Hyper-heuristic proposal

Firefighter problem

A01169881
A01222528

Description

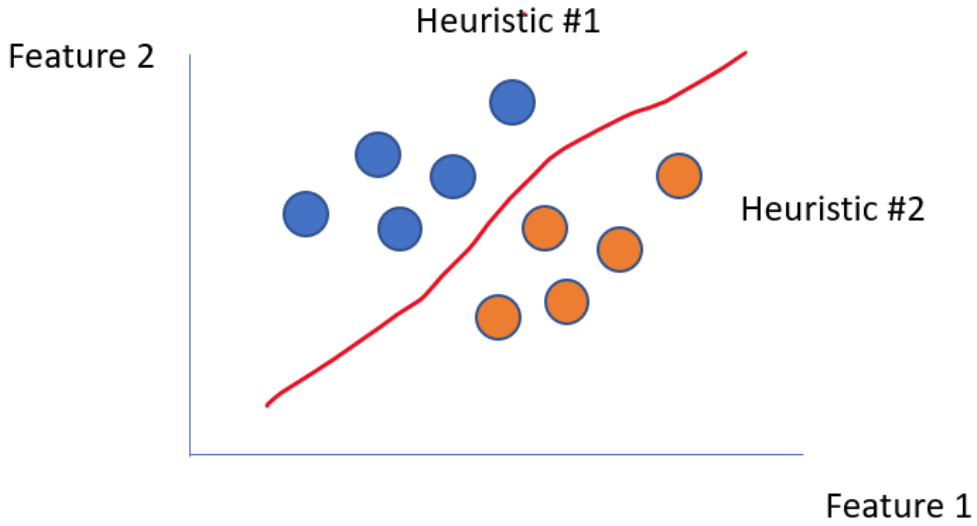
- An instance of the firefighter problem “**P**” is characterized by a set of features “**F**”.
- Given an initial condition “**I**” the problem evolves into “**P’**” after a single timestep, and the fire starts spreading in the network.
- However, a firefighter can protect a given node by taking an action “**A**” which will influence the state of the network at the next timestep.
- The action of the firefighter is decided by applying a heuristic “**H**” at each timestep.
- We can assess how good a given decision “**A**” is based on the status of the network after applying this decision. This is measured using a performance index “**S**”.
- The heuristic to apply is given by a hyper-heuristic “**HH**” based on a classifier.

Classifier

Proposal

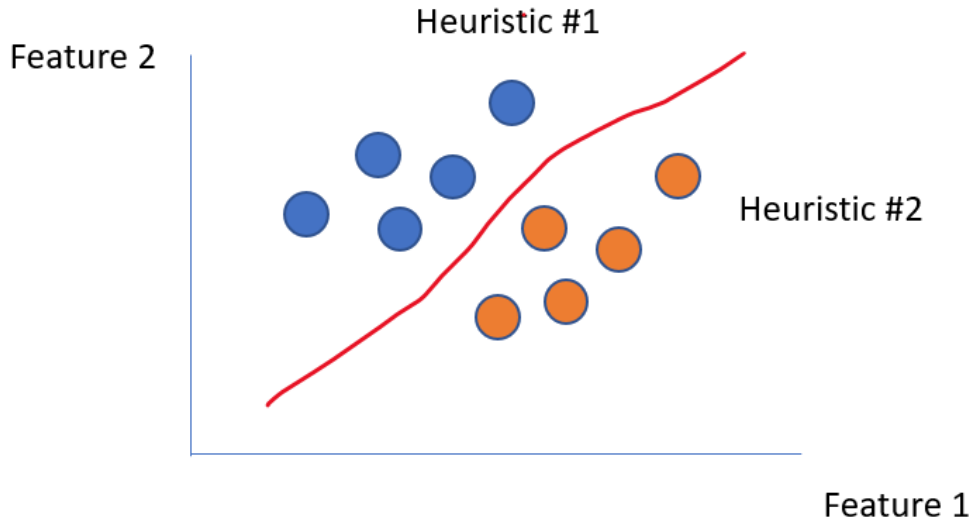
- Create a classifier based on the set of features “**F**” to choose the heuristic “**H**” that maximizes the performance index “**S**”.
- There exist multiple alternatives for classifiers. We would like to try different approaches. E.g., KNN, Neural networks, SVM, etc.

Problem representation



Training: Assigning a class

Problem representation



How to assign a class?

- The class corresponds to the heuristic that maximizes/minimizes a given performance index "**S**". That is:

$$\mathrm{HH} = \mathrm{argmin}(S(H_1), S(H_2), \ldots, S(H_n))$$

- Simulations need to be run on each problem instance for each of the heuristics to determine the class.

Performance index

- There can be different variants of the performance index depending on what we are interested in minimizing/maximizing. We would like to try with different performance indexes.
- One performance index we propose is to count the number of nodes that are on fire (on average) after K iterations of the problem given each of the heuristics. For this purpose, we perform simulations k steps into the future after using a given heuristic "**H**" for "**P**".

That is:

$$Avg\ nf(k) = \frac{1}{k}\sum_{i=1}^{k} nf(i)$$

- Where nf is the number of nodes on fire in the i^{th} iteration of the problem.