Neuroevolution under Unimodal Error Landscapes: An Exploration of the Semantic Learning Machine Algorithm

Jan-Benedikt Jagusch
NOVA IMS, Universidade Nova de Lisboa
Lisboa, Portugal
m2016022@novaims.unl.pt

Ivo Gonçalves
INESC Coimbra, DEEC, University of Coimbra
Coimbra, Portugal
icpg@dei.uc.pt

Mauro Castelli
NOVA IMS, Universidade Nova de Lisboa
Lisboa, Portugal
mcastelli@novaims.unl.pt

ABSTRACT

Neuroevolution is a field in which evolutionary algorithms are applied with the goal of evolving Neural Networks (NNs). This paper studies different variants of the Semantic Learning Machine (SLM) algorithm, a recently proposed supervised learning neuroevolution method. Perhaps the most interesting characteristic of SLM is that it searches over unimodal error landscapes in any supervised learning problem where the error is measured as a distance to the known targets. SLM is compared with the NeuroEvolution of Augmenting Topologies (NEAT) algorithm and with a fixed-topology neuroevolution approach. Experiments are performed on a total of 9 real-world regression and classification datasets. The results show that the best SLM variants generally outperform the other neuroevolution approaches in terms of generalization achieved, while also being more efficient in learning the training data. The best SLM variants also outperform the common NN backpropagation-based approach under different topologies. The most efficient SLM variant used in combination with a recently proposed semantic stopping criterion is capable of evolving competitive neural networks in a few seconds on the vast majority of the datasets considered. A final comparison shows that a NN ensemble built with SLM is able to outperform the Random Forest algorithm in two classification datasets.

CCS CONCEPTS

• **Computing methodologies** → **Machine learning approaches**; **Machine learning algorithms**; *Artificial intelligence*; *Bio-inspired approaches*;

KEYWORDS

Semantic Learning Machine, NEAT, Neuroevolution, MLP

ACM Reference Format:
Jan-Benedikt Jagusch, Ivo Gonçalves, and Mauro Castelli. 2018. Neuroevolution under Unimodal Error Landscapes: An Exploration of the Semantic Learning Machine Algorithm. In *GECCO '18 Companion: Genetic and Evolutionary Computation Conference Companion, July 15–19, 2018, Kyoto, Japan.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3205651.3205778

GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5764-7/18/07…$15.00
https://doi.org/10.1145/3205651.3205778

1 SEMANTIC LEARNING MACHINE

The Semantic Learning Machine (SLM) [2, 4] is a stochastic neural network construction algorithm originally derived from Geometric Semantic Genetic Programming (GSGP) [5]. Perhaps the most interesting characteristic of SLM is that it searches over unimodal error landscapes in any supervised learning problem where the error is measured as a distance to the known targets. The unimodal error landscape implies that there are no local optima. This means that, with the exception of the global optimum, every point in the search space has at least one neighbor with better fitness, and that neighbor is reachable through the application of the variation operators. As this type of landscape eliminates the local optima issue, it is potentially much more favorable in terms of search effectiveness and efficiency. The SLM semantic mutation operator allows to search over the space of neural networks without the need to use backpropagation to adjust the weights of the network. As the issue of local optima does not apply, the evolutionary search can be performed by simply hill climbing. The SLM mutation operator was originally specified for neural networks with a single hidden layer [2], but it was subsequently extended to be applicable to any number of hidden layers [4]. For further details the reader is referred to [2] and [4].

2 EXPERIMENTAL METHODOLOGY

SLM is compared with the NeuroEvolution of Augmenting Topologies (NEAT) [6] algorithm and with a fixed-topology neuroevolution approach. Besides these neuroevolution methods, the common fixed-topology backpropagation Multilayer Perceptron (MLP) is also tested, as well as the Support Vector Machines (SVM) algorithm. Parameter tuning is performed for all methods according to some of their specific parameters. A total of 30 runs are performed for each method. At each run, 30% of the available data is left out as unseen data in order to compute the generalization error after the parameter tuning is completed. From the 70% of the data that are used for the parameter tuning, an additional partition is performed: 70% for training and 30% for validation. For each method, the best parameter combination is selected according to the validation data performance. The best parameter combination of each method is subsequently assessed on the unseen data. This assessment yields the generalization error of a given method on a given run. From these 30 generalization errors, statistical tests can be performed to assess the significance of the results. Errors are computed as the root mean squared error.

Three main SLM variants are studied: one that uses a fixed learning step (FLS) throughout the run, and two others that compute the

optimal learning step (OLS) for each application of the mutation operator. This optimal learning step computation is equivalent to the optimal mutation step computation in GSGP [1]. The SLM-FLS variant is studied under a fixed learning step of 0.01 (no parameter tuning is performed). For SLM-OLS, the performance with and without the Error Deviation Variation (EDV) semantic stopping criterion [3] is also assessed. This semantic criterion uses information from the semantic neighborhood to decide when to stop the search before overfitting starts to occur. For all SLM variants, the mutation operator adds one neuron to each hidden layer. Each added neuron connects to a randomly selected subset of neurons from the previous layer. The SLM variants are tested with 1, 2, and 3 hidden layers. All hidden neurons have the hyperbolic tangent as their activation function.

