Global Journal of Engineering and Technology Advances, 2020, 05(03), 020–028

Global Journal of Engineering and Technology Advances

Cross Ref DOI: 10.30574/gjeta

Journal homepage: http://www.gjeta.com

(RESEARCH ARTICLE)

Diagnostic of pathology on the vertebral column machine learning - Cluster K-nearest Neighbor (CKNN) part (I)

Aissa Boudjella [1, 2, 3, *], Sarah Arab [3], Manal Y. Boudjella [4], Sarah Khiter [3] and Bachir Bellebna [3]

[1] *Bircham international University, Avda Sierra-2 (Urb. Guadamonte), Villanueva de la Cañada - Madrid 28691, Spain.*
[2] *Bircham international University .1221 Brickel Av., Suite 900 - Miami, Florida 33131 – USA.*
[3] *Etablissement Hospitalier Universitaire d'Oran, 1 Novembre 1954, Oran, Algeria.*
[4] *University of Sciences and Technology of Oran Mohamed Boudiaf ,USTO-MB, BP 1505 El M'naouar, 31000 Oran, Algeria.*

Publication history: Received on 25 November 2020; revised on 04 December 2020; accepted on 06 December 2020

Article DOI: https://doi.org/10.30574/gjeta.2020.5.3.0107

Abstract

In this investigation, we have developed a graphical user interface application to perform the diagnostic of pathology on the column vertebral based on the Cluster K-Nearest Neighbor (CKNN) classifier. The system is implemented and simulated in Anaconda, and its performance is tested on real dataset that contains 6 features and two (02) classes. Each class, abnormal and normal class consists of 210 instances, and 100 instances, respectively. A comparison of the performance of the test measurement under various test sizes (10%~50%) is carried out to predict the class label when the nearest neighbor k changes from 1 to 19. The results show that the accuracy depends on both independent parameters, the test size and k-neighbors, which gives better training accuracy than the test accuracy, in the range of [82.5% ~ 100%] and [70%~84%], respectively. When k varies from 1 to 4, a higher training accuracy, larger than 90% is observed. While the test set shows a low accuracy in the range of [74% ~ 82.5%]. Increasing the test size or/and k, does not affect significantly the accuracy. When k is larger 1, the training accuracy is approximately equal to 0.925±0.05, the test accuracy (except for k=6 and 17) is about 0.79±0.05. The prediction of the class status maybe optimized by combining the dataset set size with the k-neighbors parameters. The GUI can be useful to help the medical doctors to diagnostic the patient effectively to take a rapid decision and predict results in a reduced time lapse.

Keywords: Vertebral column; Accuracy; Test Size; Machine learning; Cluster K-Nearest Neighbor classifier.

1. Introduction

Machine learning and artificial intelligence growth are improving many research projects in the medical field from more than a decade with an approach based on the integration of pre-existing data to make a diagnosis, take a decision and predict results in a reduced time lapse.

The K-Nearest Neighbor (KNN) is one of the most used and successful types of machine learning [1, 2].The k-NN algorithm is the simplest machine learning algorithm and it is good for small datasets. The classification is based on majority of k-nearest neighbor category, the majority vote among the classification of the k objects and on memory. A model is built on the training data without using any model for fitting. It consists only of storing the training dataset. The model is able to make a prediction for a new data point, unseen data. The algorithm is able to find the closest data points in the training dataset. The KNN algorithm uses neighborhood classification as the prediction value of the new query instance [3]. If a model is able to make accurate predictions on unseen data, it is able to generalize from the training set to the test set.

* Corresponding author: Aissa Boudjella
Bircham International University, Avda Sierra-2 (Urb.Guadamonte), Villanueva de la Cañada - Madrid 28691, Spain.

The vertebral column or spine is a resistant and flexible articular bone chain that is attached to the skull at its upper extremity and to the pelvis at its lower end. In addition to its role of protector of the spinal cord, it allows statics and locomotion.

