Int J Pharm Bio Sci 2014 July; 5(3): (B) 728 - 740

Research Article **Bioinformatics**

International Journal of Pharma and Bio Sciences **ISSN 0975-6299**

PERFORMANCE EVALUATION OF SUPPORT VECTOR MACHINE - NEAREST NEIGHBOR CLASSIFIER FOR DIABETES DATASET

D. MAHALAKSHMI[*1] AND S.P. CHOKKALINGAM[2]

*[*1]Post Graduate Student, Sri Venkateswara College of Engineering, Anna University, Chennai*
[2]Associate Professor, Saveetha University, Anna University Research Scholar, Chennai

ABSTRACT

In this current scenario the automatic classification has been used for many applications such as indexing for information retrieval system, search engines, document organization and text filtering. Classification is one of the widely used techniques in the machine learning. It is a mechanism of grouping the data according to the predefined class labels. The popular classification algorithms are K-Nearest Neighbor algorithm (K-NN) and Support Vector Machine (SVM) algorithm. K-NN is a lazy learning method where classification is done by comparing the feature vectors of different points. K-NN is popular due to its simplicity and efficiency, but complexity is in finding the k value as a small k value will result in obtaining lesser information from training data. Support Vector Machine is another algorithm where an optimal hyperplane is chosen for classifying the diabetes dataset. SVM provides high accuracy and found to be popular in high dimensional data space. The hybrid classification algorithm is used to build the classification model using SVM-NN classifier is proposed. In this proposed SVM-NN classifier the impact of k parameter is reduced by considering only support vectors in order to classify the data. In SVM-NN Manhattan distance measure is used to compute the distance between the test samples and support vectors. The test samples can be compared to the class labels of the original class labels and performance can be evaluated using the confusion matrix. This proposed SVM-NN algorithm can reduce the size of the training samples and also greatly reduces the classifying time, so it can be used for large data sets. The experimental result shows that SVM-NN gives the best performance for Diabetes dataset.

KEYWORDS – Classification, K-Nearest Neighbor, Support Vector Machine, Manhattan Distance Measure.

D. MAHALAKSHMI
Post Graduate Student, Sri Venkateswara College of Engineering, Anna University, Chennai

***Corresponding author**

1.INTRODUCTION

Machine Learning is a common learning problem where the algorithm are learns from the set of instances and classifies the new instances to a class labels from the set of trained class labels. Classification is the one of the technique used in supervised learning. Supervised learning has a pair consisting of both input value and desired expected labels which is used to classify the unknown data. This method works fast and accurate. Classification is the process of grouping the data based on the predefined trained class labels.

The steps for classification:

- **Learning step:** Every sample is assumed to a predefined class is known as class label. The set of samples with class label used for model construction is known as training set. By applying the classification algorithm the classification rules are generated.
- **Classification step:** Here unknown sample is given as test data to the classification rules and predict the class label for the unknown samples. For classifying data, one can use the classifier technique are,

A.K-Nearest Neighbor

K- Nearest Neighbor is a non parametric instanced based lazy learning algorithm. KNN has been used in statistical estimation and pattern recognition. The principle behind the KNN algorithm is that to identify the data point which is nearest to the k value, where k is the integer value used to classify the data points. If the value of k is small, KNN performs moderately and if k value is high the classifier accuracy is more. But the threshold is exceeded, the accuracy of the classifier decreases. So the k value needs to be fixed approximately. Normal the k value to be fixed in the range of 3 to 10 values.

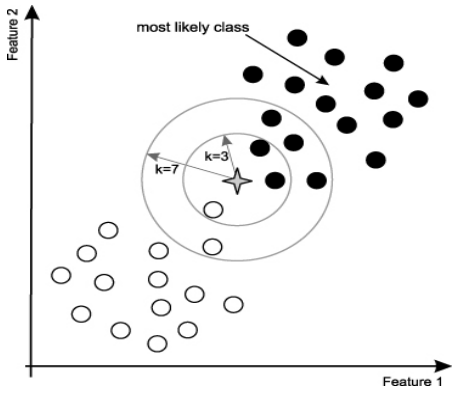


Figure 1
K-NN Classifier Representations

In the above Figure 1 show the unknown data is represented as star, the two class labels are represented in shaded circles and un-shaded circles. The objective is to find the class label for the new input data. The advantages of K-NN are simple to implement, since training sample is small. The disadvantages of K-NN are dependence on k- value and difficult to implement, when training samples are large.

B. SUPPORT VECTOR MACHINE

A Support Vector Machine is the most powerful supervised learning algorithms which is used to analyze the data and also used for classification process. The implementation of SVM is the concept of Structural Risk Minimization (SRM) in order to generate the linear separable hyperplane from a set of given training samples which gives the low classification errors. The SVM classifier algorithm builds a classifier model which assigns the new unknown test samples into either one of the class labels.

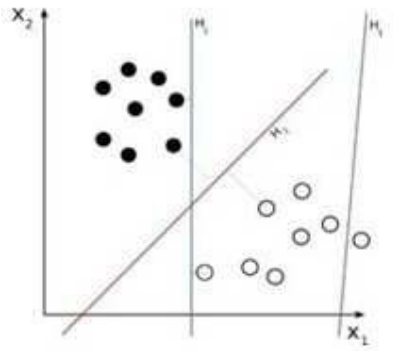


Figure 2
SVM Classifier Representations

Figure 2 shows the example SVM model representation. The data points are grouped into two categories represented as black circles and white circles. Here we have three hyper planes H1, H2 and H3 on the two axes x1 and x2. H3 cannot separate the class into two categories. H1 has a small margin and the maximum margin has H2 with linear separable by hyperplane. The formula is to calculate the linear separable hyperplane is given as,

