# Evolutionary Algorithm for HPO and NAS on the Microflower dataset

AutoML Final Project

Ben Bausch

14.9.2020

# Approach Overview

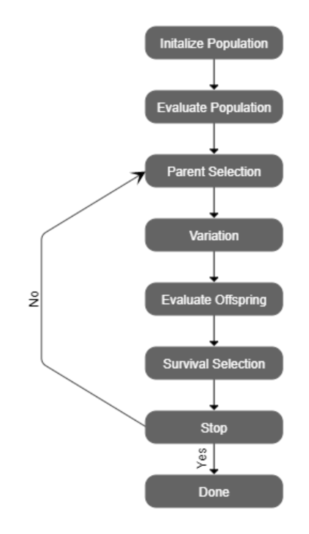



**Figure from** [1]

**Motivation :**

- relatively cheap evaluations for bad performing NNs due to early stopping and small image resolution, helps with low sample efficiency
- idea of combining objectives in tournament selection
- idea of using distribution proportional to Hypervolume of models

# Approach - Parent Selection and Population Initialization

**Population Initialization :**

- 5 models in initialization : default model, largest/smallest possible model in CS, 2 random models

**Parent selection :**

- using 1) Tournament and 2) Fitness-proportional selection
- 1) tournament group sampled by distribution proportional to size then best performing config chosen
- 2) distribution proportional to the hyper volume of different models
- use 1) every third generation otherwise use 2)
- using 1) instead of only 2) keeps the population of models more diverse

# APPROACH - VARIATION

**Producing Offspring using :**

- Simulated Binary Crossover (SBX) e.g. optimizer
- Gaussian recombination e.g. number filters in convolution
- Uniform sampling e.g number fc layers
- Sampling according to prior knowledge e.g. higher prob for using batch-normalization

# Approach - Survival Selection and Training

**NSGA-II**

- uses $(\mu + \lambda)$ strategy
- Survival Selection ranks candidates by non dominated sorting
- no tie breaker, simply use max number of possible survivors, if reach reach stop selection

**Training**

- 3 fold-cross-validation
- early stopping with patience of 10 epochs (filters out really bad and early converged models)
- at most 50 epochs

# Evaluation

**Metrics**

- performance : 3-fold crossvalidated top-3 accuracy
- size : number of model parameters

**Experimental Setup :**

- runtime for the results +- 3 hours
- Trained on single GPU : Nvidia RTX 2080 Super

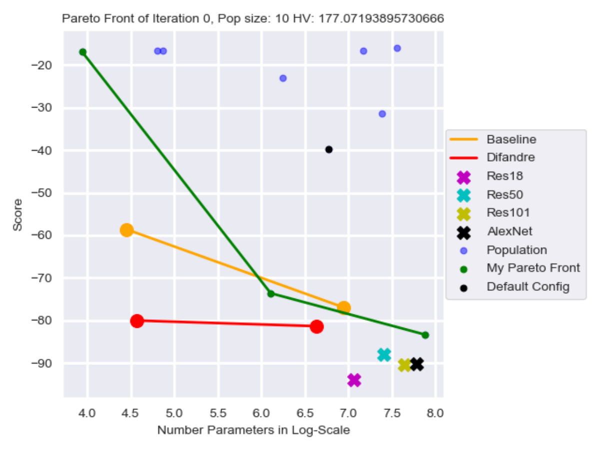
**Reproducibility :**

- every generation step saved and can be reloaded
- every model has been save and can be reloaded
- population of each generation can be loaded by loading the corresponding models
- models currently not available on Github due to size but can be provided, necessary code is available

# EVALUATION CONT'D

Statistics about the explored search space can be plotted and show its coverage by the algorithm.

## Population over time :



**Results :**

- Models found with better perfomance and smaller size than default model
- Difandre found better models of smaller size, but overall both baselines have been outperformed based on the hypervolume
- found models similar to ResNet50 and AlexNet in terms of both metrics

# References

**Thanks for your attention!**


Ben Bausch.

Automated Machine Learning University Freiburg and Hannover.