COS710 ASSIGNMENT 1 REPORT

ALEXANDER MUENDESI

Contents

Objective of Genetic Program

- The goal of this Genetic Program is to predict the Bike Trip duration given certain input variables like weather conditions and distance travelled for example.

Terminal Set

- The terminal set used consists of all the inputs in the columns in the dataset except the Trip Duration column.

Function Set

- The function set used consists of the following: {+, -, *, /}

Fitness Cases

- The first 70% of the dataset acts as the fitness cases for this program.

Raw Fitness Measure

- The summation of the absolute value of the difference between the actual values and the predicted values.
- The smaller the value of the resulting sum the better the fitness of the individual.

Parameters Used

- Population Size: 100
- Number of generations: 50
- Mutation rate: 0.70
- Crossover rate: 0.30
- Bound: Since the HITS ratio is not part of the required performance metrics, I did not find a use for this parameter.
- Maximum depth: 10
- Max offspring depth: 2
- Tournament size: 4

Initial Population Generation

- The initial population is generated using the Grow Method only. The process was restricted from generating a single Terminal node as the root in the initial population as that would be bad for diversity. The program maximises for diversity in the initial population generation stage.

Selection Method

- Tournament selection was used only in this program. Individuals where randomly selected and then the individual within that group with the best fitness is selected as the winner of the tournament.

Stopping Condition

- The program uses of a fixed stopping condition of executing for 50 generations every run then stopping and the best individual is returned from last generation.

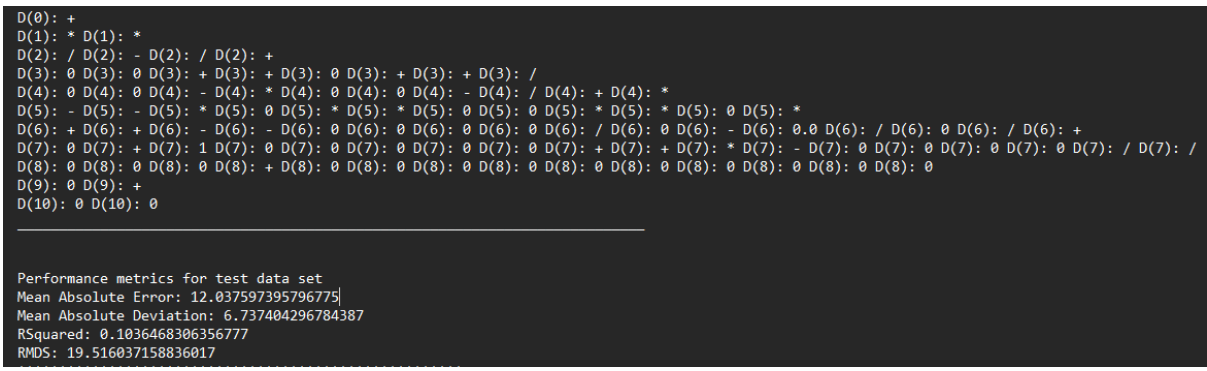
Comment on Subtree Crossover and Mutation

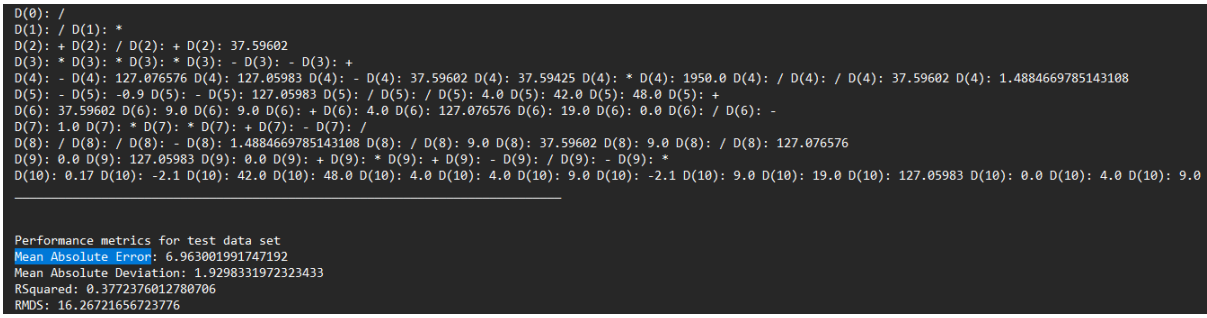
- To create the new population, subtree crossover is performed first until a desirable number of individuals are created from subtree crossover. Then mutation is applied to generate the remainder of the population.
- When the max tree depth is violated for subtree crossover and mutation, one of the original parents is randomly returned instead of pruning the offspring(This is one of the suggestions given by Koza to handle the issue of offspring depth).
- The offspring produced by the mutation operator have a maximum depth of 2.
- Subtree crossover is used for exploration in the program space whilst mutation is used for exploitation in the program space.