$$w.x + b = 0 \qquad (1)$$

In the above equation 1 explains that separating the optimal hyperplane with high accuracy, where 'w' is a weight vector, $w = \{w_1, w_2, \ldots\ldots, w_n\}$, n is the number of attributes. 'x' is the values of attributes in training set and 'b' is a scalar value. The good optimal separating hyperplane is considered as the maximum margin between the different set of classes. Margin can be calculated as,

$$d1 + d2 = \frac{2}{||w||} \qquad (2)$$

In the equation 2 represents the sum of distance from two set of classes, where d1 is the distance of one classes of hyperplane to nearest data samples and d2 is the distance of another set of classes of hyperplane to nearest data samples. The nearest data points are located in a maximum distance to hyperplane are called as Support Vectors (SV_s). The SVM is used for many real time applications and gives the high accurate classifier. It does not occur the over fitting because the separation of margin is equal between the SV_s and optimal hyperplane. The advantages of SVM are handling the high dimensional data and more accurate. In the proposed work a novel approach is created by integrating the Support Vector Machine - Nearest Neighbor (SVM-NN) and take advantages of the accuracy of SVM classifier and simplicity of nearest neighbor to build a classifier model for classification. The using of SVM classifier is to select the support vectors (SVs) and to minimize the size of training samples. And the applying the nearest neighbor classifier in the trained dataset, where finding the shortest distance between the unknown class labels of samples and trained class labels. And then group the unknown test samples to the particular category. Finally evaluate the performance of the classifier approach.

II. RELATED WORK

Han et al (1999) proposed a model Weight Adjusted K-Nearest Neighbor Classifier algorithm was proposed to overcome the problem of learns weights for different features. In the weight adjustment step, the weight of each feature is divided into small steps to improve the classification decision. The feature with the most objective function is identified and their corresponding weight is

updated. The drawback of using enhanced weight adjustment is over fitting. Min Ling et al (2005) proposed a novel method using the Multi-Label learning using K-NN classifier algorithm. In Multi-label learning, training sample is taken as the input and predict the class labels for unknown test samples. First, identify the k nearest neighbor samples and predict the class labels for unknown samples using maximum a posteriori (MAP).The limitation is performance dependent on the k value and lesser k value, the accuracy of the algorithm will be minimal. Geng et al (2008) proposed a K-Nearest Neighbor (K-NN) classifier used for information retrieval using query dependent ranking. K-NN uses a method called query-dependent ranking which is used to train a model and implement different ranking models for various types of queries. This technique uses two methods to rank the information based on the query. An online method is used as first that generates a ranking model for a given query by using the labeled neighbors in the query feature space. The Documents are ranked based on the query using the created model. Second is an offline method that creates the ranking models to increase the effectiveness of ranking. The experimental result showed that online and offline both perform well using single ranking function. The drawback of using KNN in the training is that it takes much higher time complexity. Lee et al (2011) proposed a model for text document classification using support vector machine and Euclidean distance is used for the classification decision. The SVM classifier generates a decision making using the linear separable hyperplane. The kernel function is used to map the low dimensional of input data points into the high dimensional space, so that it can be separated by a linear hyperplane. On the other hand, the parameter of soft margin; C is an important parameter to determining the performance of the SVM Classifier. To overcome the value of parameter of C, the Euclidean distance is used to compute the distance between the unknown test samples and trained class labels.

III. PROPOSED SYSTEM

The objective of the proposed work is to build the classifier model using Support Vector Machine-Nearest Neighbor (SVM-NN) classification algorithm for diabetes dataset. Here the dependency of K value is minimized and enhancing the KNN by introducing the SVM Classifier. The Support Vector (SV_s) is taken from the SVM trained model and minimizes the training data. This greatly reduces a time and also performs the better classification accuracy. The Manhattan (CityBlock) distance measure is used to compute the distance between the unknown test samples and support vectors is chosen from the each category.

IV.SYSTEM DESIGN

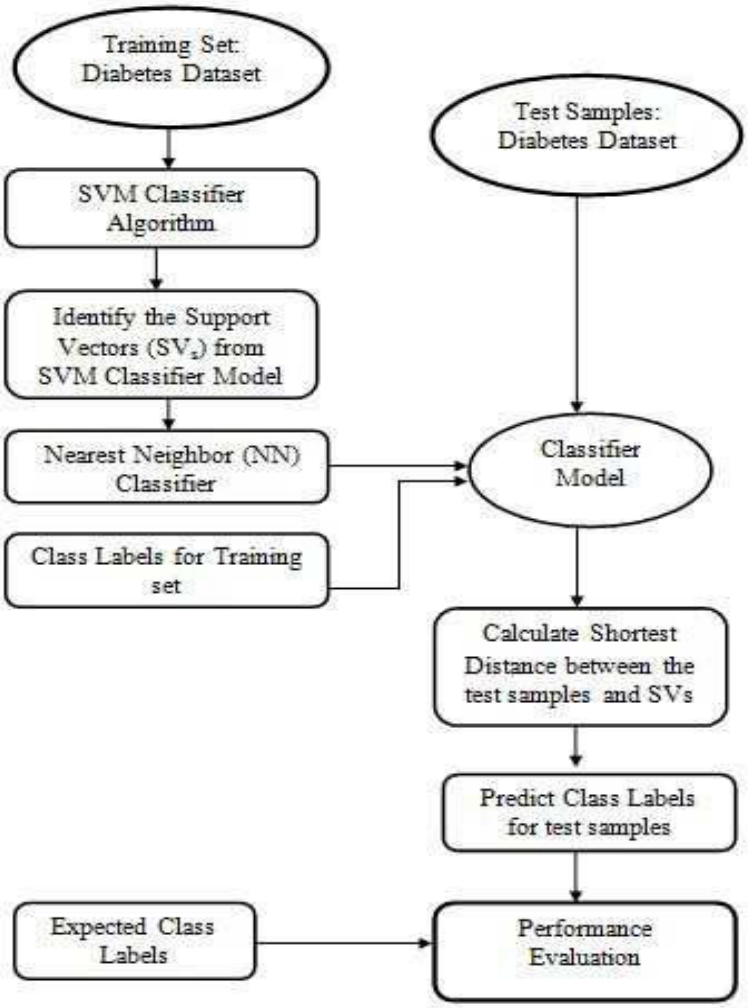


Figure 3
System Model

The above Figure 3 shows the process of system model for SVM-NN classification approach.

