CHAPTER IV

EXPERIMENTS AND RESULTS

In this work, the experiments were conducted by MATLAB executing on Intel Core i7-2600 CPU 3.40 GHz and 16-GB RAM. Five-fold cross validation was used to evaluate the proposed model DLSC. Many real world data sets with various sizes were used to test the performance of the model. Each data set was partitioned into five disjoint subsets. For each fold, four subsets were used as a training subset and the remaining subset was used as a test subset. In this study, two case studies of presented data, namely complete training data set available called *complete training data scenario* and streaming training data chunk available called *streaming training data scenario*, were created and investigated. For the first case, the entire training subset was presented to the learning process. This situation was used to test the performance of the proposed method on complete and large data set test available. For the second case, the entire training subset was randomly selected to form the streaming data chunks. Performance of the DLSC was evaluated in terms of percentage of accuracy classification (%), the number of processing or hidden nodes, and computational time (*s*) of the learning process.

4.1 Experiments for Complete Training Data Scenario

The proposed method was tested on 12 real-world data sets ranging from small to large size. The size of a data set was defined as the product of the number of attributes and the number of samples. The 11 well-known data sets were obtained from the University of California at Irvine [30], namely Iris, Yeast, Image Segmentation, Cardiotocography, Waveform, Letter Recognition, Multi-feature Digit, Liver, Spambase, Internet Advertisement, and MiniBooNE and one data set was physical protein-protein interactions of yeast Saccharomyces Cerevisiae freely available at *http://www.scucic.cn/Predict_PPI/index.htm*. These data were composed of both equal amounts of positive and negative data. The negative data were generated by the assumption that proteins with different sub-cellular localizations did not interact. The characteristics of the experimental data sets from life, physical and Computer areas contained both real and categorical attribute types. For each data set, the instances with unknown or missing values were removed. Among the 12

data sets, there were seven data sets containing multi-class data, namely, Iris, Yeast, Image Segmentation, Cardiotocography, Waveform, Letter Recognition, and Multi-feature Digit, and five data sets containing the two-class data, namely, Liver, Spambase, Internet Advertisement, Protein Interactions and MiniBooNE. The descriptions of the data sets are given in Table 1.

Table 1: Description of each data set for complete training data scenario.

Data Set	Number of Attributes	Number of Samples	Size*	Number of Classes	Area
Iris	4	150	600	3	Life
Yeast	8	1,484	11,872	10	Life
Image Segmentation	19	2,310	43,890	7	N/A
Cardiotocography	23	2,126	48,898	10	Life
Waveform	21	5,000	105,000	3	Physical
Letter Recognition	16	20,000	320,000	26	Computer
Multi-feature Digit	649	2,000	1,298,000	10	Computer
Liver	7	345	2,415	2	Life
Spambase	57	4,601	262,257	2	Computer
Internet Advertisement	1,558	2,359	3,675,322	2	Computer
Protein Interactions	398	11,188	4,452,824	2	Physical
MiniBooNE	50	130,064	6,503,200	2	Physical

Note: Size*= (Number of Attributes)* (Number of Samples)

The experimental results of the DLSC method were compared to those of six methods including three methods of standard batch learning algorithm and three methods of incremental learning algorithm. Three batch learning algorithms consisted of conventional Radial Basis Function (RBF) with Gaussian RBF, MultiLayer Perceptron (MLP) and Support Vector Machine (SVM). Three incremental learning algorithms consisted of the learning algorithm of versatile elliptic basis function neural network for one datum parameter update called VEBF method [4], Incremental Learning Vector Quantization (ILVQ) [29], and adjusted ASC classifier (ASC) [24].

4.1.1 Experimental setting for complete training data scenario

Five-fold cross validation was adopted to tune the initial parameters of each method. The parameter η for updating the width vector was set to 2. The initial width vector $\mathbf{w}^0 = [w_1^0 \; w_2^0 \; ... \; w_n^0]^T$ of Gaussian RBF, VEBF and DLSC techniques is given by

$$w_l^0 = \frac{\delta}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} \left\| \mathbf{x}_i - \mathbf{x}_j \right\|, \qquad l = 1, ..., n \tag{31}$$

where $\|\cdot\|$ is the Euclidean distance function, N is the number of data and δ is constant. For SVM, Gaussian radial basis function was used as kernel function with setting the kernel spread to 1 and the Least Square (LS) method was used to find the separating hyperplane. For ILVQ and ASC techniques, there are two pre-defined parameters namely λ and $AgeOld$. The obtained parameters setting of DLSC, VEBF, RBF, ILVQ, and ASC techniques for complete training data scenario are shown in Table 2. For two batch MLP and RBF learning algorithms, 11 data sets, except for MiniBooNE, with the number of data less than 100,000 were empirically evaluated. For SVM suitable of dichotomy classification, because of their large number of data, only four 2-class data sets, except for MiniBooNE, were tested.

Table 2: Parameter setting in each data set for complete training data scenario

Data Set	DLSC (δ)	VEBF (δ)	RBF (δ)	ILVQ ($\lambda, AgeOld$)	ASC ($\lambda, AgeOld$)
Iris	0.7	0.3	10	(21,17)	(6,6)
Yeast	0.4	1	1	(90, 50)	(70,35)
Image Segmentation	0.7	1	1	(180, 130)	(90,40)
Cardiotocography	0.5	1	15	(200, 250)	(200,150)
Waveform	0.7	0.7	1	(70, 110)	(30, 150)
Letter Recognition	0.7	1	15	(80, 100)	(130,70)
Multi-feature Digit	0.5	1	1	(100,150)	(100,100)
Liver	0.15	1	1	(16,80,)	(6,6)
Spambase	0.4	0.7	10	(90,18)	(140,150)
Internet Advertisement	0.7	1.2	15	(200,60)	(100,160)
Protein Interactions	0.7	0.5	15	(155,60)	(110,50)
MiniBooNE	0.7	0.5	N/A	(200,150)	(200,250)

For all four incremental learning methods, including VEBF, ILVQ, ASC and proposed DLSC, all 12 data sets were used. In each fold, 10 experiments of weight initialization for MLP, learning tries for DLSC and VEBF, and random initial prototypes selection for ILVQ and ASC, were repeatedly performed. The best classification accuracy (%) was chosen to make the comparison. Because the number of hidden neurons for MLP and RBF could not be predefined, the number of used hidden neurons for these two methods was varied from the number of class labels to at most five times the number of hidden neurons used by DLSC algorithm. The average with standard deviation values of classification accuracy (%), corresponding number of hidden

neurons or prototypes and computational or learning time (*s*) of each technique on each fold were provided as shown in Tables 3 – 5. The best average value and the second value for each data set are remarked on the bold typeface and underline, respectively. The value with asterisk (*) means that there is no statistical significance ($\rho = .05$) between this value and the extreme value on each data set. For assigning class label, the parameter N^0 was introduced for choosing the candidate neuron for assigning class label. The parameter N^0 was set to a small integer so that the neuron containing the number of data less than N^0 would not be chosen to assign a class label. In these experiments, the integer N^0 was set to 3 in all data sets except for Yeast data set. In this case, N^0 was set to 1.