The spine comprises 33 vertebrae stacked vertically on top of each other, formed by a movable column of 24 vertebrae and a fixed column of fused vertebrae (the sacrum and the coccyx). The vertebrae are connected by facet joints at the back of the spine. These joints allow movement between the bones of the spine. The spine is stabilized by ligaments and are separated by an intervertebral disc located between each vertebra serving as a shock absorber formed by a Fixative fibrous ring and a central pulposus nucleus. The spine is divided into 05 parts: 1)cervical formed by 07 cervical vertebrae; 2)thoracic formed by 12 thoracic vertebrae; 3)lumbar formed by 05 lumbar vertebrae; 4)sacrum formed by 05 fused vertebrae; and 5) coccys formed by 03 to 04 vertebra.

Degenerative pathologies of the vertebral column represent a non-negligible part of the activity in neurosurgery and spine surgery, in particular lumbar pathologies which are a frequent reason for consultation and leading the neurosurgeon to have to make rapid and effective decisions allowing the patient returns back to his activity as quickly as possible. Sometimes the decisions on the pathology are obvious but sometimes it is more difficult to make the right choice in complex cases.

Vijayalakshmi et al. [4] proposed a pattern recognition system to identify the pathologies of the disc hernia and Spondylolisthesis using the kNN machine learning algorithm. The experimental results showed that the system was accurate in achieving a success rate of 88.31%.

Handayani I. investigated the dataset Vertebral Column by applying K-NN algorithm for classification of disk hernia and pondylolisthesis. The author results showed that the accuracy of K-NN classifier was 83% and the average length of time needed for this classification in carrying out the classification process was 0.000212303 seconds [5].

The purpose of our work is to introduce technology and artificial intelligence methods to neurosurgery to reduce the neurosurgeon's thinking time with the capability of automatically decide if a patient has a normal or an abnormal lumbar spine and to hold the decision on the difficult case. Our work focuses on the application of artificial intelligence to pathologies of the spinal column encountered in neurosurgery: disc herniation and spondylolisthesis according to biomechanical attribute. The data have been organized in two different classes. The task consists in classifying patients as belonging to one out of two categories: Normal or Abnormal based on the features. The following convention is used for the class labels. The categories Disk Hernia and Spondylolisthesis were merged into a single category labelled as 'abnormal'. The goal is to build a machine learning model, applied to data that can learn from the measurements of six(06) input variables whose features are known, so that we can predict the class for a new 6 input dataset, consists in classifying patients as belonging to one out of two categories: Normal or Abnormal class.

This paper is organized as follows: Section 2 will present the vertebral dataset and defining different features. In the section 3, the experimental results and discussion will be presented, and finally section 4 will end the paper with conclusion.

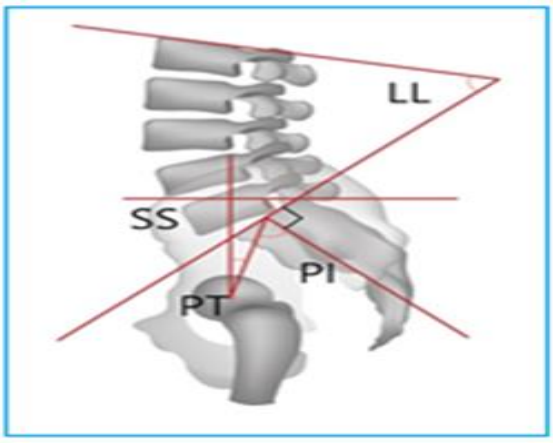


Figure 1 (PI), (PT), (LL),(SS) angles

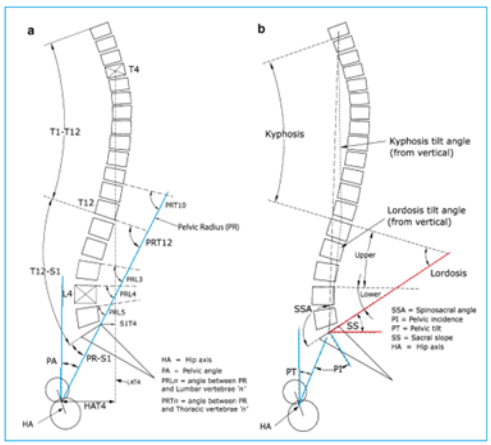


Figure 2 Pelvis Radius (PR), PelvisAngle, (PI), (PT) [7]

