PSOk-NN: A Particle Swarm Optimization Approach to Optimize k-Nearest Neighbor Classifier

Alaa Tharwat[1,2,5], Aboul Ella Hassanien[3,4,5]

[1]Dept. of Electricity- Faculty of Engineering- Suez Canal University, Ismaalia, Egypt.
[2]Faculty of Engineering, Ain Shams University, Cairo, Egypt.
[3]Faculty of Computers Information, Cairo University, Cairo, Egypt.
[4]Faculty of Computers and Information, Beni Suef University - Egypt.
[5]Scientific Research Group in Egypt (SRGE) http://www.egyptscience.net.

Swarm Work Shop - Nov. 7, 2015

- Introduction
- Theoretical Background.
- Proposed Model.
- Experimental Results.
- Conclusions and Future Work

- In machine learning field, there are two main learning approaches, namely, *supervised* and *unsupervised* learning approaches.
- There are two main techniques of supervised learning, namely, *regression* and *classification*.
- In the unsupervised approach, the targets or responses of the input data are not required to build the model.
- There are many types of classifiers, but k-Nearest Neighbour (k-NN) classifier is one of the oldest and simplest classifier.

- k-Nearest Neighbour (k-NN) is one of the most common and simple methods for pattern classification.
- In k-NN classifier, an unknown pattern is distinguished or classified based on the similarity to the known samples (i.e. labelled or training samples) by computing the distances from the unknown sample to all labelled samples and select the k-nearest samples as the basis for classification.
- The unknown sample is assigned to the class containing the most samples among the k-nearest samples (i.e. voting), thus, the k parameter must be odd.

- The main objective of the PSO algorithm is to search in the search space for the positions which are close to the global minimum or maximum solution.
- In PSO algorithm, a number of particles, agents, or elements which represent the solutions are randomly placed in the search space. The number of particles is determined by a user.
- The current location or position of each particle is used to calculate the objective or fitness function at that location.
- Each particle has three values, namely, position ($x^i \in \mathcal{R}^n$), velocity (v^i), the previous best positions (p^i), and (G) which represents the position of the best fitness value achieved.

- The velocity of each particle is adjusted in each iteration as shown in Equation (1).
- The movement of any particle is then calculated by adding the velocity and the current position of that particle as in Equation (2).

$$v^i_{(t+1)} = \text{Current Motion} + \text{Particle Memory Influnce} + \text{Swarm Influnce}$$

$$v^i_{(t+1)} = wv^i_{(t)} + C_1 r_1 (p^i_t - x^i_{(t)}) + C_2 r_2 (G - x^i_{(t)}) \tag{1}$$

$$x^i_{(t+1)} = x^i_{(t)} + v^i_{(t+1)} \tag{2}$$

where w represents the inertia weight, C_1 is the cognition learning factor, C_2 is the social learning factors, r_1, r_2 are the uniformly generated random numbers in the range of $[0 \ , \ 1]$.

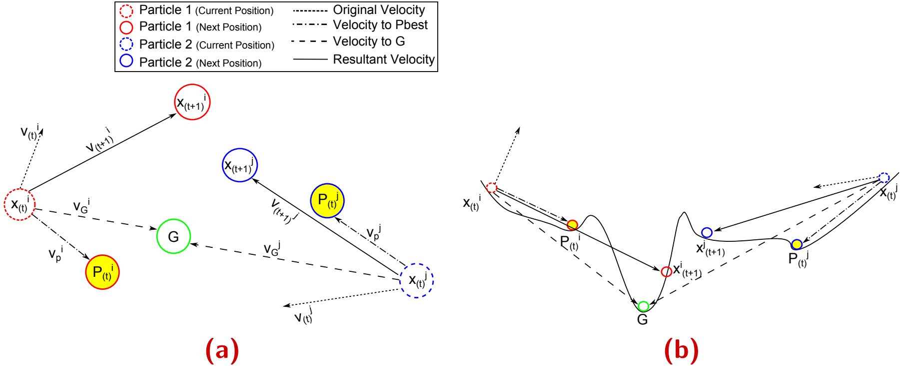


Figure: An example to show how two particles are move using PSO algorithm, (a) general movement of the two particles, (b) movement of two particle in one-dimensional space.