A. SUPPORT VECTOR MACHINE CLASSIFIER

In the training set the Diabetes dataset is taken as the input for classification model. Training a dataset is a collection of data along with a class label to generate the classifier model to predict the test data. In multidimensional dataset, data points are distribution in nature. Here diabetes dataset is taken as a multidimensional dataset. Hence, low dimensional dataset is converted into high dimensional with the help of kernel functions. The SVM model includes the number of support vectors, class labels, kernel type, and bias value of separating the hyperplane. To classify the new test data, predict the class label with the generated model and calculate the accuracy of the SVM classifier. The SVM gives a good accurate result with multi dimensional dataset. To take the advantages of SVM and KNN a hybrid approach is generated using SVM-NN and to minimize the parameter of k value. The Linear Kernel function is calculated as,

$$k(X_i, X_j) = X_i^T X_j \qquad (3)$$

In above equation 3 shows that dot product between vectors, where X_i, X_j is the two dimensional vectors.

B. SUPPORT VECTOR MACHINE –NEAREST NEIGHBOR

Here multi dimensional dataset is trained with the SVM classifier. The Support vectors are chosen from the SVM trained model is taken as the input for test data. And apply the nearest neighbor concept to classify the test data with trained model.

TABLE 1
Procedure for SVM-NN Classifier

Procedure for Support Vector Machine – Nearest Neighbor (SVM – NN) Classifier
Step1: Load the dataset and divides the data into Training set T= {(x_1, y_1), (x_2, y_2),.... (x_n, y_n)}; where 'x' is an attributes of dataset and 'y' is an label and test data X = { x_1, ..., x_n} to classify, where 'n' is an number of values.
Step 2: Apply the SVM Classifier algorithm to train the data and then identify the number of Support Vectors (SV_s) from the Training set.
Step 3: New Test data 'X 'is classified with the SV_s are used as a Training set and apply the Nearest Neighbor (NN) classifier technique in which Manhattan is used to calculate shortest distance between the new test data and SV_s.
Step 4: Identify the class label L = {1, -1} for new test data using NN.
Step 5: Performance is evaluated for SVM-NN classifier using the predicted class labels for new test data and expected class labels using confusion matrix.

The above table 1 shows the procedure for the SVM-NN classifier model. From the SVM-NN classifier model, classifies the new unknown class labels test samples from the generated model and distance between the test samples and with all the support vectors are computed. The Manhattan (CityBlock) distance measure is used to compute the distance between the unknown test samples with the trained class labels. Classify the test samples with the minimum distance between the any one of the class labels. They are Manhattan distance measure used for classifier to compute the minimum distance.

$$\text{Manhattan Distance} = \sum_{i=1}^{n} |x_i - y_i| \qquad (4)$$

In the above equations 4 says that x and y represents the data points with 'n' different attributes $\mathbf{x} = (\mathbf{x}_1, \mathbf{x}_2, \ldots, x_n)$ and a point $\mathbf{y} = (\mathbf{y}_1, \mathbf{y}_2, \ldots, y_n)$.

C.PERFORMANCE EVALUATION

The performance evaluation is done with the test data using the classification model which is obtained from the training set. The test data are compared with the class labels of the expected class labels and performance is evaluated using the confusion matrix. The confusion matrix is used to visualize the

performance of the classifier which is specified in a table layout. It is also called as contingency table. The predicted class is represented in each column and actual instance of the class is represented in each row. By using this clearly understand the misclassification between the two classes.

V. IMPLEMENTATION RESULT

The performance evaluation of SVM-NN classifier is tested with the Diabetes Dataset. The input dataset is taken as the standard benchmark Diabetes dataset which contains the 768 samples with 8 attributes along with class labels and it is collected from the source of http://archive.ics.edu/ml/datasets/Pima+Indians+Diabetes. The description of diabetes dataset is the patients are mostly females candidates at the age of 21 years old. The attributes includes that Plasma glucose level, Diastolic blood pressure, number of times pregnant, Triceps skin fold thickness, Serum insulin, Body mass index, Diabetes pedigree function, Age and class labels. It has no missing values in the Diabetes dataset. The Diabetes dataset is taken as the excel file format and it is loaded in a classifier algorithm. The entire Diabetes dataset is split into training samples and it is used to generate a model and test samples are applied to predict the class labels and compute the performance of the SVM-NN classifier.

A. SVM-NN IMPLEMENTATION RESULT

The SVM-NN classifier is implemented in the MATLAB (2012a) version 7.14.0.739 with additional SVM toolbox is used to build the classifier model. The SVM toolbox is used as LIBSVM. First construct the SVM classifier using LIBSVM tool to select the support vectors from the training samples and to minimize the size of training samples from given diabetes dataset. Here, the linear kernel function is used to convert the low dimensional feature space into high dimensional feature space for linear separable data to train the model. And test data is applied with the identified SV_s and distance between the unknown class label test samples with the trained classifier of the SVM model as SV_s.