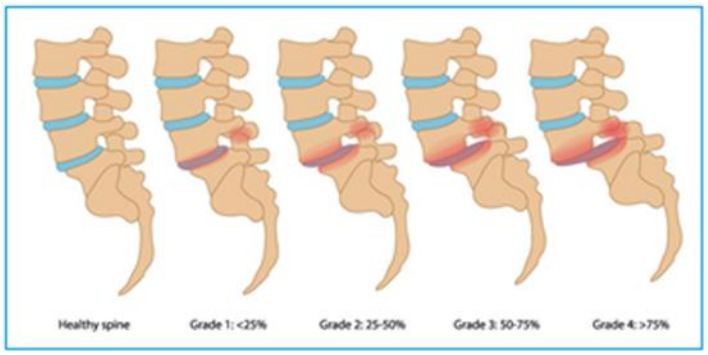


Figure 3 Grade of Spondylolisthesis (GS)

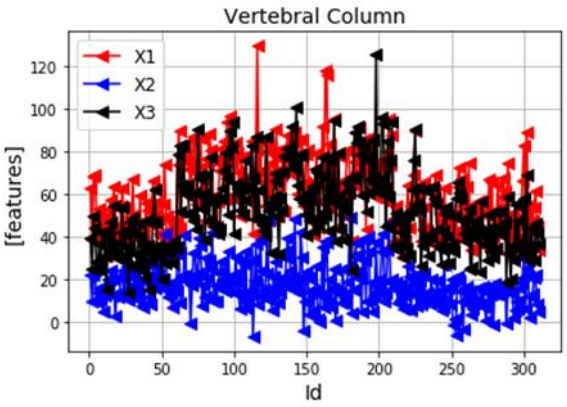
2. Data set

The data we will use for this investigation is a secondary data source, the column vertebral Data Set, [6**]**, which is a classical dataset in machine learning and statistics. This dataset is updated by the authors, replacing one inaccurate input data (degree_spondylolisthesis= 418.5430821, row =id=116) by 41.85430821. The short description of the dataset is reported in reference [6] with the total of 310 instances, eight (08) features and two classes. The abnormal class consists of 210 instances, while the normal class contains only 100 instances are used to carry out the experiment. The eight (08) features, denoted by(X_1, X_2,...X_8) and two responses (or outcome, denoted by y_1 and y_2 , abnormal class and normal class, respectively) to build our model, making this model supervised learning task. Each patient is represented in the data set by six biomechanical attributes derived from the shape and orientation of the pelvis and lumbar spine as indicated in Figures 1~3: 1)X_1=Pelvic Incidence (PI); 2) X_2=Pelvic Tilt (PT); 3) X_3 =Lumbar Lordosis angle, (LL); 4) X_4=Sacral Slope, (SS); 5) X_5=Pelvic Radius (PR), and finally 6) X_6=Grade of Spondylolisthesis and two outcomes, abnormal class y_1 with a status 0, and normal class $\mathbf{y_2}$ with a status 1.

The feature magnitudes versus the patient id number (from 1 to 310) are displayed in Figures 4a and 4b for (X_1 , X_2 and X_3) and (X_4 , X_5 and X_6), respectively. The value of the key target names is an array of strings 0 or 1. The value of feature names is a list of strings, giving the description of each feature. The aim in this investigation is to use the sixth (06) features to predict the pathology defined by each class as target name label.

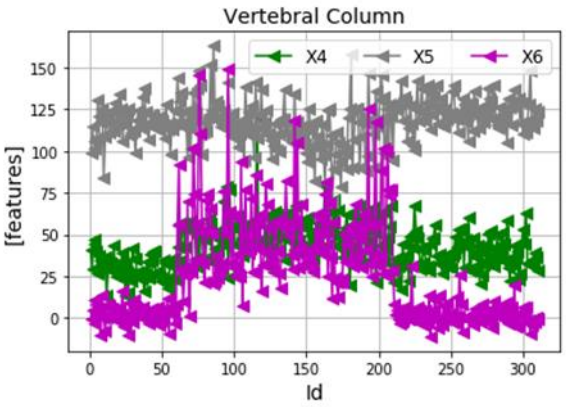
In this problem, we want to predict one of one option of the pathology using two (02) classes, abnormal and normal class. Every 6 attributes in the dataset belongs to one of these two (02) classes. This is an example of two classification problems. The desired output for a single data point is the class status of this dataset. For a particular data point, the class with defined range it belongs to, is called class 1 or class 2.

