|  |  |
| --- | --- |
| COS 314 Assignment 3  Analysis of Genetic Programming, Multi-Layer Perceptrons, and Decision Trees | Zanri Ophelia Greyling  u23586312  Sharif Ndlovu  u23763904  Aphumelele Zizipho Sineke  u25758935  Shivam Daya  u22739255 |

Table of Contents

[References 1](#_Toc198629991)

[Genetic programming 2](#_Toc198629992)

[Multi-layer Perceptron 3](#_Toc198629993)

[Decision Tree 3](#_Toc198629994)

[Result analysis 4](#_Toc198629995)

[Statistical test (Wilcoxon signed-rank test) 5](#_Toc198629996)

# References

Bobbitt, Z. (2020, February). *How to Perform the Wilcoxon Signed Rank Test*. Retrieved from Statology,org: https://www.statology.org/wilcoxon-signed-rank-test/

Dash, S. (2022, November). *Decision Trees Explained*. Retrieved from Towards Data Science: https://towardsdatascience.com/decision-trees-explained-entropy-information-gain-gini-index-ccp-pruning

Nyathi, T. (2025). Machine Learning. *COS 314 Lecture Slides*. University of Pretoria.

*Statistical Tables*. (2025). Retrieved from StatsCalculators.com: https://www.statscalculators.com/resources/statistical-tables/wilcoxon-signed-rank-table

*What is python scikit library*. (2024, April 12). Retrieved from Geeks for geeks: https://www.geeksforgeeks.org/what-is-python-scikit-library/

# Genetic programming

Genetic programming involves creating an initial population of randomly generated programs, represented as trees, out of function and terminal sets. The sets used for this algorithm are:

* Terminals: { "Open", "High", "Low", "Close", "Adj Close" }
* Functions: { "+", "-", "\*", "/", "max", "min" }

The syntax generation method used is ramped half-and-half, meaning that the population is a mix of full and partial trees.

Each population contains 10 individual programs with depths between 4 and 10. 10of these populations are created before the program terminates.

These programs are then evaluated according to a fitness function as calculated below:

Where:

* TP: correct Buy prediction
* TN: correct Don’t Buy prediction
* FP: incorrect Buy prediction
* FN: incorrect Don’t Buy prediction

Parent programs for the next generation of solutions are selected using the tournament selection method with a tournament size of 5.

Three different genetic operators are used, namely Crossover, Elitism, and Mutation.

Elitism involves selecting the top 5% of individuals, as ranked according to the fitness function, in the previous population and copying them into the new population.

The Crossover method used here is Subtree Crossover, which involves an exchange of subtree branches between two of the selected parents. This is done with a probability of 75%.

Lastly, Mutation is done of individual programs with a 25% probability. The Mutation used here involves selecting a random node in the program and replacing it with another value from the same set.

# Multi-layer Perceptron

The Multi-layer Perceptron was created using the existing Python Scikit-learn library. It is an open-source library designed to implement various machine learning and data analysis techniques, including decision trees, logistic regression, k-means clustering, and multi-layer perceptron classification.

The multi-layer perceptron algorithm provided by this library uses a “Rectified Linear Unit” or ReLU activation function, which returns f(x) = max(0, x). It uses a stochastic gradient decent solver with a constant learning rate schedule to update the weight values.

Samples in the training set are shuffled for each iteration, and the algorithm will stop when a maximum number of 200 iterations is reached, unless the algorithm has converged. Convergence is determined by a tolerance value. If the score does not increase by at least the tolerance value for a set number of iterations, the algorithm is assumed to have converged.

*Note that the code used for this section is relatively slow to run due to it having to call and load various other libraries. Pay close attention to the required libraries in the README and make sure they have been installed.*

# 

# Decision Tree

The decision tree was generated using Weka, an open-source machine learning software. It includes various tools for both transforming and analysing data and is capable of preforming many kinds of machine learning techniques, both supervised and unsupervised.

Manually, decision trees are generated through entropy calculations for each attribute. The root node is selected as the attribute with the highest information gain. The child nodes are then generated as subtrees by using subsets of the data determined by the parent attribute.

Gain is calculated as the change in entropy between the parent and the child:

A mathematical equation with numbers and symbols

AI-generated content may be incorrect.A black text on a white background

AI-generated content may be incorrect.

Where entropy, E(D) is calculated as:

A mathematical equation with numbers and symbols

AI-generated content may be incorrect.

A black text on a white background

AI-generated content may be incorrect.

# Result analysis

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **Seed** | **Training data** | | **Testing data** | |
| **Accuracy** | **Score** | **Accuracy** | **Score** |
| **Genetic Programming** | 123456789 | 0.637 | 0.743 | 0.706 | 0.780 |
| **ML Perceptron** | 123456789 | 0.942 | 0.944 | 0.970 | 0.969 |
| **Decision Tree** | 123456789 | 0.751 | 0.371 | 0.974 | 0.973 |

Genetic programming is a simpler algorithm compared to the multi-layer perceptron, returning results faster and using less storage. However, MLP returns superior results when used on both the training and testing data, performing slightly better on the testing data. This does not imply that GP gives inaccurate results, but a 70% accuracy is less desirable.

The most obvious reason for the difference in accuracy is the use of predefined Python libraries for the Multi-layer Perceptron. Because many machine learning algorithms require complex mathematical operations and matrix calculations, it is common to use methods, libraries, or algorithms that have already been developed, refined and tested. In the case of MLP , Python is a good choice to work with because it excels in the matrix calculations needed.

As for the Decision Tree, it gave the most accurate results on the training data out of all three algorithms, with MLP being a close second.

Decision trees are also the simplest of the algorithms to understand from a human perspective, as it generated an easy-to-understand tree diagram detailing its decision-making algorithm. GP has the potential to be easily interpretable, since it generates a mathematical function to make decisions. However, depending on the data and some randomization factors, the function can very quickly become extremely complex and uninterpretable.

## Statistical test (Wilcoxon signed-rank test)

The Wilcoxon signed-rank test is used to test for a significant difference between the average values of two sets of data. In this instance, we are using this test to analyse the difference between the f1 score obtained by Genetic Programming and the Multi-Layer Perceptron. For a clearer picture, we will run the algorithms repeatedly with various seeds to get a range of results.

To begin, we state our null and alternative hypotheses:

* H0 : The median difference between the two algorithms is zero, i.e. there is no significant difference between them.
* HA : The median difference is negative, i.e. MLP performs better than GP.

Next, we calculate the difference (as an absolute value) between each pair of data points. These pairs are then ordered from lowest to highest in terms of absolute difference. This is done in the attached Excel spreadsheet.

Now we find the sum of the negative differences and positive differences separately. The results are:

|  |  |
| --- | --- |
| Negative Sum | Positive Sum |
| **1,89054** | **0** |

Our test statistics will thus be T=1.89054. To obtain the critical value we refer to the “Wilcoxon Signed Rank Test Critical Values Table”, using n=7 and α=0.05, giving us C=4.

Our test statistic is less than the critical value, meaning that we reject H0 at a 0.05 level of significance. In other words, the median difference is negative, i.e. MLP performs significantly better than GP.