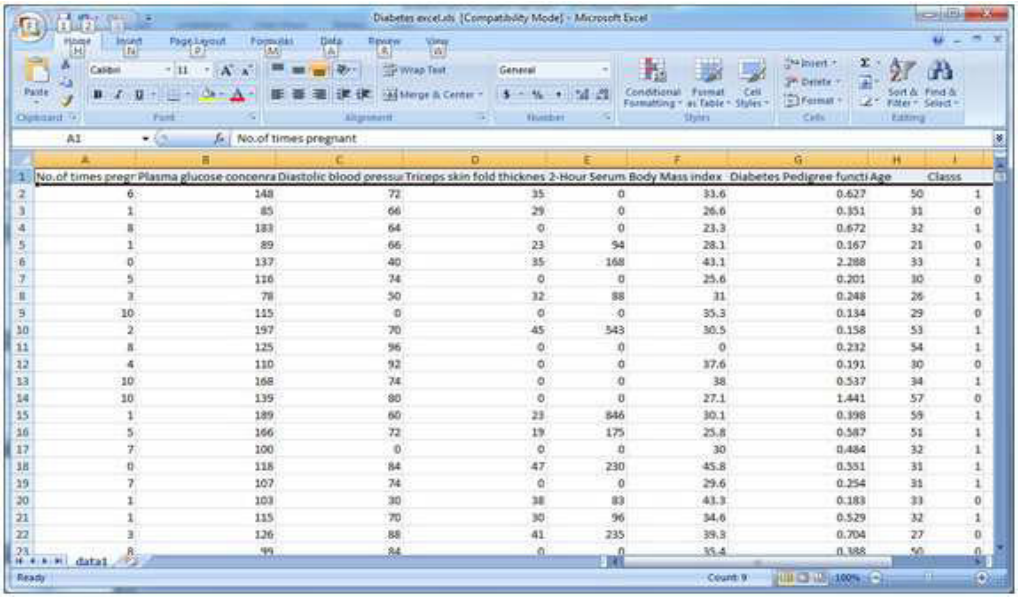


Figure 4
Input data as Diabetes Dataset

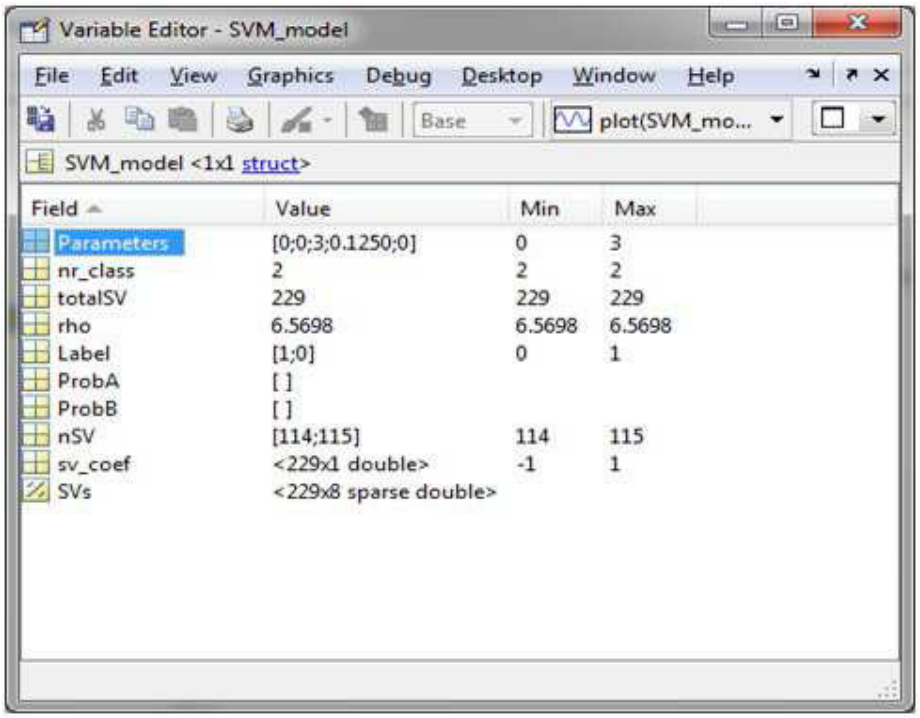


Figure 5