From this dataset of measurements, we want to build a machine learning model so that we can predict the pathology of a new set of measurements of patient, making this model supervised learning task. Supervised learning algorithms are usually applied to data that contains label information (class target name). The outcome y_1 and y_2 are based on the input data list of 6 strings. Each class is defined by the minimum, maximum and the range (Maximum-Minimum) of the features as indicated in Table 1 and displayed in Figures 5a and 5b. Both classes shows almost the same minimum value for the features (X_1, X_2 , X_3 , X_4 ,and X_6), while the maximum value of the abnormal class 0 for each feature is larger than the normal class1.



(a)

Figure 4a Features (X_1, X_2 and X_3) versus Id

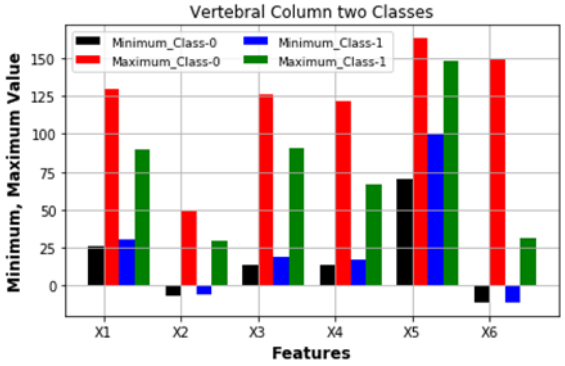


(b)

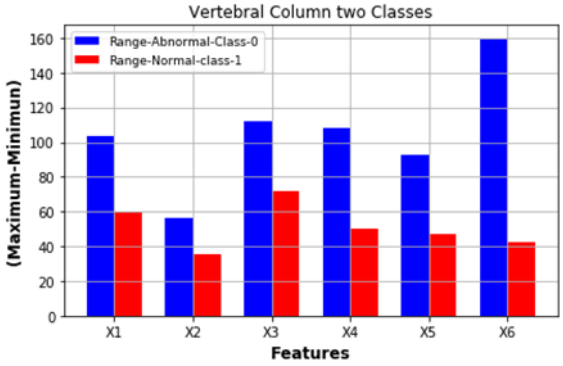
Figure 4b Features (X_4 , X_5 and X_6) versus Id

Table 1 Minimum (Min), maximum (Max) value and the range for each feature

		X_1	X_2	X_3	X_4	X_5	X_6
Class 0	Min	26.1479	−6.554948	14	13.366930	70.0825748	−10.67587
	Max	129.8340	49.431863	125.74238	121.42956	163.071040	41.8543082
	Range	[26.14~129.83]	[−6.55~49.43]	[14~125.74]	[13.36~121.42]	[70.08~163.07]	[−10. 67~ 148.753]
Class 1	Min	30.74193	−5.8459943	19.0710746	17.38697218	100.5011917	−11.0581786
	Max	89.83467	29.8941189	90.5634614	67.19545953	147.8946372	31.17276727
	Range	[30.74~89.83]	[−5.84~29.89]	[19.07~90.56]	[17.38~67.19]	[100.50~147.89]	[−11.05~31.17]



(a)



(b)

Figure 5 (a)Minimum and maximum value, (b) range(Maximum-minimum) for class 0 and class 1. The x-axis, the feature names.

