Training a Machine-Learning Model for Optimal Fitness Function Selection with the Aim of Finding Bugs

Toon Kling
t.kling@student.tudelft.nl

Supervisors: Pouria Derakhshanfar & Mitchell Olsthoorn
Responsible professor: Annibale Panichella

1. Background

- Automated Testing using Evosuite
- Optimizing criteria: Branch coverage, Mutation score
- Implemented as *Coverage criteria:*
    - Branch coverage / BRANCH
    - Input coverage / BRANCH;INPUT
    - Exception Coverage / BRANCH;EXCEPTION
- Can we predict when Branch & Exception outperform Branch?

2. Research Question

When and how does Exception Coverage increase the number of bugs detected when combined with branch coverage?
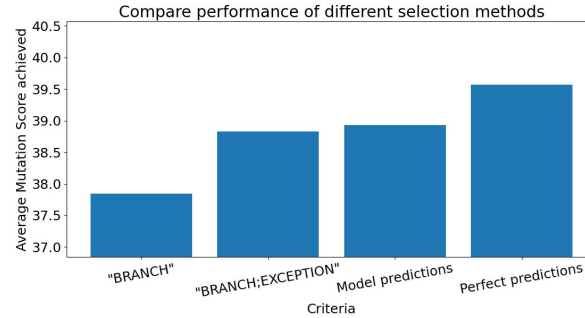
3. Method and Process

Dataset:
- SF110 + Apache Commons: 113 Java projects
- 10 runs per class for 3 time units

Metrics collected: Chidamber & Kemerer extended
Predict: Branch coverage / Mutation score improvement
Model used: Random Forest Classifier

Figure 1: Mutation Score achieved by Model

4. Results

- F1-score achieved (Mutation Score): 0.865
- Criteria selection with model outperforms other criteria 8 / 10 times
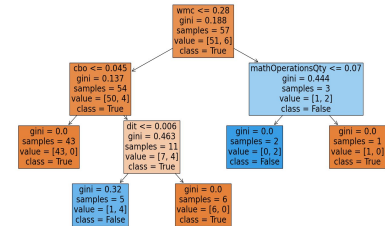

Figure 2: Example of Decision Tree

5. Conclusions

- BRANCH;EXCEPTION often performs better than BRANCH.
- Random Forest Model can predict when each criterion achieves a better Mutation Score, not Branch Coverage
- Explanatory AI