Journal of College of Education for pure sciences(JCEPS)
Web Site: http://eps.utq.edu.iq/ Email: com@eps.utq.edu.iq
Volume 7, Number 4, December 2017

Using Quantum Particle Swarm Optimization to Enhance K-Means Clustering

Firas Sabar Miften

firassebar@yahoo.com

Computer Science, Faculty of Education for Pure Sciences,

University of Thi-Qar, Iraq

Abstract

Clustering isan unsupervised data mining technique used to extract a new knowledge. It clusters a group of objects into clusters where objects in one cluster have similar fea- tures to each other and have different features from objects in other clusters. K-means algorithm creates clusters by divides the data points into clusters according to similarity criterion. The K-means algorithm select initial centroids randomly then slow convergence points to centroids. This paper suggests a method for computing the initial centroids and fast convergence by using Quantum Particle Swarm Optimization with the global searching optimization which will give algorithm more efficient, so as to get quality clustering with reducedcomplexity.

Keywords*:* k-means cluster algorithm, Particle Swarm Optimization, Clustering, Quantum

1. Introduction

Cluster analysis is one of data mining techniques used to find out a new knowledge by analyzing data. This can be achieved by grouping the objects or elements into clusters where theobjectsinonecluster have somecommonfeaturestoeachotherwhiletheyhave different features from those of other clusters. In addition, the clustering algorithms are categorized into five main approaches as presented in Figure 1 [1]. The simplest clustering algorithm for unsupervised learning is K-means algorithm. The algorithm requires input matrix of points and K initial cluster centrist use Euclidean distance between points and clusters. The algorithm starts to search for K groups with locally optimal intra-cluster by changing points from one cluster to another [2]. K-means algorithm uses the gradient descent to converge to the local optimal searching. Therefore, Quantum Particle Swarm Optimization is used to find global searching. The particle swarm optimization (PSO) is focused gradually, which is one of the commonly used evolutionary algorithms [3]. The PSO algorithm is an optimization method. Fundamentally, it uses stochastic procedures and uses the theory of fitness similar to the negative of the cost of Genetic Algorithm (GA).

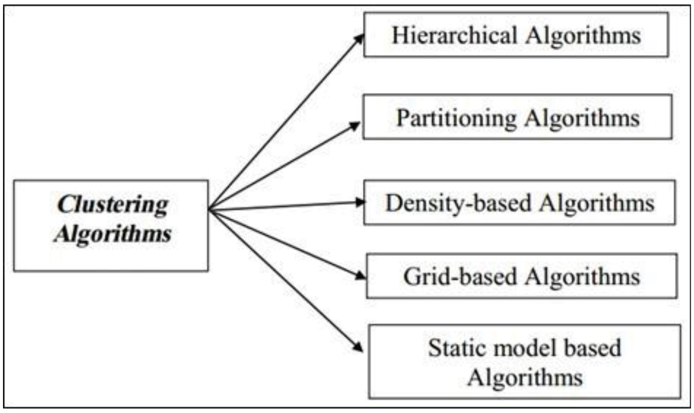

Figure 1: Data Mining Clustering Types.

A quantum is version of the PSO algorithm. It was proposed recently [4]. All particles in the quantum particle swarm optimization have a quantum behavior rather than the traditional Newtonian that was used in the PSO. Therefore, as an alternative of the Newtonian random walk, quantum motion is used in the search procedure. When a set of benchmarking functions is tested against the QPSO, the QPSO showed higher performance when compered by performance of the classical PSO. The QPSO needs large population sizes. The reduced number of parameters is one of the best features of QPSO algorithm. The QPSO only one parameter required.

1.1. The Classical PSO

It

will be very instructive to review first the basics of the PSO method in order to introduce the quantum version. In the Standard PSO model, each individual is treated as a volume-lessparticleintheD-dimensionalspace,withthepositionandvelocityof*ith*particle represented as:

$$V_i\,(t+1) = w \times V_i\,(t) + c_1{\times}random{\times}(P_i - X_i\,(t)){+}c_2{\times}random{\times}(P_g{-}X_i\,(t) \qquad (1)$$

$$X_i(t+1)) = X_i\,(t) + V_i\,(t+1) \qquad (2)$$

where

- c_1 *and* c_2 are positiveconstant.
- random is a random number in the range of [0,1].
- Parameter w is the inertia weight introduced to accelerate the convergence speed of thePSO.
- The vector $pi = (pi1, pi2, pi3, \ldots, pid)$ is the best previous position (the posi- tion giving the best fitness value) of particle i called pbest, and the vector $pg = (pg1, pg2, pg3, \ldots, pgd)$ is the position of the best particle among all the particles in the population and called gbest.