Table 2 Confusion matrix 2 by 2 array

		Actual result	
Prediction Result		Class A	Class B
	Class A	AA	AB
	Class B	BA	BB

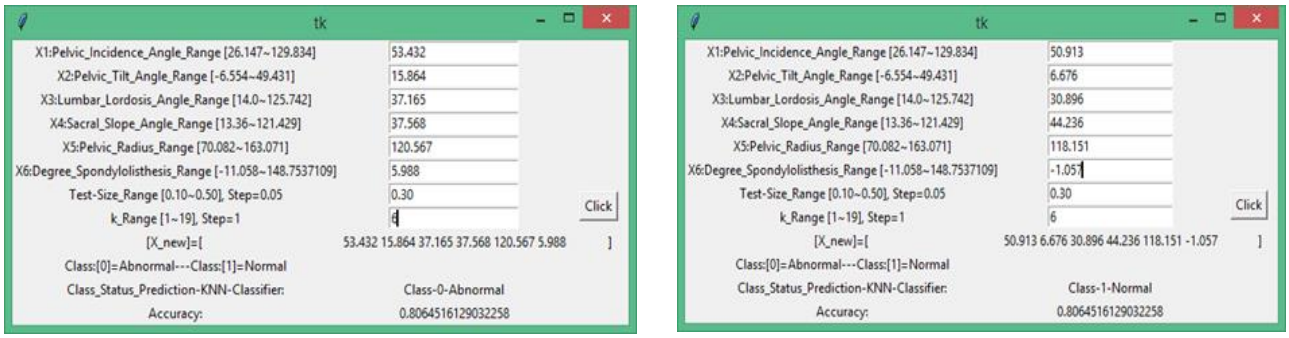
The output of confusion matrix is a two-by-two array, where the rows correspond to the true classes and the columns correspond to the predicted classes. Table 2 illustrates this meaning: by computing accuracy, which can be expressed as

$$Accuracy = \frac{AA + BB}{(AA + BB + BA + AB)}$$

One way of deciding which performance measure is suitable for the task is to consider the confusion matrix. A confusion matrix is a table of contingencies; in the context of statistical modeling, they typically describe the label prediction versus actual labels. It is common to output a confusion matrix (particularly for multiclass problems with more classes) for a trained model as it can yield valuable information about classification failures by failure type and class.

3. Graphic User Interface

To integrate the module of the classifier with the patient database, Graphic User Interface (GUI) was developed by the authors (Figure 6). The user can insert 6 features (float value), the test size (float value) and the k, neighbor parameter (integer) as input for the program classification. The minimum and the maximum value for each feature, the test size (which is limited to 9 values with the step 5%) and k –neighbors with the step 1 are indicated on the GUI. The program will assign the input data to a respective class with accuracy larger 80%, and displays all the retrieved information of the patient after clicking on the click button. If the input data of a new patient is in the range of the data set, the status will appear to be either abnormal or normal as indicated in Figure 6.



Figures 6 GUI shows the class status 1 and 2 and the features information

After achieving good result for testing, all the trained data for the selected dataset was saved to be used for classification process. These data can be called back in the program. For a given input, excluding the training and testing procedure, the classification processing time takes about few seconds. The time refers here to the time to be taken to assign the input data of 6 features (without including the processing time) to determine the class status output.

4. Results and discussion

Figures 7∼18 show the accuracy versus *k*, the nearest neighbor variable under various test and training sizes (with random state=66). The highest accuracy (100%) of the training data was observed when *k=1*, while the test data shows its lowest accuracy for each class size. In the interval of *k* [2∼19]. The accuracy of the training and test set, increases or decreases, in the interval of [0.825∼0.925] and [0.71∼0.84], respectively.

The experimental results show a maximum accuracy, larger than 80 % for the test set in the range of k = 2, (4 ∼19), larger than 90% for the training set when k = (1∼9), and 11. The class label extraction of the test data succeeds in 83% (k=5,6, test size=15%), 82%(k=2,9 test-size=35%), and 84%(k=16, 19, 10%). While the training set shows a higher accuracy 100% for all the training sizes when k=1, larger than 91% when k=2, and larger than 90% when k=3 and 4.