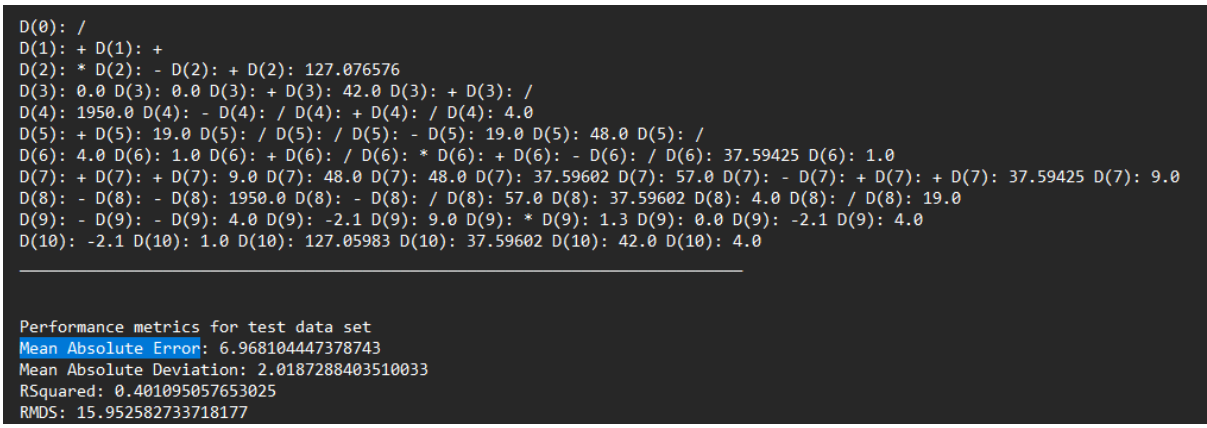
Comment on Dataset used

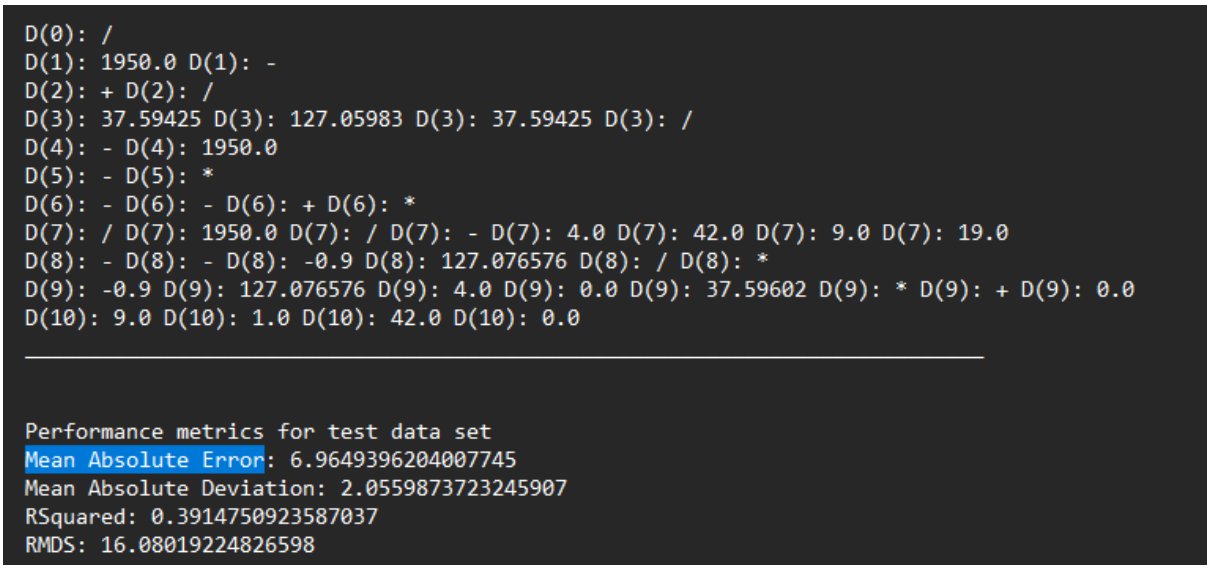
- A 150 000 subset was extracted from the original dataset and 70% was used for training and the remaining 30% was used for testing.
- This subset was extracted by selecting the first 150 000 items in the original dataset.

