Using Restricted Boltzmann Machines as the model of an estimation of distribution algorithm

1 Description

Estimation of Distribution Algorithms (EDAs, [1, 2, 3]) are optimization algorithms that use probabilistic models to capture the most relevant features of the solutions with higher values. EDAs have been applied to a variety of optimization problems from domains such as Bioinformatics [4], Energy [5, 6], vehicle [7] design, etc.

Restricted Boltzmann Machines (RBMs) are probabilistic generative neural networks that use an energy model to define the probability of the states. They have been proposed as an alternative to classical probabilistic models in the context of EDAs [8, 9].

2 Objectives

The goal of the project is to implement in Python an estimation of distribution algorithm that uses an RBM. The work presented in [8, 9] can be taken as a reference of how to insert RMBs in EDAs. The project should: 1) Implement the union of the RBM learning and inference procedures with the EDA. 2) Test the implemented algorithm in the optimization of some functions (see [10] for examples of functions). 3) Answer to the following questions in the report:

- What class of problems can be solved with the NN? (e.g., supervised vs unsupervised problems)
- What is the network architecture? (e.g., type and number of layers, parameters, connectivity, etc.).
- What is the rationale behind the conception of the NN?
- How is inference implemented? (e.g., How is the information extracted from the network?). Type of prediction or type of inference process.
- What are the learning methods used to learn the network ? Algorithms used for learning the network.

As in other projects, a report should describe the characteristics of the design, implementation, and results. A Jupyter notebook should include calls to the implemented function that illustrate the way it works.

3 Suggestions

- The *DEAP* library `https://code.google.com/p/deap/` could be used as a basis to create the EDA. One EDA (UMDA) is implemented as part of this library. Different implementations of RBMs in Python are available, including sklearn. Therefore it is possible to combine these implementations to complete the project.

References

[1] H. Mühlenbein. The equation for response to selection and its use for prediction. *Evolutionary Computation*, 5(3):303–346, 1997.

[2] P. Larrañaga and J. A. Lozano, editors. *Estimation of Distribution Algorithms. A New Tool for Evolutionary Computation*. Kluwer Academic Publishers, Boston/Dordrecht/London, 2002.

[3] J. A. Lozano, P. Larrañaga, I. Inza, and E. Bengoetxea, editors. *Towards a New Evolutionary Computation: Advances on Estimation of Distribution Algorithms*. Springer, 2006.

[4] R. Armañanzas, I. Inza, R. Santana, Y. Saeys, J. L. Flores, J. A. Lozano, Y. Van de Peer, R. Blanco, V. Robles, C. Bielza, and P. Larrañaga. A review of estimation of distribution algorithms in bioinformatics. *BioData Mining*, 1(6):doi:10.1186/1756–0381–1–6, 2008.

[5] Márcio Henrique da Silva and Roberto Schirru. A self-adaptive quantum PBIL method for the nuclear reload optimization. *Progress in Nuclear Energy*, 74:103–109, 2014.

[6] Komla Folly. Parallel PBIL applied to power system controller design. *Journal of Artificial Intelligence and Soft Computing Research*, 3(3):215–223, 2013.

[7] Wencong Su and Mo-Yuen Chow. Performance evaluation of an EDA-based large-scale plug-in hybrid electric vehicle charging algorithm. *Smart Grid, IEEE Transactions on*, 3(1):308–315, 2012.

[8] Vui Ann Shim, Kay Chen Tan, Chun Yew Cheong, and Jun Yong Chia. Enhancing the scalability of multi-objective optimization via restricted Boltzmann machine-based estimation of distribution algorithm. *Information Sciences*, 248:191–213, 2013.

[9] Malte Probst, Franz Rothlauf, and Jörn Grahl. Scalability of using restricted Boltzmann machines for combinatorial optimization. *European Journal of Operational Research*, 256(2):368–383, 2017.

[10] Malte Probst. Generative adversarial networks in estimation of distribution algorithms for combinatorial optimization. *CoRR*, abs/1509.09235, 2015.