4.1.2 Experimental results for complete training data scenario

It can be seen from Table 3 that DLSC provides the highest classification accuracy (%) on nine data sets, namely, Iris, Yeast, Image Segmentation, Waveform, Letter Recognition, Multi-feature Digit, Liver, Protein Interaction and MiniBooNE. DLSC gives the second highest accuracy (%) on Cardiotocography, Internet Advertisement and Protein Interaction. For comparing with batch learning algorithms, the accuracy of DLSC method is statistically and significantly higher than those of both MLP and RBF methods on Yeast, Image Segmentation, Waveform, Letter Recognition, Multi-feature Digit, and Protein Interaction and higher than those of SVM on three out of four two-class data sets. There is no statistical difference of accuracy (%) among DLSC, MLP and RBF methods on Iris and among DLSC, MLP, RBF and SVM on Liver. Only Cardiotocography, Spambase, and Internet Advertisement, the accuracy of DLSC method is slightly less than that of MLP method, providing the highest values on these three data sets. Lower accuracy was obtained because these data sets contain a large number of zeros resulting in the small value of a standard deviation in each dimension of the data sets. For comparing with other three incremental learning algorithms, the average accuracy of DLSC is higher than those of VEBF, ILVQ and ASC on all of both multi-class and two-class data sets. The accuracy of DLSC is statistically and significantly different from those of VEBF, ILVQ and ASC on Cardiotocography, Multi-feature Digit, Liver, Spambase, Internet Advertisement and Protein Interaction data sets. Moreover, the DLSC method provides the minimum value of the rank average of classification accuracy (Rank Average = 1.25).

Table 3: Comparison results of average accuracy with standard deviation $(\bar{x} \pm sd)$ for complete training data scenario.

Data Sets	Batch Learning Algorithms			Incremental Learning Algorithms			
	MLP	RBF	SVM	DLSC	VEBF	ILVQ	ASC
Iris(3)	**98.67 ± 1.63**	98 ± 1.63*	N/A	**98.67 ± 1.63**	98 ± 1.63*	98 ± 1.63*	98 ± 1.67*
Yeast(10)	47.71 ± 0.61	41.98 ± 2.32	N/A	**57.75 ± 1.35**	56.33 ± 1.29*	55.26 ± 2.07*	52.83 ± 2.47
Image Segmentation(7)	84.07 ± 1.2	49.7 ± 3.12	N/A	**91.86 ± 0.74**	78.96 ± 2.02	90.52 ± 2.60*	91.3 ± 2.35*
Cardiotocography(10)	**80.15 ± 2.67**	23.03 ± 2.98	N/A	74.16 ± 2	54 ± 2.55	62.51 ± 1.39	60.49 ± 1.47
Waveform(3)	78.36 ± 1.67	69.8 ± 3.11	N/A	**85.24 ± 0.74**	84.56 ± 0.79*	83.34 ± 1.16	83.92 ± 1.02*
Letter Recognition (26)	83.2 ± 2.34	73.12 ± 0.71	N/A	**87.64 ± 0.42**	80.74 ± 0.38	81.8 ± 0.59	87.6 ± 1.04*
Multi-feature Digit (10)	75.5 ± 3.28	30.24 ± 5.5	N/A	**97.65 ± 0.8**	77.5 ± 5.09	85.5 ± 1.21	93.4 ± 0.84
Liver(2)	76.23 ± 4.36*	71.88 ± 4.36*	68.7 ± 6.05*	**76.81 ± 5.8**	67.25 ± 4.26	67.83 ± 4.24	63.19 ± 3.95
Spambase(2)	**94.63 ± 0.54**	85.22 ± 2.18	88.55 ± 1.2	91.07 ± 1.01	81.44 ± 2.31	74.24 ± 0.87	71.75 ± 1.27
Internet Advertisement(2)	**97.37 ± 0.59**	94.11 ± 1.09	95.53 ± 0.71	96.02 ± 0.31	55.21 ± 5.78	92.54 ± 1.82	91.86 ± 0.55
Protein Interaction (2)	82.46 ± 0.77	70.3 ± 0.44	76.39 ± 0.43	**85.76 ± 1.41**	61.92 ± 0.89	61.32 ± 0.65	57.48 ± 1.84
MiniBooNE (2)	N/A	N/A	N/A	**88.47 ± 0.18**	60.6 ± 6.49	86.19 ± 0.39	84.4 ± 0.34
Rank Average	2.64	4.82	3.25	**1.25**	4.33	4.00	4.08

Referring to Table 4, it is shown that the number of hidden neurons of SVM is not included to make a comparison because there is only one separating hyperplane for all used two-class data sets. The number of hidden neurons of DLSC is the lowest on ten data sets, namely, Yeast, Image Segmentation, Cardiotocography, Waveform, Multi-feature Digit, Liver, Spambase, Internet Advertisement and MiniBooNE. The DLSC provides the second minimum numbers on Letter Recognition and Protein Interaction. For comparison with batch learning algorithms, the average number of hidden neurons of DLSC is statistically and significantly less than those of both MLP and RBF methods on eight data sets. There is no statistical and significant difference of the number of hidden neurons between DLSC and MLP on Iris and Yeast data sets and also between DLSC and RBF on Liver data sets. For comparison with three incremental learning algorithms, the number of hidden neurons of DLSC method is statistically and significantly less than that of both prototype-based classifier ILVQ and ASC methods on all 12 data sets. The number of hidden neurons of DSCL is significantly less than that of VEBF method on Image Segmentation, Liver, Spambase, Internet Advertisement, and MiniBooNE. There is no statistical difference between DLSC and VEBF on Iris, Yeast, Cardiotocography, Waveform, and Multi-feature Digit. However, the number of hidden neurons of VEBF outperforms that of DLSC on Letter Recognition and Protein Interaction. Moreover, the DLSC method provides the minimum value of the rank average of number of hidden neurons (Rank Average = 1.33).

Table 5 shows the ratio between the number of hidden neurons or prototypes of other methods and the number of hidden neurons of DLSC method in each data set. Since the number of hidden neurons for MLP and RBF is fixed and varied within five times the number used by DLSC, the number of hidden neurons of MLP, RBF and DLSC is not much different as compared to the other three incremental learning algorithms. For MiniBooNE, the number of hidden or prototypes of VEBF, ILVQ and ASC is just much larger than that of DLSC. For comparing with the MLP and RBF, the number of hidden neurons of both methods varied from the number of class labels to at most five times number of neurons used by DLSC algorithms. So, the number of hidden neurons of both methods is in this range.

However, for comparing with the other three incremental learning algorithms, the number of hidden neurons of the DLSC is considerably less than that of other incremental algorithms (12.4 times for VEBF, 33.8 times for ILVQ and 24.3 times for ASC).

From Table 6, it can be seen that the average learning time of DLSC is lower than that of the others on eight data sets, namely, Iris, Yeast, Image Segmentation, Cardiotocography, Waveform, Letter Recognition, Spambase, and MiniBooNE. DLSC yields the second in learning time (*s*) on Protein Interaction. For comparison with batch learning algorithms, the average learning time of DLSC is statistically and significantly less than that of MLP method for all data sets and that of RBF methods on 10 data sets except for Internet Advertisement. For SVM only four data sets of 2-class classification, based on least square method for solving hyperplane, the learning time of SVM is less than that DLSC on Liver, Internet Advertisement and Protein Interaction. For comparison with incremental learning algorithms, the learning time of DLSC method is significantly less than that of VEBF method on almost all data sets except only for Liver data set having small size. The learning time of DLSC is significantly less than that of ASC method on eight data sets. There is no statistical difference between DLSC and ASC for Liver. The learning time of DLSC is statistically slower than that of ASC only for Multi-feature Digit. The time of DLSC is slower than that of ILVQ only Multi-feature Digit and Internet Advertise having large number of features. Because Multi-feature Digit and Internet Advertise data sets contain a very large number of attributes (greater than one forth and a half of the number of samples for Multi-feature Digit and Internet Advertisement, respectively, as shown in Table 1), the learning algorithm of DLSC took a long time for eigenvector and eigenvalue computations of the corresponding covariance matrices of sizes 649×649 for Multi-feature Digit, and $1{,}558 \times 1{,}558$ for Internet Advertisement in the parameter updating step. Moreover, Table 7 shows the number of times of computational time of each method with respect to the learning time of DLSC method on each data set. The average number of computational time of MLP, RBF, SVM, VEBF, ILVQ, and ASC methods is approximately 3,458, 52, 2, 33, 44 and 55 times more than those of DLSC method, respectively.