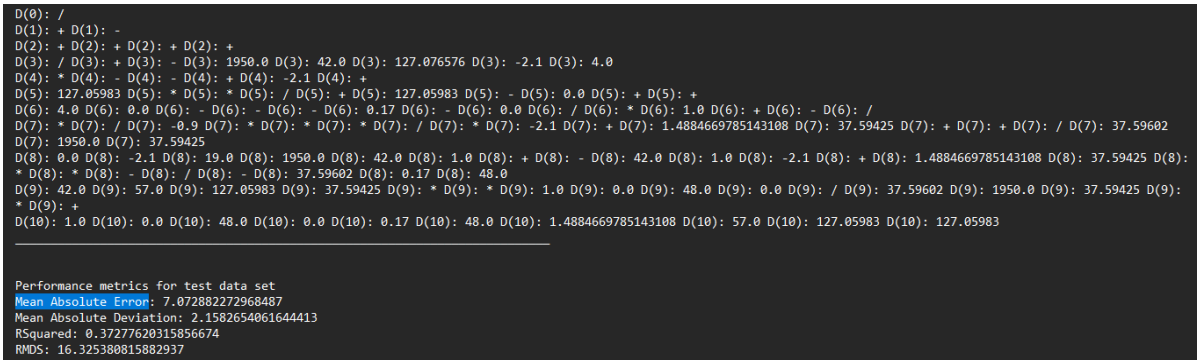
Solutions Found

- The below trees are printed out in a breadth first manner where D(x) represents the depth of a node with x an integer.
- Solution 1
    - Seed = 0
    - 

- Solution 2
    - Seed =1
    - 

- Solution 3
  - Seed =2
  - 

- Solution 4
  - Seed =3
  - 

- Solution 5
  - Seed =5
  - 

- Solution 6
  - Seed =7

```
D(0): -
D(1): 0.0 D(1): -
D(2): / D(2): 1.0
D(3): + D(3): +
D(4): + D(4): -2.1 D(4): / D(4): -
D(5): - D(5): / D(5): / D(5): 0.0 D(5): + D(5): 127.05983
D(6): 1950.0 D(6): -2.1 D(6): - D(6): * D(6): / D(6): 0.17 D(6): / D(6): -
D(7): * D(7): - D(7): 4.0 D(7): / D(7): - D(7): * D(7): 42.0 D(7): 19.0 D(7): * D(7): 127.05983
D(8): + D(8): - D(8): + D(8): / D(8): - D(8): / D(8): 19.0 D(8): 1.4884669785143108 D(8): / D(8): - D(8): 1.0 D(8): +
D(9): 57.0 D(9): 57.0 D(9): 37.59425 D(9): 0.0 D(9): 1.0 D(9): -2.1 D(9): 37.59425 D(9): -2.1 D(9): 1.3 D(9): 127.076576 D(9): 1.4884669785143108
D(9): 37.59602 D(9): 0.17 D(9): 4.0 D(9): 1950.0 D(9): -2.1 D(9): -0.9 D(9): 37.59425
_____


Performance metrics for test data set
Mean Absolute Error: 7.0977460285023435
Mean Absolute Deviation: 1.9188155492491297
RSquared: 0.38010603193390846
RMDS: 16.229710098302085
```