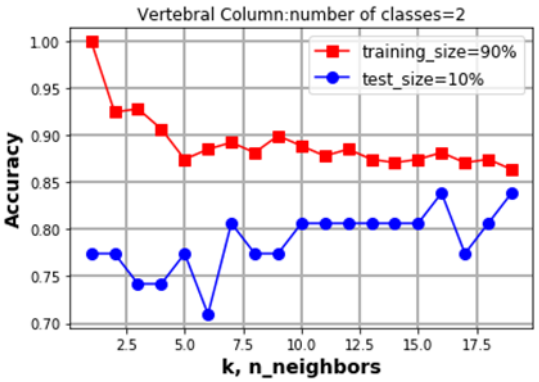


Figure 7 Accuracy versus k-neighbors. Test-size:10%

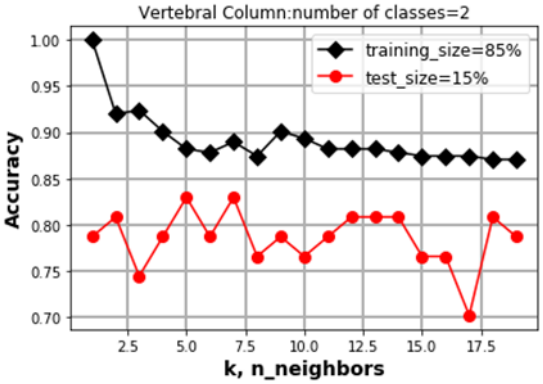


Figure 8 Accuracy versus k-neighbors. Test-size:15%

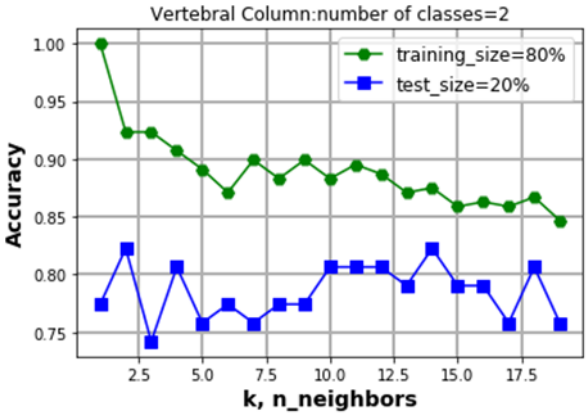


Figure 9 Accuracy versus k-neighbors. Test-size:20%

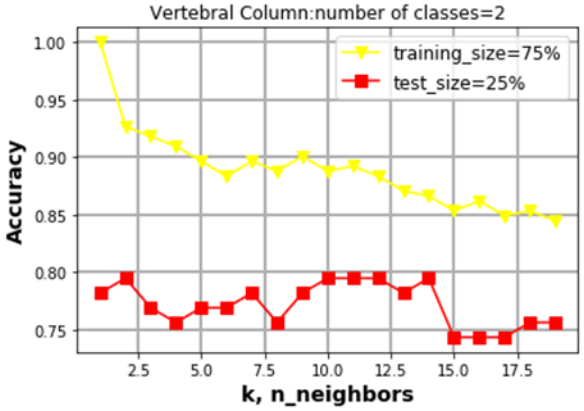


Figure 10 Accuracy versus k,-neighbors.Test-size:25%

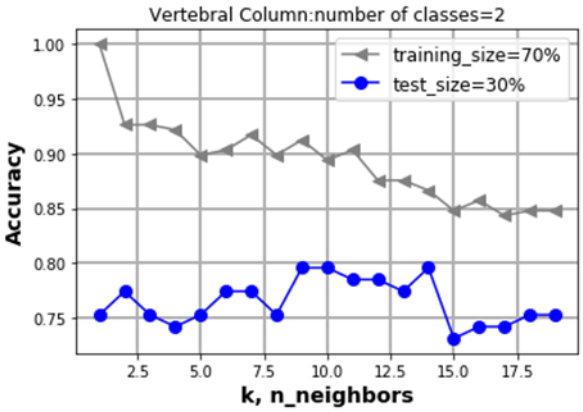


Figure 11 Accuracy versus k-neighbors. Test-size: 30%

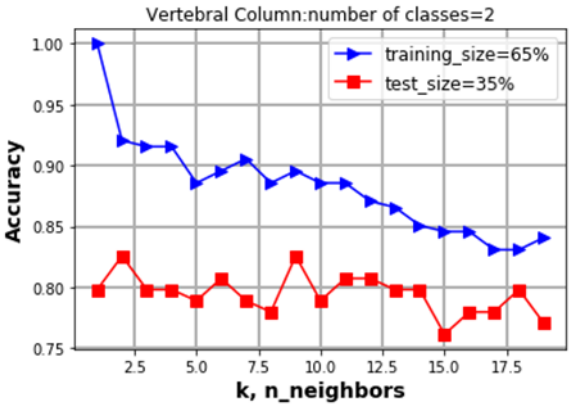


Figure 12 Accuracy versus k-neighbors. Test-size:35%.