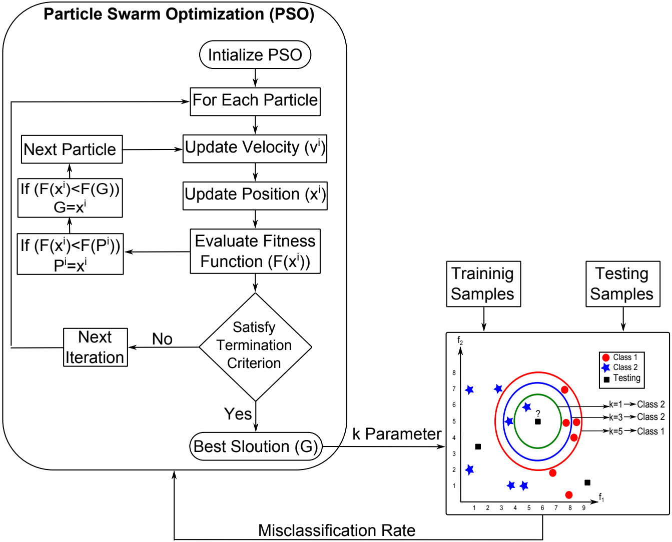


Figure: PSOk-NN algorithm searches for the optimal k parameter which minimizes the misclassification rate of the testing samples.

Table: Description of the training data used in our simulated example.

Pattern No.	Class 1 (ω_1)		Class 2 (ω_2)	
	f_1	f_2	f_1	f_2
1	7	1	3	3
2	5	2	4	4
3	9	2	7	4
4	10	4	5	5
5	8	4	6	5
6	11	4	6	10
7	9	9	4	11
8	9	11	2	11
9	10	9	2	6
10	8	6	5	9

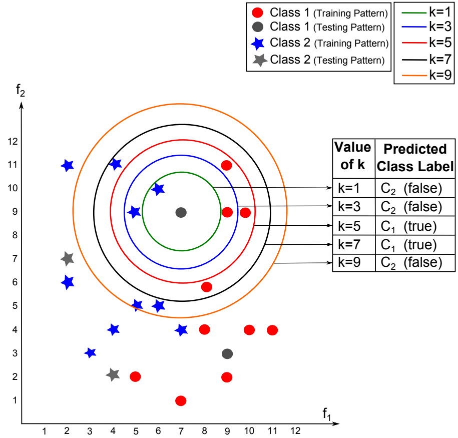


Figure: Example of how k parameter controls the predicted class labels of the unknown sample, hence controls the misclassification rate.

Table: Description of the testing data used in our simulated example and its predicted class labels using k-NN classifier using different values of k.

Testing Samples			True Class Label (y_i)	Predicted Class Labels ($\hat{y}_i$)				
No. of Sample	f_1	f_2		k=**1**	k=**3**	k=**5**	k=**7**	k=**9**
1	7	9	1	**2**	**2**	1	1	**2**
2	4	2	2	**1**	2	2	2	2
3	9	3	1	1	1	1	1	1
4	2	7	2	2	2	2	2	2
Misclassification Rate (%)				50	25	0	0	25

The bold values indicate the wrong class label.

Initial Values					
Particle No.	Position (x^i)	Velocity (v^i)	Fitness Function ($\mathcal{F}$)	P^i	G
1	1	0	100	-	-
2	9	0	100	-	-
3	5	0	100	-	-
4	3	0	100	-	-
First Iteration					
1	1	5.6	50	1	-
2	9	-5.6	25	9	-
3	5	0	**0**	5	G
4	3	2.8	25	3	-
Second Iteration					
1	5	3.36	0	5	G
2	5	-3.36	0	5	G
3	5	0	0	5	G
4	5	-1.68	0	5	G

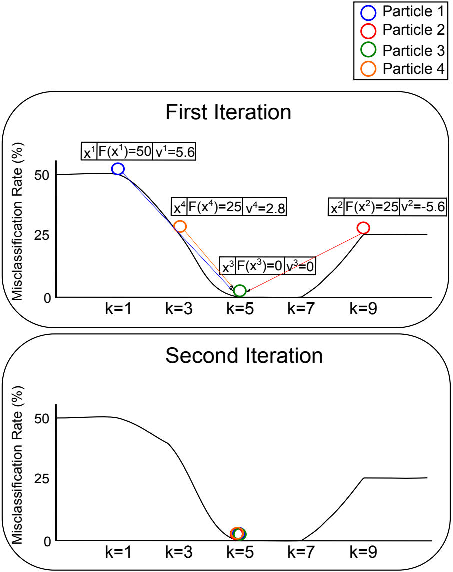


Figure: Visualization of how PSO algorithm searches for the best k value which achieves the minimum misclassification rate.

Table: Data sets description.