- Solution 7
  - Seed =8

```
D(0): +
D(1): / D(1): /
D(2): + D(2): + D(2): + D(2): *
D(3): - D(3): 42.0 D(3): * D(3): 37.59425 D(3): + D(3): / D(3): + D(3): +
D(4): 37.59425 D(4): * D(4): / D(4): 127.076576 D(4): 1950.0 D(4): 1.0 D(4): 1.4884669785143108 D(4): 48.0 D(4): / D(4): - D(4): + D(4): /
D(5): 1.0 D(5): 37.59602 D(5): 0.0 D(5): 1950.0 D(5): 127.05983 D(5): + D(5): + D(5): - D(5): / D(5): + D(5): + D(5): /
D(6): 37.59425 D(6): - D(6): 37.59425 D(6): 37.59602 D(6): * D(6): + D(6): + D(6): / D(6): / D(6): + D(6): 9.0 D(6): - D(6): +
D(7): 127.076576 D(7): 48.0 D(7): 19.0 D(7): 0.17 D(7): 1.4884669785143108 D(7): 19.0 D(7): 1.3 D(7): * D(7): 0.0 D(7): 9.0 D(7): * D(7): + D(7): -
D(7): 37.59602 D(7): 37.59425 D(7): / D(7): 48.0 D(7): 48.0 D(7): - D(7): -
D(8): 0.17 D(8): / D(8): 0.0 D(8): 48.0 D(8): 9.0 D(8): 37.59425 D(8): / D(8): 0.0 D(8): - D(8): / D(8): 9.0 D(8): 9.0 D(8): / D(8): 1950.0
D(9): 127.05983 D(9): 4.0 D(9): 42.0 D(9): 19.0 D(9): 0.17 D(9): 127.05983 D(9): 19.0 D(9): 37.59602 D(9): 127.05983 D(9): +
D(10): 19.0 D(10): 42.0
_____


Performance metrics for test data set
Mean Absolute Error: 7.125940812005493
Mean Absolute Deviation: 2.255328764438522
RSquared: 0.38093816039058126
RMDS: 16.218813284000397
```

- Solution 8
  - Seed =10

```
Generation: 49
Fitness: 1047149.9129437442
Num Nodes in Fittest Individual: 35
Best Depth: 0

D(0): +
D(1): 1.4884669785143108 D(1): -
D(2): / D(2): /
D(3): + D(3): 127.05983 D(3): * D(3): +
D(4): - D(4): 1950.0 D(4): / D(4): * D(4): - D(4): 1950.0
D(5): 19.0 D(5): 127.076576 D(5): / D(5): 42.0 D(5): / D(5): + D(5): * D(5): 4.0
D(6): 37.59602 D(6): + D(6): 42.0 D(6): 1.0 D(6): -0.9 D(6): 48.0 D(6): * D(6): 1.0
D(7): 0.0 D(7): 1.0 D(7): 19.0 D(7): 19.0
_____


Performance metrics for test data set
Mean Absolute Error: 9.739621392976419
Mean Absolute Deviation: 4.143123321669823
RSquared: 0.12607139456854022
RMDS: 19.270369165751156
```

- Solution 9
  - Seed =11

```
D(0): *
D(1): + D(1): 1.0
D(2): / D(2): 1.4884669785143108
D(3): + D(3): -
D(4): - D(4): + D(4): 37.59425 D(4): /
D(5): 37.59425 D(5): / D(5): + D(5): 127.076576 D(5): / D(5): *
D(6): * D(6): 127.076576 D(6): + D(6): * D(6): - D(6): / D(6): 0.0 D(6): +
D(7): - D(7): 9.0 D(7): - D(7): / D(7): 4.0 D(7): / D(7): 4.0 D(7): * D(7): * D(7): * D(7): 1.0 D(7): +
D(8): 42.0 D(8): + D(8): + D(8): 37.59425 D(8): / D(8): / D(8): + D(8): + D(8): / D(8): 1.0 D(8): 42.0 D(8): - D(8): 9.0 D(8): 4.0 D(8): * D(8):
37.59602
D(9): 1950.0 D(9): 42.0 D(9): / D(9): - D(9): / D(9): - D(9): 0.0 D(9): 19.0 D(9): 1950.0 D(9): 42.0 D(9): + D(9): 37.59425 D(9): / D(9): + D(9): +
D(9): * D(9): + D(9): -
D(10): 4.0 D(10): 127.076576 D(10): 9.0 D(10): 9.0 D(10): 48.0 D(10): 19.0 D(10): 48.0 D(10): 1.3 D(10): 37.59602 D(10): 42.0 D(10): -2.1 D(10):
57.0 D(10): 1950.0 D(10): 127.05983 D(10): 42.0 D(10): 0.0 D(10): 9.0 D(10): 19.0 D(10): 1950.0 D(10): 1.0 D(10): 57.0 D(10): 42.0
_________________________________________________________________


Performance metrics for test data set
Mean Absolute Error: 8.327357193099385
Mean Absolute Deviation: 3.590628805197939
RSquared: 0.3444209922611611
RMDS: 16.690316658021132
```