Table 4: Comparison results of the average number of hidden neurons or prototypes with standard deviation $(\bar{x} \pm sd)$ for complete training data scenario.

Data Sets	Batch Learning Algorithms			Incremental Learning Algorithm			
	MLP	RBF	SVM	DLSC	VEBF	ILVQ	ASC
Iris(3)	5.20 ± 2.04*	**3.00 ± 0.00**	N/A	3.40 ± 0.49*	**3.00 ± 0.00**	25.60 ± 8.40	15.4 ± 1.85
Yeast(10)	18.60 ± 8.38*	25.00 ± 5.76	N/A	**17.00 ± 1.90**	17.20 ± 0.40*	158.20 ± 46.74	304.80 ± 40.39
Image Segmentation(7)	11.20 ± 2.99	12.20 ± 2.23	N/A	**7.40 ± 0.49**	11.80 ± 0.98	459.80 ± 143.50	451.60 ± 19.99
Cardiotocography(10)	25.60 ± 6.84	24.80 ± 4.60	N/A	**10.20 ± 0.45**	11.00 ± 0.71*	466.80 ± 82.86	547.00 ± 16.90
Waveform(3)	5.00 ± 1.26	6.00 ± 0.00	N/A	**3.00 ± 0.00**	**3.00 ± 0.00**	110.60 ± 20.98	138.00 ± 36.77
Letter Recognition (26)	30.40 ± 7.84	60.80 ± 0.98	N/A	27.20 ± 0.75	**26.00 ± 0.00**	722.20 ± 18.78	106.88 ± 2.9
Multi-feature Digit (10)	44.80 ± 2.40	60 ± 3.81	N/A	**12.60 ± 1.34**	**12.60 ± 0.55**	189.80 ± 46.70	423.60 ± 32.68
Liver(2)	9.60 ± 0.49	8.20 ± 0.75*	N/A	**7.40 ± 0.80**	9.40 ± 1.02	48.20 ± 17.66	31.40 ± 15.32
Spambase(2)	18.20 ± 2.14	19.80 ± 0.75	N/A	**11.60 ± 1.62**	18.80 ± 0.75	167.60 ± 64.10	113.40 ± 25.10
Internet Advertisement(2)	6.20 ± 1.33	9.40 ± 0.49	N/A	**3.60 ± 0.49**	12.60 ± 0.80	252.60 ± 96.42	52.20 ± 8.61
Protein Interaction (2)	23.20 ± 2.79	28.00 ± 1.26	N/A	13.80 ± 5.15	**2.00 ± 0.00**	237.80 ± 86.83	122.8 ± 26.75
MiniBooNE (2)	N/A	N/A	N/A	**6.60 ± 0.55**	892.8 ± 69.61	621.80 ± 62.54	222.20 ± 35.23
Rank Average	3	3.36		**1.33**	2.17	5.25	5.25

Table 5: The ratio of hidden neurons or prototypes of each method with respect to that of DLSC.

Data Sets	MLP	RBF	DLSC	VEBF	ILVQ	ASC
Iris	1.5	0.9	1.0	0.9	7.5	4.5
Yeast	1.1	1.5	1.0	1.0	9.3	17.9
Image Segmentation	1.5	1.6	1.0	1.6	62.1	61.0
Cardiotocography	2.5	2.4	1.0	1.1	45.8	53.6
Waveform	1.7	2.0	1.0	1.0	36.9	46.0
Letter Recognition	1.1	2.2	1.0	1.0	26.6	3.9
Multi-feature Digit	3.6	4.8	1.0	1.0	15.1	33.6
Liver	1.3	1.1	1.0	1.3	6.5	4.2
Spambase	1.6	1.7	1.0	1.6	14.4	9.8
Internet Advertise	1.7	2.6	1.0	3.5	70.2	14.5
Protein Interaction	1.7	2.0	1.0	0.1	17.2	8.9
MiniBooNE	N/A	N/A	1.0	135.3	94.2	33.7
Average	**1.8**	**2.1**	**1.0**	**12.4**	**33.8**	**24.3**

4.1.3 Comparison Results on Effect of Order of Presented Classes

The order of learned classes does not affect the performance of DLSC because parameter update for each class does not concern with the distribution of each other classes. Thus, the experiment on the effect of order of presented input pattern was also evaluated. Only four incremental learning algorithms (DLSC, VEBF, ILVQ and ASC) were performed. The hold-out validation was used to evaluate the models. Each of 11 data sets was divided into two subsets, namely, training and test subsets. The percentages of the training and test sets are 80 % and 20 %, respectively. For training process, 10 distinctive orders of presented patterns of training set were randomly generated. Table 8 shows the comparison results of average accuracy (%) and the average number of hidden neurons or prototypes with standard deviation on 10 distinctive orders of 11 data sets. For the perspective of

Table 6: Comparison results of the average learning time (*s*) with standard deviation ($\overline{x} \pm sd$) for complete training data scenario.

Data Sets	Batch Learning Algorithms			Incremental Learning Algorithms			
	MLP	RBF	SVM	DLSC	VEBF	ILVQ	ASC
Iris (3)	0.96 ± 0.80	0.12 ± 0.00	N/A	**0.01 ± 0.00**	0.02 ± 0.01	0.07 ± 0.01	0.08 ± 0.01
Yeast (10)	12.19 ± 10.09	4.16 ± 0.98	N/A	**0.14 ± 0.02**	0.32 ± 0.03	2.52 ± 0.12	1.18 ± 0.06
Image Segmentation (7)	2.76 ± 1.26	3.62 ± 0.60	N/A	**0.09 ± 0.00**	0.78 ± 0.08	7.78 ± 1.31	2.28 ± 0.06
Cardiotocography (10)	3,009 ± 1,890	2.97 ± 0.54	N/A	**0.11 ± 0.01**	0.61 ± 0.05	3.62 ± 0.14	1.30 ± 0.04
Waveform (3)	1.52 ± 0.67	10.65 ± 0.03	N/A	**0.16 ± 0.00**	1.37 ± 0.01	10.53 ± 1.09	6.64 ± 0.76
Letter Recognition (26)	5,891.5 ± 5,719.6	266.03 ± 3.62	N/A	**0.79 ± 0.03**	4.78 ± 0.03	106.88 ± 2.90	21.29 ± 1.44
Multi-feature Digit (10)	44,434 ± 271	226.15 ± 6.07	N/A	19.39 ± 2.29	430.34 ± 11.15	**5.33 ± 0.07**	8.13 ± 1.89
Liver (2)	0.32 ± 0.01	0.68 ± 0.07	**0.03 ± 0.02**	0.17 ± 0.02	0.10 ± 0.03	0.20 ± 0.01	0.17 ± 0.01
Spambase (2)	23.44 ± 7.83	28.53 ± 0.64	5.23 ± 0.14	**0.75 ± 0.04**	21.32 ± 0.58	11.85 ± 0.89	6.95 ± 0.72
Internet Advertisement (2)	2,042 ± 1,163	39.82 ± 0.18	**1.75 ± 0.02**	104.71 ± 6.55	11,394,513	14.82 ± 1.42	124.58 ± 44.84
Protein Interaction (2)	83,370 ± 79,982	154.09 ± 1.76	**9.28 ± 0.25**	30.90 ± 4.01	538.22 ± 3.93	49.53 ± 1.78	127.27 ± 10.17
MiniBooNE (2)	N/A	N/A	N/A	**14.51 ± 0.60**	2,642,193	22,912 ± 37	7,931 ± 664
Rank Average	5.73	5.09	**1.25**	**1.25**	1.67	3.25	3.42

Table 7: Ratio of learning time of each method with respect to that of DLSC.

