An Analysis of Particle Swarm Optimization

Samrat Banerjee and Shripad Tak, MSc Computer Applications, Symbiosis Institute of Computer Studies & Research,Pune

METHODOLOGY

There is a difficulty in simple functions with single highest value (unimodal), where regrouping is unnecessary since particles quickly move towards a suboptimal solution which affects the accuracy of the final result solution from some supervised learning, unsupervised learning and reinforcement learning algorithms. To reduce this difficulty to the minimum, we seeked some solutions from some Supervised learning, Unsupervised learning and Reinforcement learning algorithm. A family of other learning algorithms, other than Swarm Intelligence from which PSO originates will be taken into consideration in its improvement. There will be a concerted effort to improve the PSO algorithm by extensive extreme programming techniques and advantageous features of other clustering techniques in data mining will be taken to make the algorithm more efficient and inexpensive.

Supervised learning algorithms like, Random Forest which gives much more accurate outcomes / predictions than regression models. The Random forest algorithm will take the final probability from average probability of all the attributes of the data particles.

KNN - (K Nearest Neighbor) where we have labelled data particles in several groups or clusters where in all the items in a group are homogenous. A unlabelled data particle can be classified by calculating the nearest neighboring cluster.

Unsupervised learning algorithms like K means will pick out k number of data particles as centroids and every other data particle will form a cluster with the centroid closest to that respective particle.

Reinforcement learning algorithms like Markov Decision Process where the data particle learns to react to its environment. The data particle will take an action to transit between the starting state and the end state (clusters) and compare its rewards, after which it will join the cluster from which it receives the highest reward value.

Extreme programming using Python will be implemented to enhance the quality of the algorithm by testing it parallel to development(Test Driven Development). Extreme programming includes writing the test cases(PyUnit) before writing the code so as to ensure that the code is most efficient and computationally inexpensive.

REFERENCES

[1] R. Eberhart, and J. Kennedy, (1995) A New Optimizer Using Particles Swarm Theory, Proc. Sixth International Symposium on Micro Machine and Human Science (Nagoya, Japan), IEEE Service Center, Piscataway, NJ, pp. 39-43.

[2] Amreen Khan, Prof. Dr. N.G.Bawane , Prof. Sonali Bodkhe, An Analysis of Particle Swarm Optimization with Data Clustering-Technique for Optimization in Data Mining.(IJCSE) International Journal on Computer Science and Engineering
Vol. 02, No. 07, 2010, 2223-2226

[3] J. Kennedy, and R Eberhart, (1995), Particle Swarm Optimization, IEEE Conference on Neural Networks, pp. 1942-1948, (Perth, Australia), Piscataway, NJ, IV, 1995.

[4] A. P. Engelbrecht. (2005), Fundamentals of Computational Swarm Intelligence. Wiley, 2005.

[5] Kavitha Sooda, T. R. Gopalakrishnan Nair A Comparative Analysis for Determining the Optimal Path using Particle Swarm Optimization and Genetic Algorithms, International Journal of Computer Applications (0975 – 8887) Volume 32– No.4, October 2011

[6] Ming Li, Wenqiang Du, and Fuzhong Nian, An Adaptive Particle Swarm Optimization Algorithm Based on Directed Weighted Complex Network, School of Computer and Communication, LanZhou University of Technology, Lanzhou 730050, China, 2 April 2014

[7] Riccardo Poli,Analysis of the Publications on the Applications of Particle Swarm Optimisation, Hindawi Publishing Corporation Journal of Artificial Evolution and Applications Volume 2008, Article ID 685175, 10 pages doi:10.1155/2008/685175 30 November, 2007

[8] Particle Swarm Optimization, Edited by Aleksandar Lazinica p. cm. ISBN 978-953-7619-48-0 1. Particle Swarm Optimization I. Aleksandar Lazinica