The steps involved here is the population size is first determined, and the velocity and position of each particle are initialized. Each particle moves according to [5] , and the fitness is then calculated. Meanwhile, the best positions of each swarm and particles are recorded. Finally, as the stopping criterion is satisfied, the best position of the swarm is the final solution. The main steps are given asfollows:[3]

1. Set the swarm size. Initialize the velocity and the position of each particle randomly.
2. For each i, evaluate the fitness value of xi and update the individual best position P_i , if better fitness is found.
3. Find the new best position of the whole swarm. Update the swarm best position xi. if the fitness of the new best position is better than that of the previous swarm.
4. If the stopping criterion is satisfied, then stop.
5. For each particle, update the position and the velocity according (1) and (2). Go to step2.

1.2. Quantum Particle Swarm Optimization

All particles in the QPSO algorithm move under quantum manner rules rather than the classical Newtonian random. In the classical PSO algorithm, all particles are moving to the optimum location. The particles are then attracted to location of optimum through the optimization process. The global optimum is led by such attraction.

In equation (3) see nothing then a random average of the local and global bests of the swarm particles. In quantum algorithm.

In Quantum Particle Swarm Optimization,uses the following equationto moves the particle according to:

$$mbest = \frac{1}{M}\sum_{i=1}^{M} P_i = \left(\frac{1}{M}\sum_{i=1}^{M} P_{i1},\ \frac{1}{M}\sum_{i=1}^{M} P_{i2},\ \cdots,\ \frac{1}{M}\sum_{i=1}^{M} P_{id}\right) \quad (3)$$

$$p_{id} = \varphi * P_{id} + (1-\varphi) * P_{gd}, \quad \varphi = rand\,() \quad (4)$$

$$X_{id} = p_{id} \pm \alpha * |mbest_d - X_{id}| * \ln(1/u), \quad u = Rand() \quad (5)$$

Where

- *mbest* is the mean best position among the particles.
- P ED a stochastic point between id 2ED and 2C@ is the local attractor on the *dth* dimension of the *ith* particle.
- Φ and u are a random umber distributed uniformly on [0,1].
- A is a parameter of QPSO that is called Contraction-Expansion Coefficient.

The

- A is Contraction Expansion Coefficient, a parameter of QPSO.

The Quantum Particle Swarm Optimization algorithm steps are.

1. An array of particles is initialized with a random position.
2. By Eq(3),the mean best position is determined among the particles
3. The desired objective function is evaluated for each particle and compare with the particle's previous best: If the current value is less than the previous best value, then the current value is set to the best value. That is, if B::E < B:2E ;PDAJ :E= 2E

4. The current global position minimum is determined among the particle's best positions. That is: $g = \arg\min_{1 \le i \le M}(f(P_i))$ M is the population size).
5. The current global position is Compared to the previous global, if the current global position is less than the previous global position then the current global is set to the global position.
6. For each the particledimension, get a stochastic point between$P_{gd}$and P_{id}, Eq(4)
7. Get the new position by stochastic equation Eq(5).
8. Repeat from step (2) to step (7) until satisfying a stop criterion is OR a prespecified number of iterations are completed.[4]

1.3. The classical k-means clustering algorithm

1) Initialization: K objects which is randomly selected serve as center of initial k clustering collection.

2) Distribution: Searching the nearest clustering center of each sample, and distribute the sample to the corresponding clustering.

3) Modifying clustering center: Calculate the mean value of every new divided clustering center.

4) Calculation of deviation and judgment.