Data Sets	MLP	RBF	SVM	DLSC	VEBF	ILVQ	ASC
Iris	135.2	16.7	N/A	1.0	3.4	9.3	10.9
Yeast	86.7	29.6	N/A	1.0	2.2	17.9	8.4
Image Segmentation	31.6	41.4	N/A	1.0	9.0	89.0	26.0
Cardiotocography	2,7347.0	27	N/A	1.0	5.5	32.9	11.8
Waveform	9.3	65.2	N/A	1.0	8.4	64.4	40.6
Letter Recognition	7,453.5	336.6	N/A	1.0	6.0	135.2	26.9
Multi-feature Digit	229.2	11.7	N/A	1.0	22.2	0.3	0.4
Liver	1.9	3.9	0.2	1.0	0.6	1.2	1.0
Spambase	31.4	38.2	7.0	1.0	28.5	15.9	9.3
Internet Advertise	19.5	0.4	0.02	1.0	108.8	0.1	1.2
Protein Interaction	2,698.4	5.0	0.3	1.0	17.4	1.6	4.1
MiniBooNE	N/A	N/A	N/A	1.0	182.1	157.9	546.6
Average	**3,458**	**52**	**2**	**1.0**	**33**	**44**	**57**

average accuracy results, DLSC and ILVQ provide the highest average accuracy on Iris data set. VEBF achieves the highest average accuracy only on Wavefrom data set. The accuracy results of DLSC outperform those of VEBF, ILVQ and ASC, obviously on the remaining 10 data sets. For the perspective of the average number of hidden neurons or prototypes as shown in Table 9, the lowest average number of hidden neurons of DLSC is obtained on Iris, Image Segmentation, Letter Recognition, Spambase, and Internet Advertisement. The same lowest average number of hidden neurons of DLSC and VEBF appears on Waveform while VEBF yielded the lowest average number of hidden neurons on Yeast, Liver, and Protein Interaction. Furthermore, the effect of class order is also expressed in terms of average value of the standard deviation values of accuracy and the numbers of hidden neurons or prototypes on 11 data sets as shown in Figure 7, For the effect of order on the accuracy, the average standard deviation of DLSC (0.43) is obviously less than that of VEBF, ILVQ and ASC (7.15, 1.26, and 2.30, respectively). For the effect of order on the number of hidden neurons or prototypes, the average of standard deviation value of

Table 8: Comparison results of average accuracy (%) and standard deviation $(\bar{x} \pm sd)$ on holdout validation of ten distinctive orders presented patterns

Data Sets	DLSC	VEF	ILVQ	ASC
Iris	**100.00 ± 0.00**	97.00 ± 5.26	**100.00 ± 0.00**	99.33 ± 1.33
Yeast	**56.05 ± 0.86**	28.85 ± 2.64	53.72 ± 1.58	45.45 ± 4.03
Image Segmentation	**93.94 ± 0.22**	70.32 ± 5.85	87.73 ± 1.89	89.52 ± 2.37
Cardiotocography	**70.49 ± 0.66**	42.82 ± 5.87	61.39 ± 1.66	56.99 ± 1.86
Waveform	83.70 ± 0.00	**84.28 ± 0.35**	83.76 ± 0.50*	82.58 ± 1.00
Letter Recognition	**88.17 ± 0.05**	67.17 ± 5.23	70.04 ± 1.56	86.76 ± 0.73
Multi-feature Digit	**96.88 ± 0.13**	66.43 ± 4.16	77.80 ± 2.47	91.65 ± 0.48
Liver	**75.36 ± 2.25**	64.93 ± 7.85	71.30 ± 3.09	60.72 ± 6.43
Spamase	**90.62 ± 0.10**	80.00 ± 3.57	72.97 ± 1.11	70.20 ± 2.11
Internet Advertisement	**96.39 ± 0.00**	39.07 ± 28.43	91.93 ± 0.49	90.30 ± 1.85
Protein Interaction	**85.31 ± 0.16**	55.87 ± 6.46	62.53 ± 0.76	56.58 ± 1.25

Table 9: Comparison results of number of hidden neurons or prototypes and standard deviation ($\bar{x} \pm sd$) on holdout validation of ten distinctive orders presented patterns.

Data Sets	DLSC	VEBF	ILVQ	ASC
Iris	**3.00 ± 0.00**	3.90 ± 0.70	31.30 ± 9.61	12.60 ± 2.69
Yeast	18.40 ± 1.28	**14.40 ± 0.92**	185.40 ± 67.52	209.10 ± 50.77
Image Segmentation	**7.00 ± 0.00**	14.00 ± 0.63	328.70 ± 97.01	321.90 ± 69.51
Cardiotocography	12.80 ± 1.54*	**11.10 ± 0.54**	598.00 ± 199.02	489.60 ± 34.91
Waveform	**3.00 ± 0.00**	**3.00 ± 0.00**	35.10 ± 9.66	56.40 ± 23.43
Letter Recognition	**26.70 ± 0.64**	30.80 ± 2.27	490.50 ± 85.65	3,420.70 ± 273.87
Multi-feature Digit	11.90 ± 0.94*	**11.40 ± 0.49**	108.90 ± 29.70	397.60 ± 43.57
Liver	7.10 ± 1.87	**5.40 ± 0.66**	24.10 ± 9.97	19.30 ± 9.74
Spambase	**11.20 ± 1.60**	16.00 ± 1.26	147.50 ± 50.73	147.60 ± 46.26
Internet Advertisement	**4.70 ± 0.64**	11.20 ± 1.99	128.90 ± 33.15	47.70 ± 10.40
Protein Interaction	4.70 ± 1.19	**2.90 ± 1.22**	240.40 ± 38.41	86.80 ± 23.25

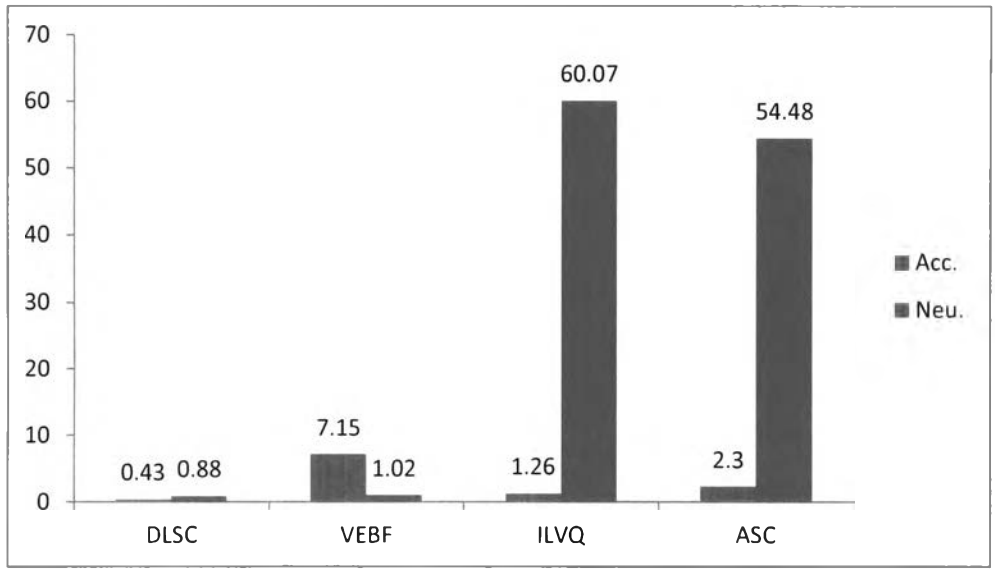

Figure 7: The average of standard deviation values of classification accuracy (Acc.) and the number of hidden neurons (Neu.) for ten distinctive presented orders

DLSC (0.88) is also clearly less than that of VEBF, ILVQ and ASC (1.02, 60.07, and 5 4.48, respectively).

