformity ensures that splits are guided by the reliability of the class distribution in the node.

## Distance from Uniformity:

## A uniform distribution indicates maximal uncertainty in class assignments.

## By penalizing nodes with higher uncertainty (closer to uniformity), I encourage more confident predictions in later stages of the tree.

## KL Divergence as a Measure:

## KL divergence quantifies how much a node’s class distribution diverges from a uniform distribution: DKL(P∣∣U)=∑i=1nP(i)log⁡(P(i)U(i))D\_{\text{KL}}(P || U) = \sum\_{i=1}^{n} P(i) \log\left(\frac{P(i)}{U(i)}\right)DKL​(P∣∣U)=i=1∑n​P(i)log(U(i)P(i)​) where PPP is the observed class probability, and UUU is the uniform distribution.

## Regularization via Adjusted Alpha:

## The split probability α\alphaα is adjusted based on the KL divergence:

## This ensures that nodes with uncertain distributions are less likely to make confident split decisions.

## Implementation Overview

## Calculate Distance from Uniformity:

## Compute the KL divergence between the node’s class distribution and a uniform distribution.

## Adjust Alpha Dynamically:

## Modify the split probability α\alphaα based on the calculated KL divergence.

## Soft Split Decision:

## Use the adjusted α\alphaα to probabilistically route samples during inference.

## Simulation for Robust Predictions:

## Perform multiple routing simulations for each sample and average the predicted probabilities to smooth the results.

## Steps I Took

## Enhanced Split Logic:

## The \_soft\_split function was updated to incorporate the distance from uniformity via KL divergence.

## This function adjusts the probability of going left or right based on the node’s uncertainty.

## Dynamic Weighting:

## The \_distance\_from\_uniform function calculates KL divergence to inform the split probability adjustments.

## Nodes with high KL divergence (low uncertainty) are weighted more heavily in routing decisions.

## Stochastic Predictions:

## The predict\_proba method runs multiple simulations for each sample.

## The averaged probabilities from all simulations ensure robustness against noise and overfitting.

## Advantages

## Reduced Overfitting:

## By dynamically penalizing uncertain splits, this approach discourages overconfident routing in noisy or ambiguous regions of the tree.

## Improved Generalization:

## Promotes smoother decision boundaries by incorporating uncertainty into routing.

## Robustness:

## Averaging over multiple simulations ensures stability in predictions.

## Disadvantages

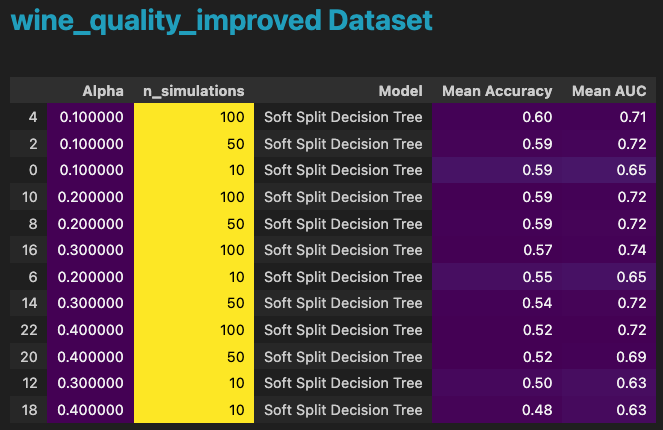
## Increased Computational Cost:

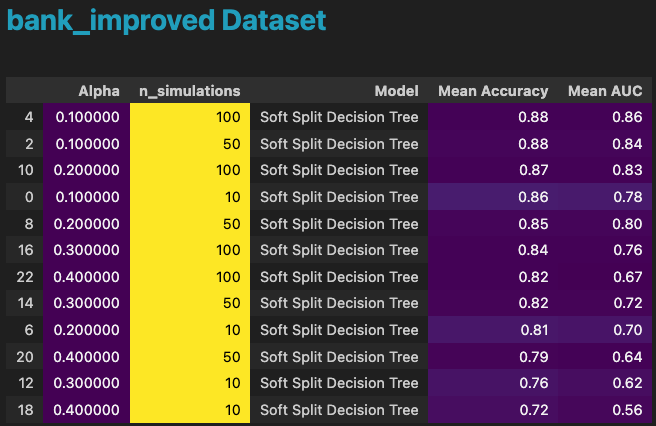
## Multiple simulations during inference increase computational complexity compared to standard decision trees.

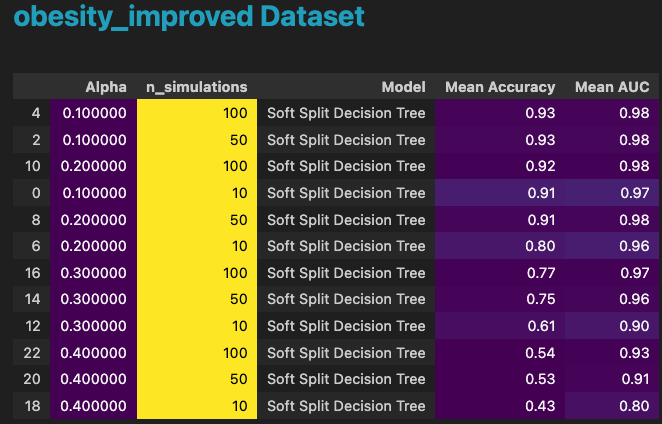
## Sensitivity to Hyperparameters:

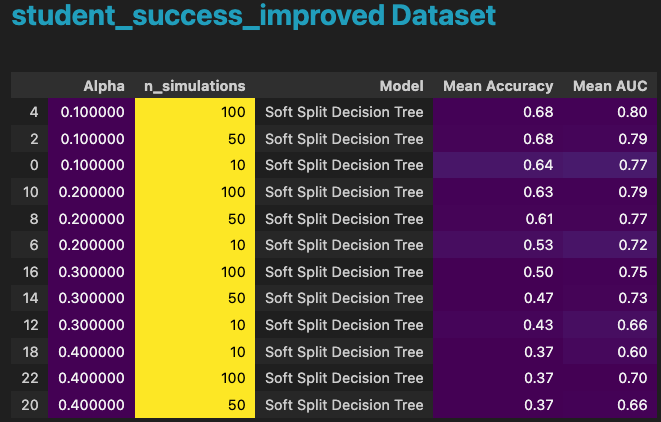
## The effectiveness of the method depends on the choice of α and the number of simulations

## 3.2 Results and Comparison









## 

## Results Table – Task 3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | Accuracy | AUC |
| Obesity | Original DT | - | - | 0.928 | 0.957 |
| Obesity | Improved Soft DT | 0.1 | 100 | 0.928 | 0.980 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | Accuracy | AUC |
| Adult Income | Original DT |  |  | 0.442 | 0.575 |
| Adult Income | Improved Soft DT |  |  | 0.483 | 0.707 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | Accuracy | AUC |
| Wine Quality | Original DT |  |  | 0.596 | 0.614 |
| Wine Quality | Improved Soft DT |  |  | 0.595 | 0.717 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | Accuracy | AUC |
| Bank Campaign | Original DT |  |  | 0.871 | 0.696 |
| Bank Campaign | Improved Soft DT |  |  | 0.875 | 0.855 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Dataset | Method | Alpha | N\_simulations | Accuracy | AUC |
| Student Success | Original DT |  |  | 0.679 | 0.726 |
| Student Success | Improved Soft DT |  |  | 0.681 | 0.797 |