The NEAT parameters are taken from [6]. The following combination of parameters are tuned: the adding-node-mutation (ANM) probability, the adding-link-mutation (ALM) probability, and the link weight coefficient (LWC). The fixed-topology neuroevolution approach assessed is a simple genetic algorithm that optimizes the weights of a given NN. This method is referred to as fixed-topology neuroevolution (FTNE). For FTNE, the following configuration is used: tournament selection of size 5; a whole arithmetic crossover; a Gaussian mutation operator with standard deviation of 0.01; a crossover rate of 0.05; and a mutation rate of 1. For the hidden topology the following topologies are assessed: one hidden layer with one hidden neuron, one hidden layer with two hidden neurons, two hidden layers with two hidden neurons each, and three hidden layers with five hidden neurons each. All three neuroevolution approaches use the same total number of evaluations per run: 500 generations are conducted, with 100 individuals being created at each generation. For the MLP, parameter tuning is performed over the learning rate, momentum, number of epochs, and hidden topology. In the SVM tuning, different complexity parameters are tested, as well as different kernels.

A total of 9 well-known real-world regression and classification datasets are considered in the experiments. The four classification datasets used are: Wisconsin Diagnostic Breast Cancer (Cancer); German Credit Data (Credit); Pima Indians Diabetes (Diabetes); and Connectionist Bench (Sonar). The five regression datasets used are: Bioavailability (Bio); Toxicity (LD50); Parkinson Speech (Parkinson); Plasma Protein Binding levels (PPB); and Student Performance (Student). Statistical significance is assessed with Mann-Whitney U tests, using a Bonferroni correction, and considering a significance level of $\alpha = 0.05$.

3 MAIN RESULTS

Table 1 presents the p-values of the generalization error comparisons between the best SLM variant and the other methods considered after parameter tuning is performed. The best SLM variants achieve superior generalizations in two thirds of the comparisons (24 out of 36). In the remaining comparisons, no method is found to be superior to the best SLM variant. With respect to the training data performance, the best SLM variant is always superior, with statistical significance, to MLP and the other neuroevolution methods (all p-values below 2.77×10^{-5}).

Table 1: p-values are presented for the comparisons where the SLM variant is superior, with statistical significance, to the opposing method. The comparisons where no statistically significant differences are found are left blank.

Dataset	MLP	NEAT	FTNE	SVM
Cancer	1.51×10^{-6}	9.13×10^{-7}	1.59×10^{-4}	1.01×10^{-6}
Credit	9.13×10^{-7}	2.58×10^{-3}		9.13×10^{-7}
Diabetes	9.13×10^{-7}			9.13×10^{-7}
Sonar		1.13×10^{-5}		
Bio	5.37×10^{-6}	9.13×10^{-7}	2.48×10^{-6}	
LD50		9.13×10^{-7}		
Parkinson	9.13×10^{-7}	1.01×10^{-6}	9.13×10^{-7}	1.13×10^{-5}
PPB	7.13×10^{-6}	9.13×10^{-7}	1.26×10^{-4}	
Student	1.03×10^{-5}	2.54×10^{-4}	4.27×10^{-5}	

Table 2: Median number of hidden neurons and computational time in seconds of SLM-OLS + EDV and NEAT in a selected subset of datasets

Number of hidden neurons		
Dataset	SLM-OLS + EDV	NEAT
Cancer	9	144
Diabetes	7	104
Sonar	6	131
Bio	8	131
Student	21	106
Computational time in seconds		
Dataset	SLM-OLS + EDV	NEAT
Cancer	0.08	162.68
Diabetes	0.55	310.81
Sonar	0.11	207.61
Bio	1.16	491.06
Student	0.82	320.18

In a closer comparison with NEAT, the most efficient SLM variant (SLM-OLS + EDV) is able to evolve significantly smaller NNs, while also requiring a significantly smaller computational effort (table 2).

In a final assessment, an ensemble of NNs built with SLM-OLS + EDV is compared against the Random Forest algorithm. The resulting ensembles are compared with the same number of elements (30). In a comparison in the Cancer and the Diabetes datasets, SLM is able to outperform Random Forest with statistical significance: Cancer (p-value 1.011×10^{-6}), Diabetes (p-value 9.127×10^{-7}).

ACKNOWLEDGMENTS

This work was also financed through the Regional Operational Programme CENTRO2020 within the scope of the project CENTRO-01-0145-FEDER-000006.

REFERENCES

[1] Ivo Gonçalves, Sara Silva, and Carlos M Fonseca. 2015. On the generalization ability of geometric semantic genetic programming. In *Genetic Programming*. Springer, 41–52.

[2] Ivo Gonçalves, Sara Silva, and Carlos M. Fonseca. 2015. Semantic Learning Machine: A Feedforward Neural Network Construction Algorithm Inspired by Geometric Semantic Genetic Programming. In *Progress in Artificial Intelligence*. Lecture Notes in Computer Science, Vol. 9273. Springer, 280–285.

[3] Ivo Gonçalves, Sara Silva, Carlos M Fonseca, and Mauro Castelli. 2017. Unsure when to stop? ask your semantic neighbors. In *Proceedings of the Genetic and Evolutionary Computation Conference*. ACM, 929–936.

[4] Ivo Gonçalves. 2017. *An Exploration of Generalization and Overfitting in Genetic Programming: Standard and Geometric Semantic Approaches*. Ph.D. Dissertation. Department of Informatics Engineering, University of Coimbra, Portugal.

[5] Alberto Moraglio, Krzysztof Krawiec, and Colin G Johnson. 2012. Geometric semantic genetic programming. In *International Conference on Parallel Problem Solving from Nature*. Springer, 21–31.

[6] Kenneth O Stanley and Risto Miikkulainen. 2002. Evolving neural networks through augmenting topologies. *Evolutionary computation* 10, 2 (2002), 99–127.