4.1.4 Influence of center selection and initial width parameter on classification accuracy and the number of hidden neurons of DLSC

To test the influence of center initialization, 10 distinct orders of each 11 data sets were generated. The pre-defined δ parameter was set according to Table 2. For each order, the first data point was selected as the center of the first hidden neuron. Then after discarding the covered data from the learning process, the first data point was also chosen as the center of new hidden neuron for the remaining training set as well. The influence of center vector selection on accuracy (%) and number of hidden neurons on each data set are shown in Figures 8 and 9, respectively. From Figure 8 it is shown that the accuracy lines of most data sets was approximately straight. This implies that the accuracy is not affected by center vector selection in most data set. Only Liver and Yeast data sets when the accuracies were slightly affected by the center selection. From Figure 9, it is shown that the number of hidden neurons is also slightly affected by the center selection. The range of

numbers of hidden neurons for each data set and for each center selection does not exceed five.

To test the influence of the δ parameter, various values of δ parameter, varied from 0.05 to 1.4 incremented by 0.05, were investigated with selecting the same center vectors for all values on each data set. The results on accuracy (%) are shown in Figure 10 where the accuracy is so slightly affected by this range for four data sets including Image Segmentation, Multi-feature Digit, Spambase, and Internet Advertisement. For the rest of data sets, the accuracy slightly fluctuates from 0.3 to 1.4. However, the accuracy dramatically dropped when the δ value was less than 0.3 except for Liver getting higher accuracy at $\delta = 0.15$. For the results on the number of hidden neurons as shown in Figure 11, the numbers of neurons is slightly affected when the δ ranges in between 0.5 to 1.14 for almost all data sets, except for Yeast data set for which the numbers of neurons fluctuate and decline. For δ less than 0.5, the numbers of neurons dramatically increase for most data sets but, the numbers of neurons gradually decrease for four data sets namely, Iris, Yeast, Waveform and Liver. The sizes of four data sets are quite small.

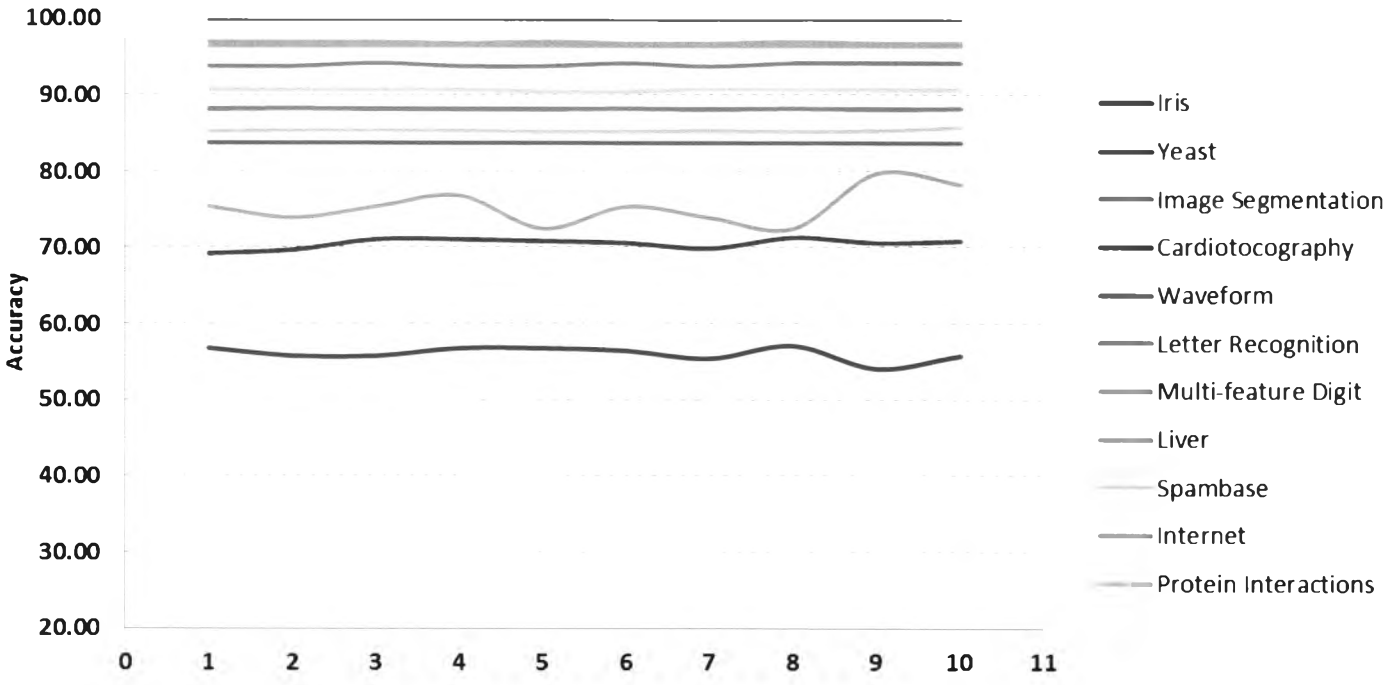

Figure 8: Influence of different initial center vectors on classification accuracy (%)

