1 Context & Motivation

Explanation and interpretability are crucial features required from machine learning (ML) users to increase their confidence in the models [1]. Indeed, the General Data Protection Regulation (GDPR) introduced them the option to demand explanations of automated-decisions.

Interpretability in this case means the ability to explain the properties of a model employing understandable terms to a human in the function of only some particular data and model outputs [2]. An explanation, on the other hand, comprises a set of self-contained features from the interpretable domain which leads a given input to produce a specific decision, without requiring any further explanation [3, 4, 5]. Model interpretability can help the users to detect and correct bias in the training data, to point out potential data perturbation that might lead to change a model output, to ensure that only the correct features influence the output, and to understand the underlying phenomenon such as in physics and social sciences [6, 7].

On the one hand, state-of-the-art ML methods can achieve high accuracy performance. Therefore, it usually hard to understand the mechanisms by which they work due to their huge parameter space. These methods are considered black-box models [5]. On the other hand, some machine learning models can offer high interpretability, but with a drop in their predictive performance. Clearly, there is a trade-off choice between accuracy, explanation, and interpretability of ML models [8, 9].

In contrast, genetic programming (GP), an evolutionary computation (EC) technique, follows a bio-inspired strategy by which a population of solutions representing models is evolved during a determined number of generations. A fitness function is then used to evaluate the performance of the models and to probabilistically select them to go through mutation and crossover operations. Evolutionary computing-based techniques can work without requiring beforehand from users the structure of the solution. Moreover, they are domain-independent and can solve problems by starting from high-level statements. Furthermore, evolutionary computing-based methods can be combined with other heuristic techniques (e.g., local search) to improve the quality of the solutions [10] by taking into account the semantic of each generated solution [11, 12]. Finally, genetic programming evolved models are normally interpretable by humans [13].

2 Goal

The aim of this internship is to work on one of the following topics.

1. **Post-hoc explanation through evolutionary computing-based techniques**: explore how evolutionary computing-based methods can be used to provide both model agnostic post-hoc explanation and local explanation for the prediction of black-box models.

2. **Mixed-integer linear programming (MILP) and genetic programming**: investigate the use of MILP as a way to guide evolutionary computing methods when searching the optimal solutions, as well as to improve their explainability.

3. **Genetic programming and reinforcement learning**: investigate how domain knowledge can be used to reward an agent according to the function or operators it selects to evolve a set of GP models.

4. **Federated learning and genetic programming**: investigate how the concept of federated learning can be explored by evolutionary computing techniques.

3 Profile

The internship requires excellent machine learning skills, the ability to work on cross-disciplinary problems, and good programming experience, preferably in Python.
References