5) Repeat (2)(3) until there is not changing.[2]

2. k-means with QPSO

In QPSO based clustering algorithem, each particle has own cluster centers. where C_{ij}= (C_{i1}, C_{i2} ···, C_{ik}) represents the K cluster centered vectors, C_{ij} refer to the jth cluster centered vector of the ith particle. Particle position encoding structure is as follows:

Algorithm 1: k-means using Quantum PSO.

```
Centers=zeros (number_of_clusters, size_of_features, population)
for i=1 to population
Centers(:,:,i)=Features(rand,:)
Clusters(i,:) = k-Means(Features, number_of_clusters, Centers(:,:,i))
fit(i)= fitness function (Clusters (I, :))
end
b=min(fit)
pgmin= Centers(:,:,b);
pmin= Centers;
```

```
fitmin=fit;
fgmin=min(fit);
do
find out mbest using Eq.(12)
for i=1 to population
for j=1 to number_of_clusters
for s=1 to size_of_features
fi=rand(1);
p=fi*pmin(j,s,i)+(1-fi)*pgmin(j,s);
u=rand(1);
Centers(j,s,i)=p+((-1)^ceil(0.5+rand(1)))*(alpha*abs(mbest- Centers (j,s,i))*log(1/u));
Centers (j,s,i)=min(Centers(j,s,i),vmax);
Centers (j,s,i)=max(Centers (j,s,i),-vmax);
End for
End for
End for
for i=1 to population
Clusters(i,:) = k-Means(Features, number_of_clusters,  Centers(:,:,i))
fitnew(i)= fitness function (Clusters (i, :))
end for
for i=1 to population
if fitnew(i)<fitmin(i)
fitmin(i)=fitnew(i);
pmin(:,:,i)= Centers(:,:,i);
end if
if fitnew(i)<fgmin
pgmin= Centers(:,:,i);
fgmin=fitnew(i);
end if
end for
Until termination criterion is met
cluster_centers=pgmin;
Clusters = k-Means(Features, number_of_clusters, cluster_centers)
```

The functions measurethe clustering quality can be used as fitness functions. When determining the clustering partitions by using different clustering criterion can yield variable results when applied to same data. The (SSE) Sum ofSquaresError criterion was most commonly used by the clustering criterion functions . SSE is calculated by the total sum of the squared distance between all data points and its center of cluster. The equation of calculated the SSE is given by:

$$E = \sum_{j=1}^{K} \sum_{x_i \in v_j} \left\| x_i - v_j \right\|^2 \quad (6)$$

3. Results

The effectiveness of using the Quantum Particle Swarm Optimization in clustering algorithm which was been proposed.This section tests the performance of proposed algorithm and compares with the performance of traditional k-means and k-means with PSO.

Table 1. Comparison of proposed method with traditional k-means and k-means with PSO.

dataset	number of samples	number of attribute	Method	Iterations	accuracy
Synthetic	80	2	traditional k-means	8	90
			k-means with PSO	6	92
			k-means with QPSO	4	95
Iris	150	4	traditional k-means	25	89
			k-means with PSO	16	90
			k-means with QPSO	12	94
ORL	400	6	traditional k-means	30	91
			k-means with PSO	21	93
			k-means with QPSO	17	98

4. Conclusion

This paper uses a quantum particle swarm optimization to Enhance K-Means Clustering algorithm. The primary convergence of the classical K-Means algorithm is accelerated and weakened. Also, the performance is improved to attain the global optimization. Experimental results of algorithm show not only can well used with clustering algorithm in different data set types, nevertheless also the new learning strategy improve the convergence effectively.

References

[1] J.Han,J.Pei,M.Kamber,Datamining:conceptsandtechniques,Elsevier,2011.

[2] J. A. Hartigan, M. A. Wong, Algorithm as 136: A k-means clustering algorithm, Journal of the Royal StatisticalSociety.SeriesC(AppliedStatistics)28(1)(1979)100–108.

[3] Y. Shi, et al., Particle swarm optimization: developments, applications and resources, in: evolutionary computation,2001.Proceedingsofthe2001Congresson,Vol.1,2001,pp.81–86.

[4] J. Sun, B. Feng, W. Xu, Particle swarm optimization with particles having quantum behavior, in: EvolutionaryComputation,2004.CEC2004.Congresson,Vol.1,2004,pp.325–331.

[5] M.-S. Wu, J.-H. Jeng, J.-G. Hsieh, Schema genetic algorithm for fractal image compression, Engineering Applications of Artificial Intelligence 20 (4) (2007)531–538.