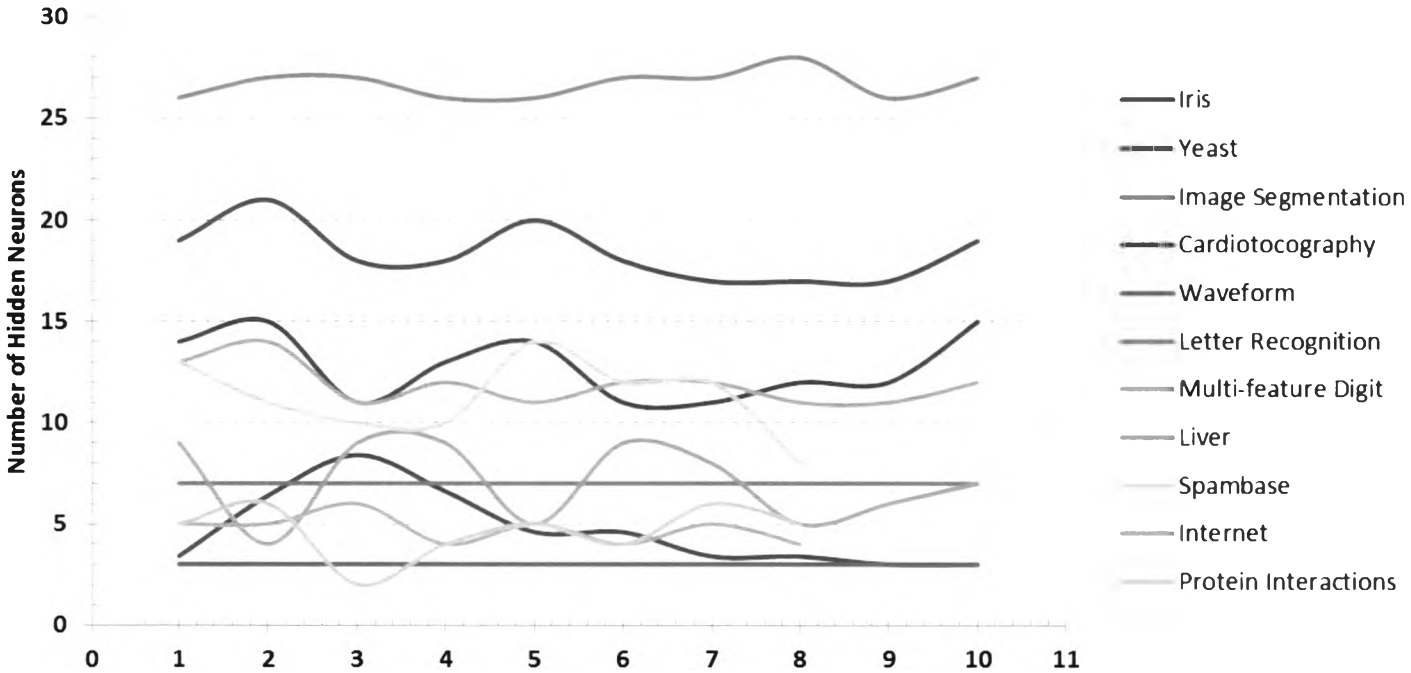

Figure 9: Influence of ten different initial center vectors of DLSC method

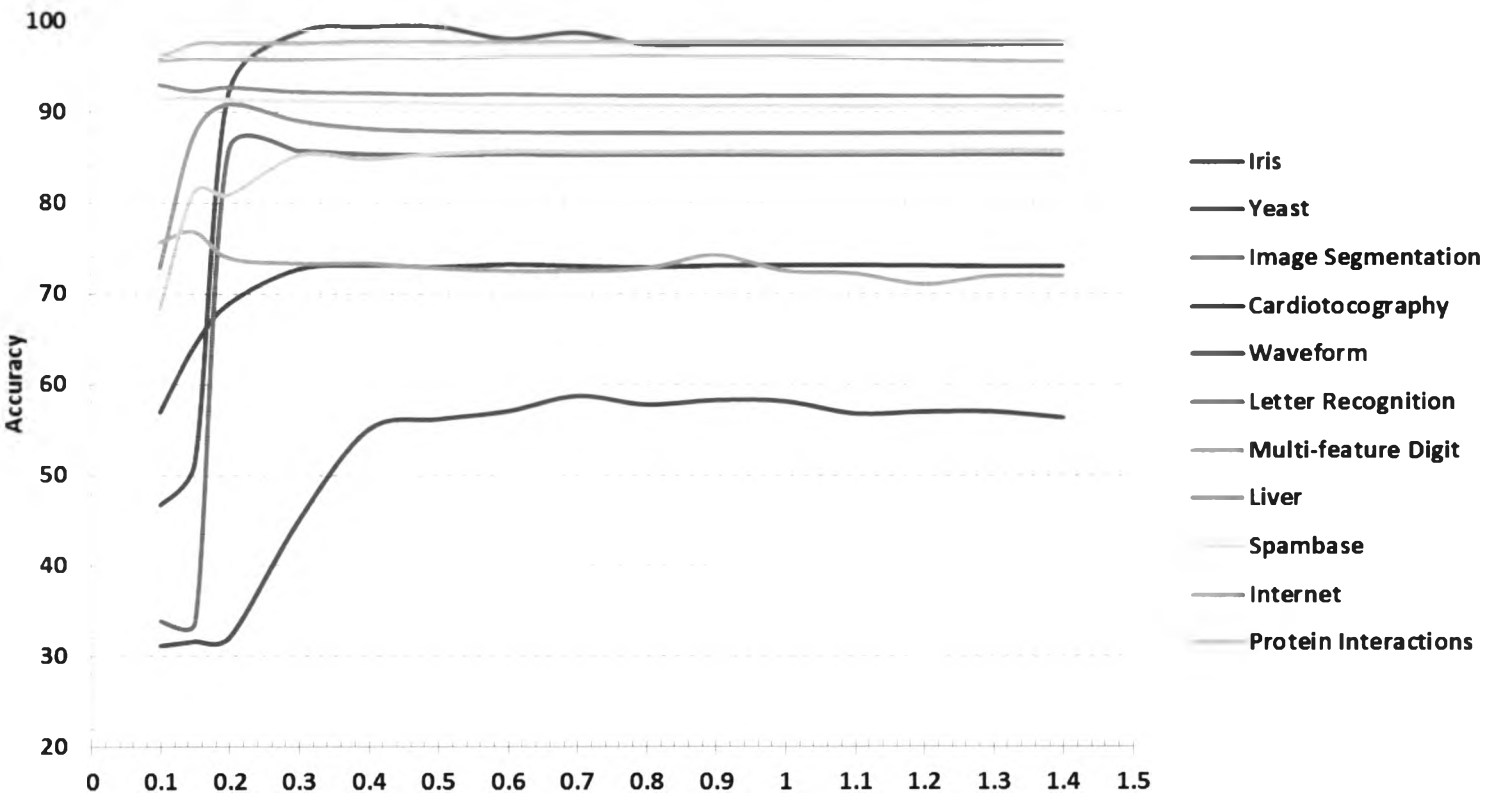

Figure 10: Influence of the predefined δ setting versus accuracy (%)