Data set	Dimension	Samples	Classes
Iris	4	150	3
Ionosphere	34	351	2
Liver-disorders	6	345	2
Ovarian	4000	216	2
Breast Cancer	13	683	2
Wine	13	178	3
Sonar	60	208	2
Pima Indians Diabetes	8	768	2
$ORL_{32\times32}$	1024	400	40
$Yale_{32\times32}$	1024	165	15

Dataset	PSOk-NN	GAk-NN	ACOk-NN
	Misclassification Rate	Misclassification Rate	Misclassification Rate
Iris	**1.4667$\pm$0.4216**	4$\pm$0	2.6667$\pm$0
Iono	**13.1429$\pm$0**	17.1429$\pm$0	16.9143$\pm$0.5521
Liver	**30.9302$\pm$1.4708**	31.9767.$\pm$0	35.4651$\pm$7.4898$\times 10^{-15}$
Ovarian	**13.0556$\pm$0.2928**	14.2321$\pm$0.2145	13.8889$\pm$0
Breast Cancer	**30.3021$\pm$(0.8037)**	31.0850$\pm$7.4898$\times 10^{-15}$	32.2581$\pm$7.4898$\times 10^{-15}$
Wine	**23.0899$\pm$0**	24.7191$\pm$3.7449$\times 10^{-15}$	28.3146$\pm$0.7106
Sonar	17.45$\pm$0	21.1538$\pm$0	**17.3077$\pm$2.0271**
Diabate	24.7448$\pm$0.9025	**22.9167$\pm$3.7449**$\times 10^{-15}$	26.0417$\pm$7.4898$\times 10^{-15}$
ORL$_{32\times 32}$	**8.5$\pm$0**	9.5$\pm$0	**8.5$\pm$0**
Yale$_{32\times 32}$	**21.9512$\pm$3.7449**$\times 10^{-15}$	**21.9512$\pm$3.7449**$\times 10^{-15}$	25.8537$\pm$0.7713

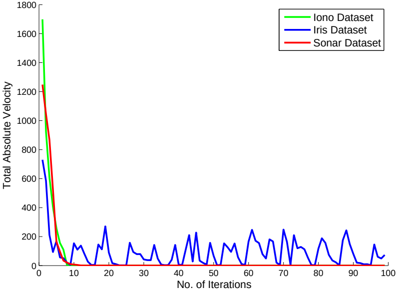
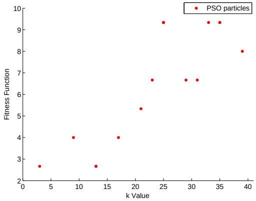
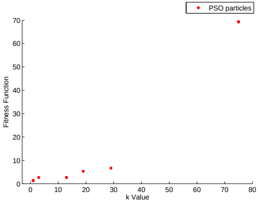


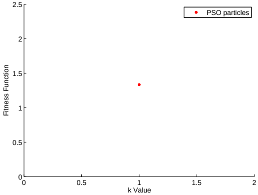
Figure: Toal absolute velocity of the PSOk-NN algorithm using Iono, Iris, and Sonar datasets.



(a) After the first iteration

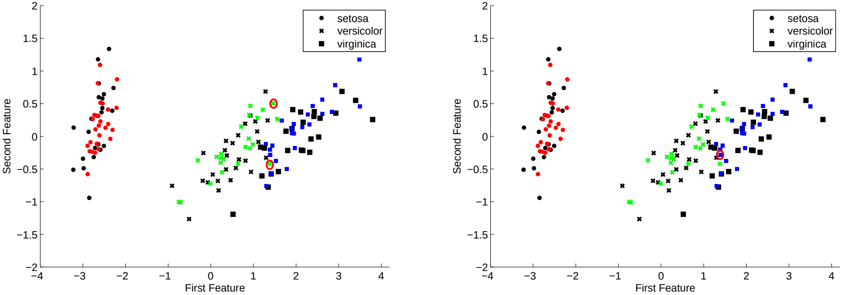


(b) After the second iteration



(c) After the tenth iteration

Figure: Visualization of the movements of all particles of PSOk-NN algorithm till it reaches to the optimal solution which achieved the minimum misclassification rate.



(a) After the first iteration

(b) After the tenth iteration

Figure: Misclassification samples after the first and tenth iterations using PSOk-NN algorithm.

- PSOk-NN algorithm achieved the minimum misclassification error in eight of the datasets (80%) compared with the other two algorithms.
- PSOk-NN algorithm converges to the optimal solution faster than the other two algorithms due to the use of linearly decreasing inertia weight in PSO algorithm.
- GAk-NN fluctuating up and down, while PSOk-NN algorithm is more stable during converging to the optimal solution because in PSO, the best solution gives information to all other particles to move to the optimal solution, while in GA the all agents are changed randomly without any guiding from any agent.

Thank You

Qurstions

SCIENTIFIC RESEARCH GROUP IN EGYPT

SRGE

SCIENTIFIC RESEARCH GROUP IN EGYPT