Model Generated from SVM classifier using LIBSVM Tool

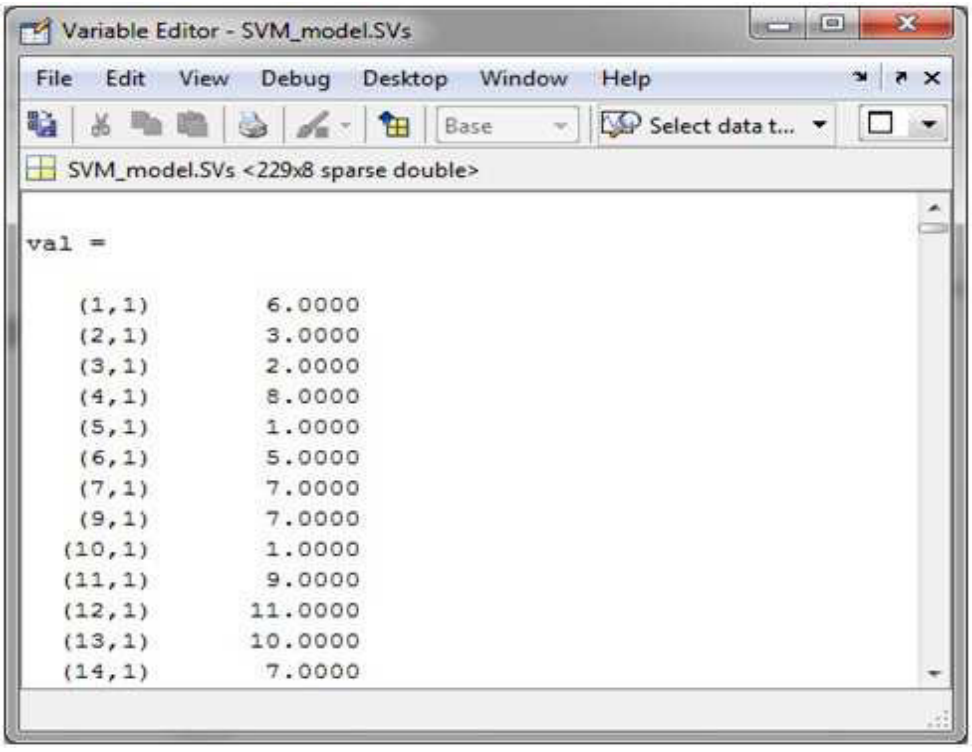
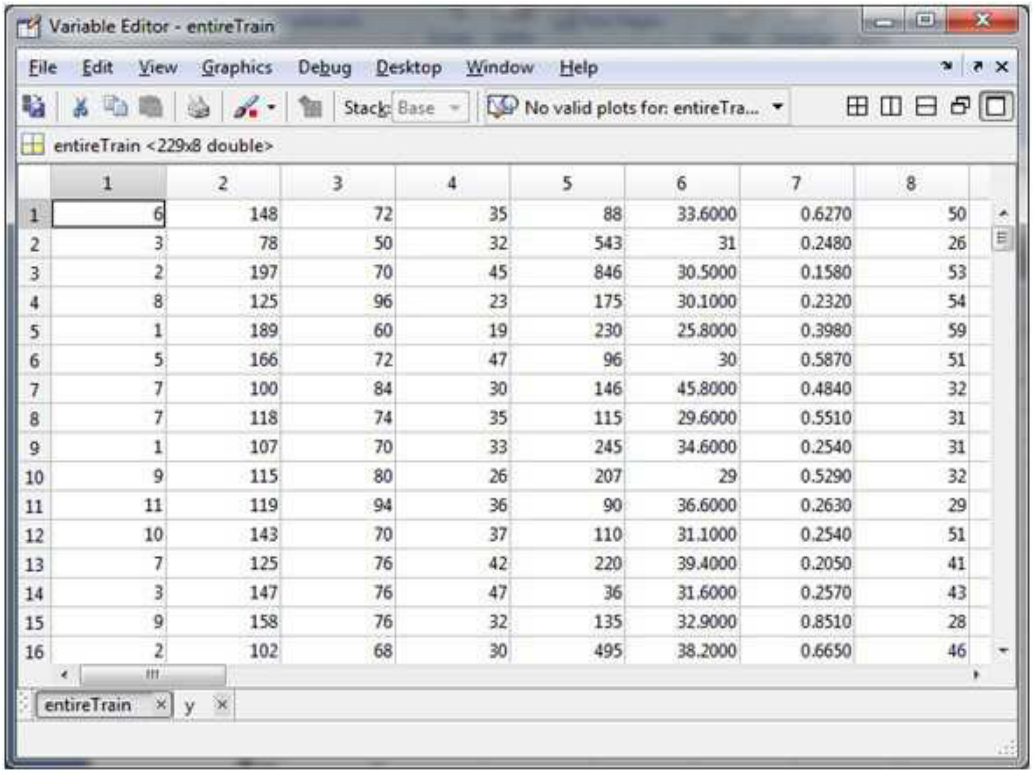
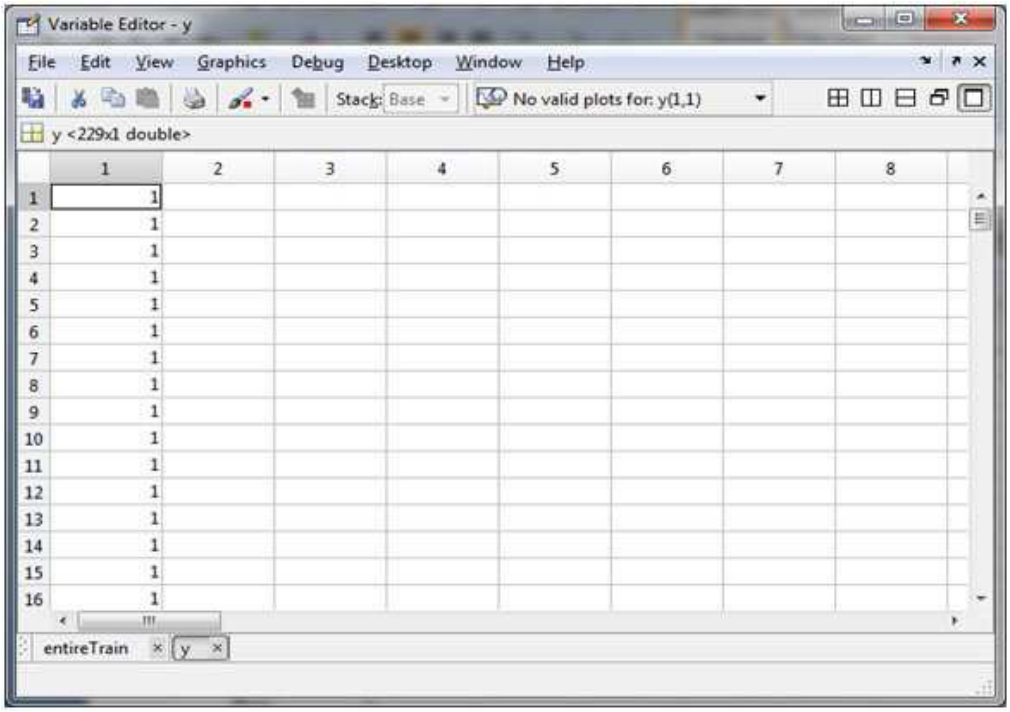