- Solution 10
  - Seed =15

```
D(0): +
D(1): + D(1): +
D(2): 1.4884669785143108 D(2): / D(2): / D(2): -
D(3): + D(3): / D(3): -2.1 D(3): * D(3): - D(3): /
D(4): / D(4): 4.0 D(4): + D(4): + D(4): + D(4): - D(4): / D(4): * D(4): - D(4): /
D(5): 1950.0 D(5): 127.05983 D(5): 37.59425 D(5): 1.4884669785143108 D(5): + D(5): 0.17 D(5): - D(5): 9.0 D(5): * D(5): 19.0 D(5): + D(5): / D(5): /
D(5): 1.3 D(5): 1.4884669785143108 D(5): 0.17 D(5): + D(5): -
D(6): 19.0 D(6): 1.0 D(6): * D(6): / D(6): - D(6): - D(6): / D(6): 4.0 D(6): + D(6): + D(6): / D(6): 37.59602 D(6): 48.0 D(6): 9.0 D(6): * D(6):
1.4884669785143108
D(7): 127.05983 D(7): 1950.0 D(7): 42.0 D(7): 48.0 D(7): + D(7): / D(7): 9.0 D(7): - D(7): 1950.0 D(7): 127.05983 D(7): 37.59425 D(7): 19.0 D(7):
9.0 D(7): / D(7): 4.0 D(7): 57.0 D(7): / D(7): -
D(8): 19.0 D(8): 1.0 D(8): + D(8): - D(8): 4.0 D(8): / D(8): + D(8): + D(8): 9.0 D(8): 42.0 D(8): 48.0 D(8): 4.0
D(9): 57.0 D(9): 57.0 D(9): 19.0 D(9): 19.0 D(9): * D(9): * D(9): 37.59425 D(9): 1.4884669785143108 D(9): + D(9): 1.0
D(10): 127.05983 D(10): 1950.0 D(10): 1.0 D(10): 19.0 D(10): 19.0 D(10): 1.0
_________________________________________________________________


Performance metrics for test data set
Mean Absolute Error: 10.230412924187343
Mean Absolute Deviation: 6.2776823209105475
RSquared: 0.2467593073206601
RMDS: 17.890354413953574
```

-

Summary of results

- Average value for MAE: 8.25
- Average value for MedAE: 3.31
- Average value for R^2: 0.312
- Average value for RMSE: 17.04
- Best value for MAE: 6.96
- Best value for MedAE: 1.91
- Best value for R^2: 0.40
- Best value for RMSE: 15.95
- From the results we see from R^2 that the genetic program is a bit below average(0.5) in terms of being able to predict the bike trip duration accurately.
- The small MedAE shows that the values produced by the genetic program are relatively close to each other and not spread out.
- A MAE of 8.25 shows that the predictions made by the genetic program are not very accurate but still decent enough to predict some of the results.
- Considering the average value of the RMSE we see that the genetic program is again struggling to predict very accurate but is still very close to the worst performing model in the paper.

Estimated Program Runtime and Running Program

- Note on running non compiled code: The Java project was created using VS Code with no build tools. It does not compile with the usual javac command. Instead, one must click the button on the top right(like triangle) and run it from there. Since there

are no build tools compile time is almost non-existent. Note you must open App.java before clicking the button. Also please make sure you open VS Code in the COS710_Assignments directory as the dataset won't be picked up by program otherwise.

- Java version: 17.0.3.7
- **Estimated execution time**: 30 minutes per run. Currently program is set to run only 1 run.
- Command to run jar file: java -jar COS710_Assignments.jar.

Comment and comparison of Results

- Comparing to the results in the paper we see that the Genetic Program evolved tends to perform better in the MAE category than GBM and KNN most of the time depending on the seed.
- However the Genetic program struggles with the R^2 performance measure to beat any of the 4 models shown in the paper. This shows that in terms of predicting accurate values the Genetic Program is the worst from the existing models.
- For the MedAE we see that the Genetic program does exceptionally well compared to the other 4 models.
- For the RMSE performance measure we see that the Genetic Program tends to perform just a bit worse than the LR model. However, since the RMSE of the Genetic Program is not exceptionally high it does not produce excessively large errors when predicting.
- Overall we consider this genetic program to be just slightly better than the LR training model presented in the paper, but worse than all the other 3 training models used.
- Finally, it is worth stating that the performance of the genetic program could potentially be significantly improved if more processing power was available to explore deeper trees, a larger number of generations and larger populations, something which on the current system would just take too long to run.

System specifications of computer

- AMD Ryzen 3 3200U
- 16GB RAM
- 1TB HDD
- Dual core