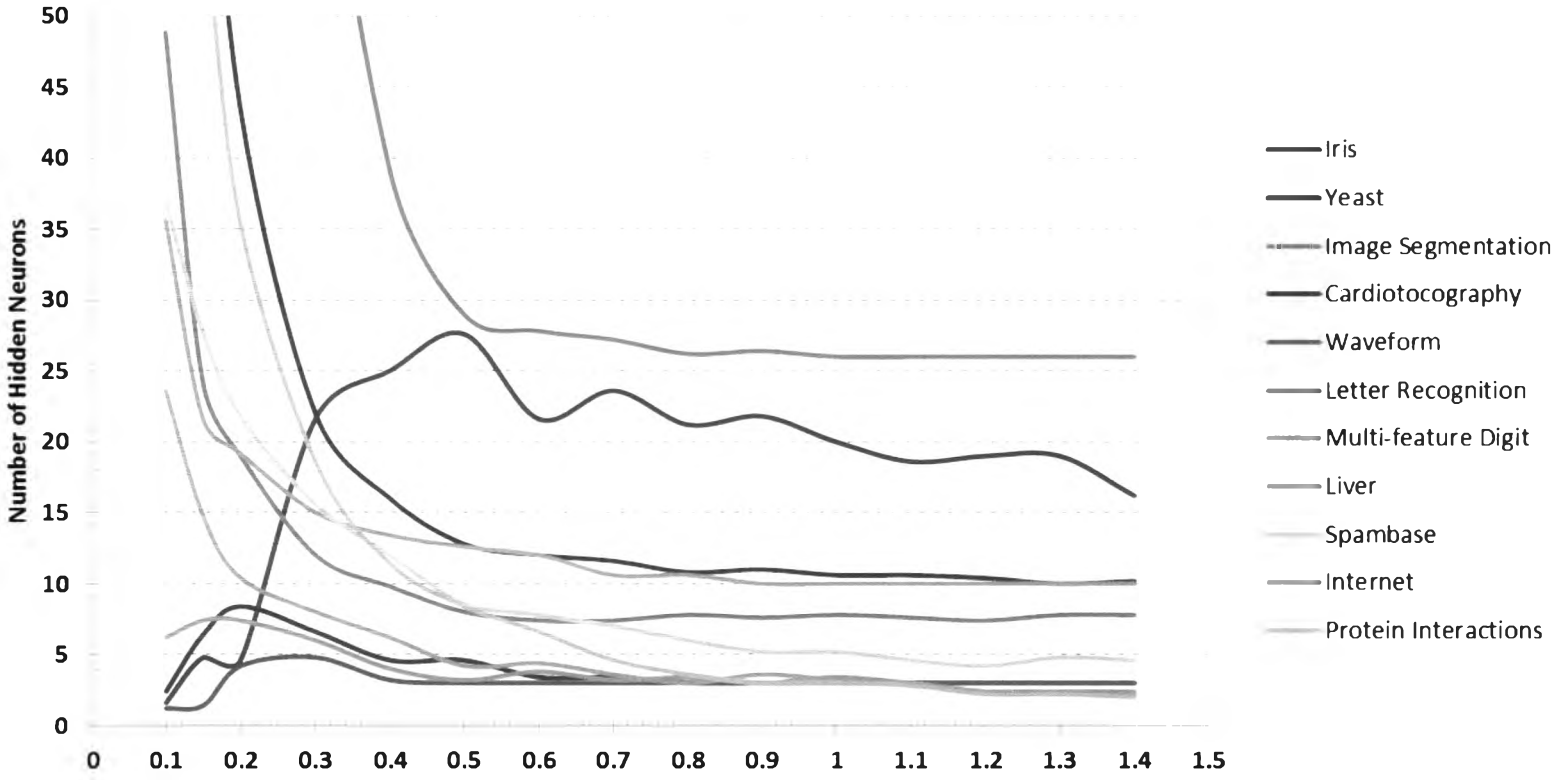

Figure 11: Influence of the predefined δ setting versus the number of hidden neurons

4.2 Experiments for Streaming Training Data Scenario

As complete training data scenario, the experiments for streaming training data chunk scenario were conducted by MATLAB executing on Intel Core i7-2600 CPU 3.40 GHz and 16-GB RAM. The performance of the DLSC was evaluated in terms of percentage of accuracy classification, the number of processing or hidden nodes, and the computational time of learning process. In this case, the results of the proposed method DLSC were compared with four incremental learning methods, namely, Versatile Elliptic Basis Function (VEBF) neural network [4], Incremental Learning Vector Quantization (ILVQ) [29], Chunk Incremental Linear Discriminant Analysis (CILDA) [26] and Robust Incremental Learning methods (RIL) [32] in which the exponential forgetting function is set to 1 for stationary class labels. The 11 real world data sets with various sizes were employed to evaluate the performance of the proposed method. The first nine data sets were also used for complete training data scenario. For all data sets, the instances with missing values were neglected. Data set description is given in Table 10.

Table 10: Description of each data set for streaming training data scenario

Data Set	Number of Attributes	Number of Samples	Size*	Number of Classes	Area
Iris	4	150	600	3	Life
Yeast	8	1,484	11,872	10	Life
Image Segmentation	19	2,310	43,890	7	N/A
Waveform	21	5,000	105,000	3	Physical
Letter Recognition	16	20,000	320,000	26	Computer
Liver	7	345	2,415	2	Life
Spambase	57	4,601	262,257	2	Computer
Internet	1,558	2,359	3,675,322	2	Computer

Advertisement					
Protein Interactions	398	11,188	4,452,824	2	Physical
MiniBooNE	50	130,064	6,503,200	2	Physical
Forest Cover Type	54	581,012	31,374,648	7	Life

Note: Size*= (Number of Attributes)* (Number of Samples)

4.2.1 Experimental setting for streaming training data scenario

For experiment in an incremental environment, five-fold cross validation was used to achieve the appropriate parameters and test the performance of the proposed and compared methods. Each data set was partitioned into five disjoint subsets. For each fold, four subsets were used as a training subset and the rest subset was used as a test subset. Then, the 25 % of the total training subset was randomly selected as the first data chunk for creating an initial model structure. To create the stream of data chunks, *v* data points from the training samples were randomly chosen to create a data chunk where v stand for the number of data in a chunk. A data chunk was repeatedly created until the training data was empty. Moreover, the order of the training data set altered. Then, the test data set was used to test the performance of each model. The parameter N^0 was set to 3 in all data sets except for Yeast data set which was set to 1 because there was a class label having only one datum. The parameter η for updating the width vector was set to 2, while the initial width vector $\mathbf{w}^0 = [w_1^0 \; w_2^0 \; \ldots \; w_n^0]^T$ of VEBF. Thus, the DLSC techniques was slightly change with respect to Equation (31) by

$$w_l^0 = \frac{\delta}{N_1^2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_1} \left\| \mathbf{x}_i - \mathbf{x}_j \right\|, \qquad l = 1, \ldots, n \tag{32}$$

where $\|\cdot\|$ is the Euclidean distance function, N_1 is the number of data in the first chunk and δ is constant. Five-fold cross validation was use to tune the relevant parameter for each method. A parameters setting of DLSC, VEBF, ILVQ, CILDA and RIL techniques are shown in Table 11. For CILDA and ILVQ, 1-nearest neighbor method was used as a classifier.

Table 11: Parameter setting in each data set for streaming training data scenario

Data Set	DLSC (δ)	VEBF (δ)	ILVQ ($\lambda, AgeOld$)
Iris	0.7	0.3	(21,17)
Yeast	0.4	1	(90, 50)
Image Segmentation	0.7	1	(180, 130)
Waveform	0.7	0.7	(70, 110)
Letter Recognition	0.7	1	(80, 100)
Liver	0.15	1	(16,80,)
Spambase	0.4	0.7	(90,18)
Internet Advertisement	0.7	1.2	(200,60)
Protein Interaction	0.7	0.5	(155,60)
MiniBooNE	0.7	0.5	(200,150)
Forest Cover Type	0.05	0.7	(280,180)

4.2.2 Experimental results for streaming training data scenario

The five-fold cross validation was used to evaluate the performance of the models. For each fold, ten distinctive orders of data points in training data set were generated. The classification accuracy, number of hidden or processing neurons and computational time were measured on the test data set as shown in Tables 12-14. The statistical independent $t-test$ was also used to test the difference between the best average value and the other. The value with asterisk (*) means that there is no statistical significance $(\rho \geq .05)$ between the best value and the value of other method on each data set. The best and second average values for each data set were shown in the bold typeface and underline, respectively. Some data sets could not run CILDA and RIL because of singularity problem during finding the weight matrix. The rank average of each method on the number of used experimental data sets was given in the last row for each Table.

Classification accuracy is shown in Table 12. The average accuracy of DLSC is highest on eight data sets in which the average of the DLSC and the other methods is significantly different on six data sets including Image Segmentation, Liver, Letter Recognition, Protein-Protein Interaction, and Forest Cover Type. In the Iris and Yeast data sets, the averages of the DLSC are highest and there is no statistically significant difference between the averages of the DLSC and others as expressed with asterisk (*). For Waveform and MiniBooNE data sets, the accuracy of DLSC method is slightly less than that of RIL method but is greater than the other three methods. For Spambase data set, the accuracy of DLSC method is slightly less than that of CILDA method but is more than the other methods as well. Moreover, the standard deviation of the DLSC is significantly less than that of the other methods in most data sets. This implies that the influence of order of the feed data points in the training process does not affect the accuracy of the proposed DLSC comparing to the other methods. The rank average of DLSC is the best (1.10).