Figure 6
Support Vectors are identified from Training Samples



	1	2	3	4	5	6	7	8
1	6	148	72	35	88	33.6000	0.6270	50
2	3	78	50	32	543	31	0.2480	26
3	2	197	70	45	846	30.5000	0.1580	53
4	8	125	96	23	175	30.1000	0.2320	54
5	1	189	60	19	230	25.8000	0.3980	59
6	5	166	72	47	96	30	0.5870	51
7	7	100	84	30	146	45.8000	0.4840	32
8	7	118	74	35	115	29.6000	0.5510	31
9	1	107	70	33	245	34.6000	0.2540	31
10	9	115	80	26	207	29	0.5290	32
11	11	119	94	36	90	36.6000	0.2630	29
12	10	143	70	37	110	31.1000	0.2540	51
13	7	125	76	42	220	39.4000	0.2050	41
14	3	147	76	47	36	31.6000	0.2570	43
15	9	158	76	32	135	32.9000	0.8510	28
16	2	102	68	30	495	38.2000	0.6650	46

Figure 7
Training Samples for SVM-NN classifier



	1	2	3	4	5	6	7	8
1	1							
2	1							
3	1							
4	1							
5	1							
6	1							
7	1							
8	1							
9	1							
10	1							
11	1							
12	1							
13	1							
14	1							
15	1							
16	1							

Figure 8
Class Labels for Training Samples

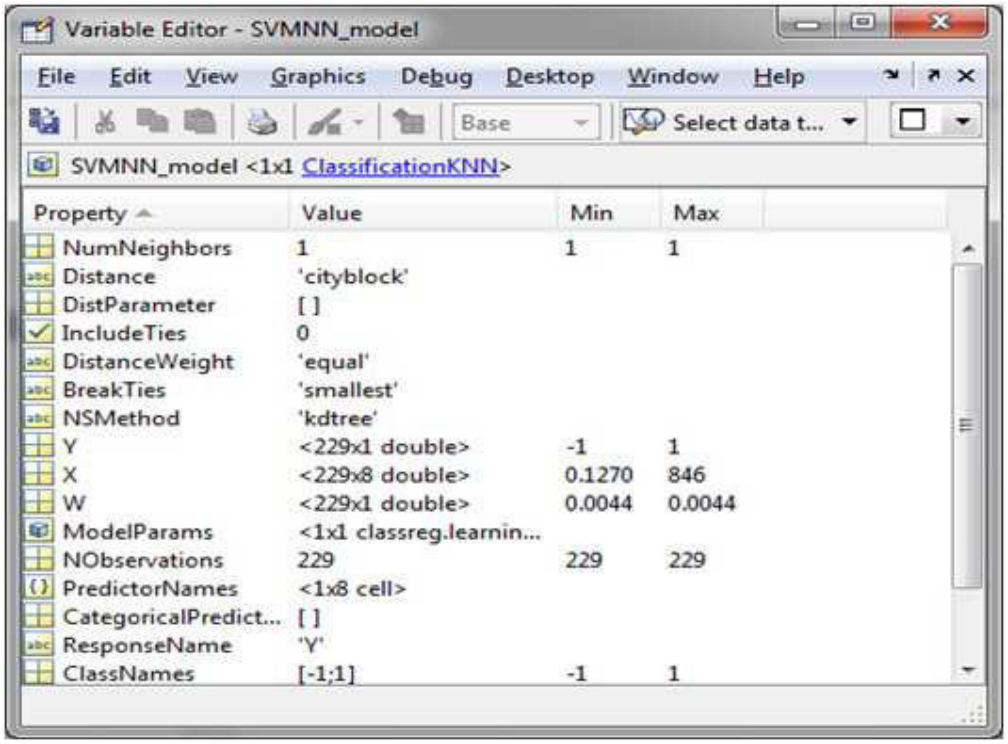
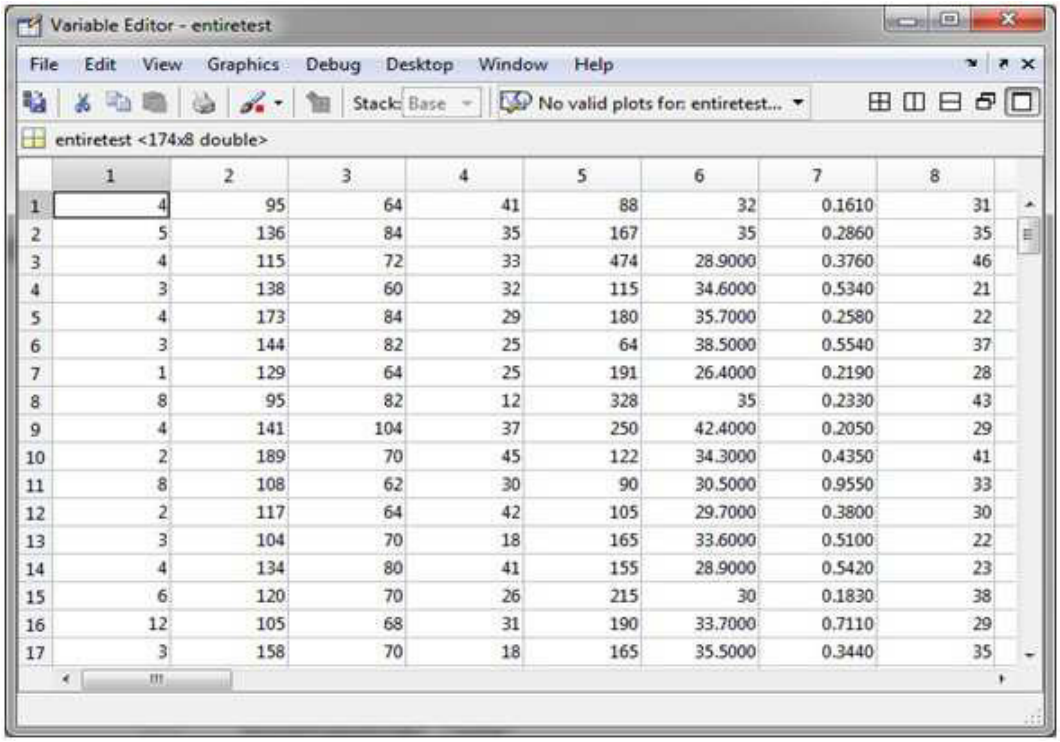


Figure 9
Model Generated for SVM-NN Classifier



	1	2	3	4	5	6	7	8
1	4	95	64	41	88	32	0.1610	31
2	5	136	84	35	167	35	0.2860	35
3	4	115	72	33	474	28.9000	0.3760	46
4	3	138	60	32	115	34.6000	0.5340	21
5	4	173	84	29	180	35.7000	0.2580	22
6	3	144	82	25	64	38.5000	0.5540	37
7	1	129	64	25	191	26.4000	0.2190	28
8	8	95	82	12	328	35	0.2330	43
9	4	141	104	37	250	42.4000	0.2050	29
10	2	189	70	45	122	34.3000	0.4350	41
11	8	108	62	30	90	30.5000	0.9550	33
12	2	117	64	42	105	29.7000	0.3800	30
13	3	104	70	18	165	33.6000	0.5100	22
14	4	134	80	41	155	28.9000	0.5420	23
15	6	120	70	26	215	30	0.1830	38
16	12	105	68	31	190	33.7000	0.7110	29
17	3	158	70	18	165	35.5000	0.3440	35

Figure 10
Test samples to predict class labels

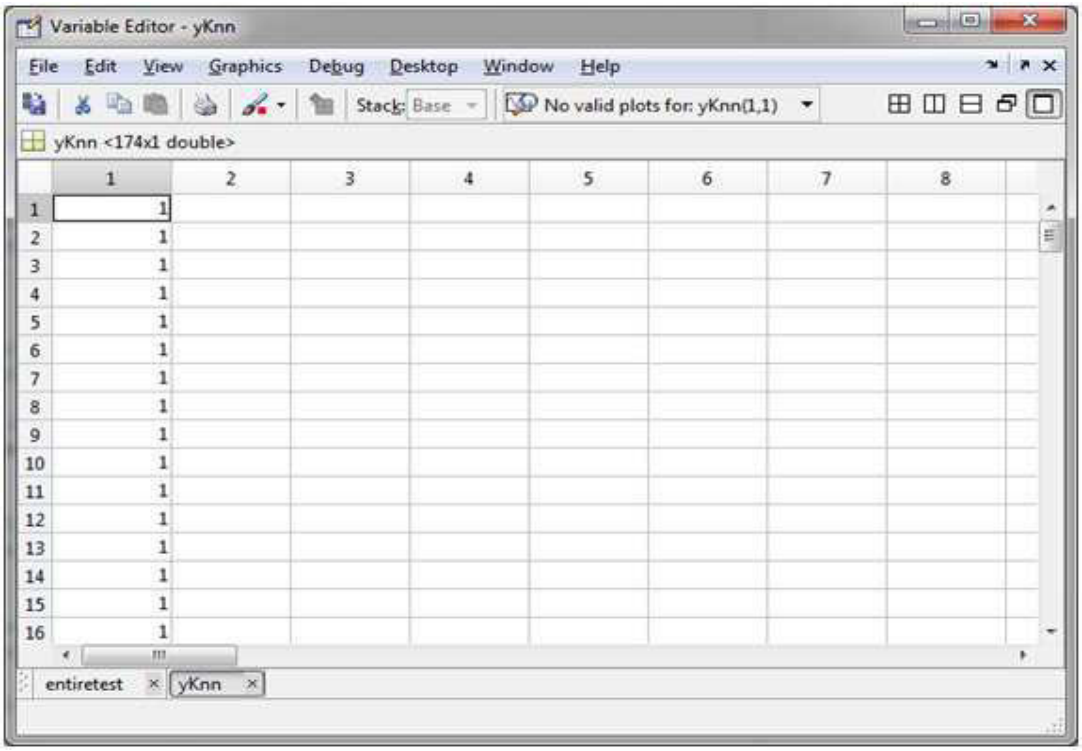


Figure 11
Class Labels predicted for test samples

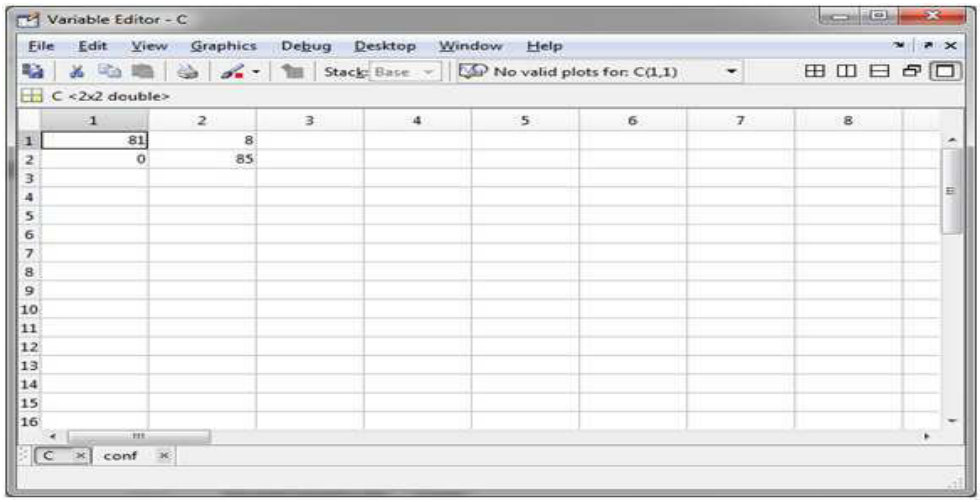


Figure 12
Confusion Matrix for SVM-NN classifier for Diabetes Dataset

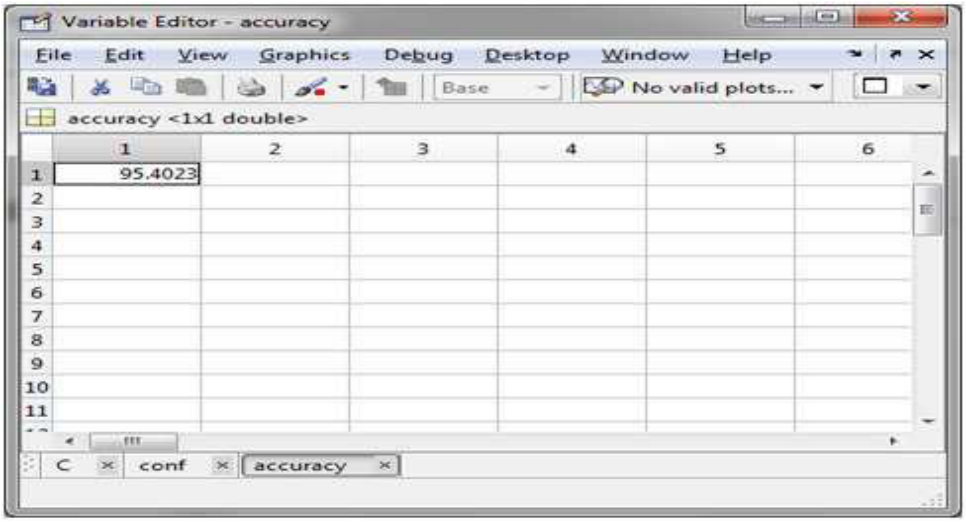


Figure 13
Performance Evaluation of SVM-NN classifier

$$Accuracy = \frac{(TP + TN)}{(TP + FP + FN + TN)} \quad (5)$$
$$= \frac{(81 + 85)}{(81 + 8 + 0 + 85)}$$
$$= 95.40\ \%$$

The above equation 5 shows the formula for classifier accuracy. Figure 4 shows the Diabetes dataset is taken as input data, Figure 5 represents the model generated for SVM classifier using LIBSVM tool, Figure 6 shows that support vectors are identified from the training samples, Figure 7 shows the training samples for SVM-NN classifier, Figure 8 represents the class labels for training samples, Figure 9 shows that model generated for SVM-NN classifier based on Figure 7 and 8, Figure 10 represents the test samples, Figure 11 class labels predicted for test samples, Figure 12 confusion matrix for SVM-NN classifier for diabetes dataset, Figure 13 shows the accuracy for SVM-NN classifier. The highest classification accuracy is achieved with SVM-NN of 95.4 %.

VI. CONCLUSION

The hybrid approach for diabetes data classification builds the classifier that incorporates the SVM and K-NN classification algorithms. The proposed SVM-NN algorithm efficiently minimizes the size of training data points and used for large dataset. The performance evaluation of the SVM-NN approach using Manhattan distance measures in order to reduces the classifying time for high dimensional datasets. The performance of hybrid approach using the diabetes dataset performs well and gives higher classification accuracy. The highest classification accuracy is achieved with SVM-NN of 95.4%. The performance of the classifier is evaluated by confusion matrix evaluation technique. The future extension can be experimenting with using the different kernel functions for better accuracy.