For the number of hidden or processing neurons as shown in Table 13, the average numbers of hidden neurons of CILDA and RIL are equal to the average

number of data points in the training set and the number of class labels, respectively. The average number of hidden neurons of CILDA is the worst for all ten data sets. Although the average number of hidden neurons of RIL is the minimum value for all data sets, the learning process cannot cope with the data with new class label. So, the three methods, DLSC, VEBF and ILVQ, are compared. The number of hidden neurons of DLSC was statistically and significantly less than that of VEBF and ILVQ on eight data sets, namely, Iris, Image Segmentation, Letter Recognition, Waveform, Protein-Protein Interaction, MiniBooNE, Spambase, and Internet. For Liver data set, the average value of number of hidden neurons of ILVQ is the lowest among the three methods but is not statistically different with the proposed DLSC. For Forest Cover Type and Yeast, the average value of number of hidden neurons is the lowest. As classification accuracy, the standard deviation values of the DLSC are significantly less than that of the other methods in almost data sets. This implies that the influence of order of the feed data points in the training process does not affect the number of hidden neurons of the proposed DLSC. The rank average of DLSC is 2.27.

For learning time (s) as shown in Table 14, the taken learning time of CILDA is the fastest for all data set. The CILDA method consumes the learning time for only during updating within-class scatter matrix and between-class scatter matrix. Although the learning time is the lowest, the one drawback of this method is that it took so long time in assigning a class label for a new sample. This is because computing the distance between the new sample and each of all training data set. The learning time of DLSC comes in the second for the nine data sets as shown, except for Liver and Forest Cover Type. The learning time of DLSC is slower slightly than the time of RIL method. For Forest Cover Type, the initial width vector of DLSC is quite small but takes quite long. However, it is the tradeoff between learning time and the accuracy for Forest Cover Type data set. The rank average for learning time of DLSC is 2.36.

Table 12: Comparison results of average with standard deviation $(\overline{x} \pm sd)$ of classification accuracy on eleven data sets.

Datasets	DLSC	VEBF	ILVQ	CILDA	RIL
Iris	**97.47 ± 1.45**	92.13 ± 5.92	95.73 ±4.14*	96.17 ± 3.47*	96.67 ± 0.00
Image Segmentation	**91.77 ± 0.80**	69.27 ±10.52	84.78 ± 1.76	78.48 ± 8.66	83.74 ± 2.11
Liver	**73.33 ± 4.54**	59.77 ± 6.85	60.29 ± 5.61	62.75 ± 6.58	63.35 ± 6.77
Yeast	**56.03 ± 2.48**	42.62 ±12.03	49.63 ± 3.03	25.72 ± 10.77	55.13 ±2.90*
Letter Recognition	**87.62 ± 0.42**	58.64 ± 2.33	80.2 ± 1.17	38.86 ± 3.33	55.51 ± 0.8
Waveform	85.25 ± 0.75	70.79 ±14.19	81.71 ± 1.34	78.21 ± 1.08	**85.87 ± 0.87**
Protein Interaction	**89.31 ± 1.36**	50.28 ± 3.52	59.73 ± 0.67	80.94 ± 0.54	76.26 ± 0.59
MiniBooNE	87.88 ± 0.49	59.65 ±11.44	86.19 ± 0.5	87.58 ± 1.36	**90.07 ± 0.25**
Forest Cover Type	**80.25 ± 1.14**	63.58 ± 0.25	73.98 ± 13.12	51.3 ± 13.12	70.11 ± 0.15
Spambase	90.76 ± 1.01	68.77 ± 7.49	70.92 ± 2.44	**91.47 ± 0.83**	N/A
Internet	**95.93 ± 0.40**	64.3 ± 20.90	89.58 ± 2.42	N/A	N/A
Rank Average	**1.27**	4.45	3.1	3.5	2.22

Table 13: Comparison results of average with standard deviation $(\bar{x} \pm sd)$ of number of hidden neurons on eleven data sets.

Datasets	DLSC	VEBF	ILVQ	CILDA	RIL
Iris	3.76 ± 0.72	4.28±0.98	23.04±9.53	120	3
Image Segmentation	16.96 ± 1.93	19.68 ± 1.57	196.16 ± 56.53	1,848	7
Liver	31.48 ± 5.55	47.84 ± 4.5	27 ± 15.62	276	2
Yeast	54.56 ± 7.93	19.08 ± 1.91	149.36 ± 72.21	1,187.4	10
Letter Recognition	30.36 ± 3.34	235.44 ± 14.17	670.48 ± 51.47	16,000	26
Waveform	3.16 ± 0.47*	5.52 ± 2.93	177.84 ± 71.3	4,000	3
Protein Interaction	8.56 ± 3.08	37.48 ± 13.43	190.2 ± 59.39	895.06	2
MiniBooNE	78 ± 7	2,691 ± 423	2,285 ± 43	10,4051.2	2
Forest Cover Type	2,830 ± 248	88 ± 4	1,550 ± 90	464,809.6	7
Spambase	**13.8 ± 2.43**	20.04 ± 1.95	137.44 ± 27.27	3,681.2	N/A
Internet	**7.8 ± 1.59**	18.72 ± 2.48	137.04 ± 47.56	N/A	N/A
Rank Average	2.27	2.81	3.64	5	1

Table 14: Comparison results of average with standard deviation $(\bar{x} \pm sd)$ of learning time with standard deviation on eleven data sets.

Datasets	DLSC	VEBF	ILVQ	CILDA	RIL
Iris	0.02 ± 0.004	0.04 ± 0.004	0.07 ± 0.005	**0.003 ± 0.000**	0.02 ± 0.002
Image Segmentation	1.17 ± 0.01	1.23 ± 0.08	5.77 ± 0.59	**0.03 ± 0.01**	27.26 ± 6.3
Liver	0.15 ± 0.05	0.36 ± 0.04	0.19 ± 0.03	**0.007 ± 0.009**	0.1 ± 0.03
Yeast	0.23 ± 0.06	0.54 ± 0.07	2.43 ± 0.43	**0.02 ± 0.006**	5.91 ± 1.12
Letter Recognition	0.88 ± 0.1	18.68 ± 0.84	109.78 ± 4.61	**0.16 ± 0.02**	493 ± 145
Waveform	0.32 ± 0.02	2.37 ± 0.6	11.93 ± 1.76	**0.07 ± 0.01**	33.37 ± 8.7
Protein Interaction	21.14 ± 3.73	2,266 ± 605	47.25 ± 2.13	**6.99 ± 0.38**	5,624 ± 542
MiniBooNE	65 ± 20	936 ± 106	603 ± 59	**2.74 ± 0.08**	1,673 ± 153
Forest Cover Type	202,913 ± 60,915	2,451 ± 86	38,034 ± 465	**69 ± 6**	27,536 ± 1,395
Spambase	1.24 ± 0.59	19.93 ± 1.24	8.28 ± 0.48	**0.18 ± 0.05**	N/A
Internet	257 ± 51	29,229 ± 1967	**33.5 ± 1.43**	N/A	N/A
Rank Average	2.36	3.36	3.45	**1**	4.11