REFERENCES

1. Blanzieri, E. & Bryl, A, "Instance-based span filtering using SVM nearest neighbor classifier", In Proceedings of the 20th international florida artificial intelligence research society conference, May 7-9, Key West, Florida,USA, pp.441-442, (2007).
2. Geng, X., Arnold, A., Qin, T., Liu, T., Li, H., & Shum, H, "Query dependent ranking using K-Nearest Neighbor", Annual ACM Conference on Research and Development in Information Retrieval. In Proceedings of the 31st annual international ACM SIGIR conference on research and development in information retrieval, pp. 115-122, (2008).
3. Lee, L. H., Isa, D., Choo, W. O., & Chue, W.Y, "High relevance keyword extraction facility for Bayesian text classification on different domains of varying characteristic", Expert Systems with Applications, doi: 10.1016, (2011).
4. Han, E. H., Karypis, G. & Kumar, V, "Text categorization using weighted adjusted K-Nearest Neighbor classification", Technical Report, Department of Computer Science and Engineering , Army HPC Research center, University of Minnesota, Minneapolis, USA, (1999).
5. Joachims, T, "Text categorization with support vector machines: learning with many relevant features", In Proceedings of the 10th European conference on machine learning (ECML-98), pp.137-142, 1998, (2008).
6. Lee, L. H., Chin Heng Wan., Rajprasad Rajkumar, Dino Isa," A Hybrid text classification approach with low dependency on parameter by integrating K- nearest neighbor and support vector machine ", Expert systems with Applications 39, 11880-11888, (2012).
7. Lee, L. H., Isa, D., Choo, W, O., & Chue, W. Y, "High relevance keyword extraction facility for Bayesian text classification on different domains of varying characteristic", Expert Systems with Applications, doi:10.1016/j.eswa.2011.07.116, (2011).
8. Min-Ling Zhang, Zhi-Hua Zhou, "Ml-KNN: A Lazy Learning Approach to Multi-label Learning", National Laboratory for Novel Software Technology Nanjing university, Nanjing 210093, (2005).
9. Osuna, R. G. Lecture Notes CS 790: Introduction to pattern recognition, Dayton, Ohio, USA: Wright State University.
10. K.P. Soman, Shyam Diwakar, V.Ajay, "Insight into Data Mining Theory and Practice", Publisher, PHI Learning Private Limited, (2006).
11. Lee, L.H., Wan, C, H., Rajkumar, R., & Isa, D., "An Enhanced support vector Machine Classification framework by using Euclidean distance function for text document categorization", Applied Intelligence.DOI:10.1007/s10489-011-0314-z, (2011).