In the range of k [2~10], the training accuracy varies from 86% to 92.5%. While the test size shows lower accuracy in comparison with the training one in the range of [70% ~84%]. Considering a single nearest neighbor, (k=1) the prediction on the training set is perfect. But when more neighbors are considered, the model becomes simpler and the training accuracy drops. The test set accuracy for using a single neighbor is lower than when using more neighbors, indicating that using the single nearest neighbor leads to a model that is too complex. On the other hand, when considering more than 10 neighbors, the model is too simple and performance is not worse. The best performance is somewhere in the range of [2~10].

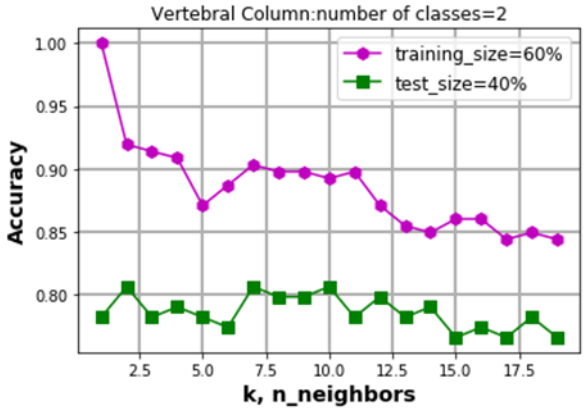


Figure 13 Accuracy versus k-neighbors. Test-size: 40%

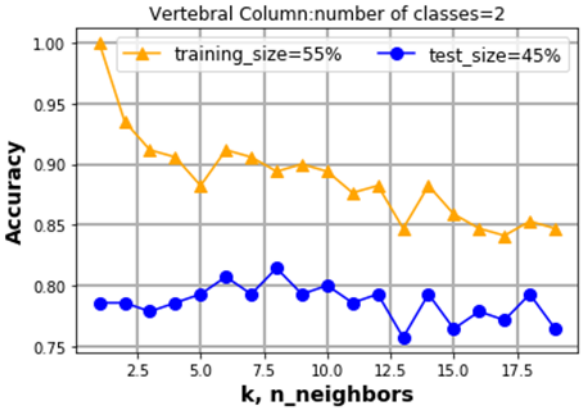


Figure 14 Accuracy versus k-neighbors. Test-size: 45%

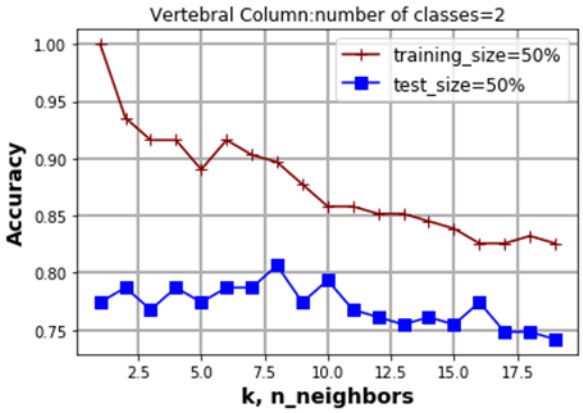


Figure 15 Accuracy versus k-neighbors. Test-size: 50%

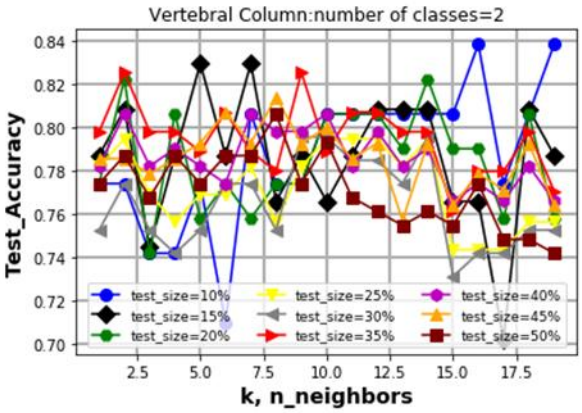


Figure 16 Test Accuracy versus k-neighbors, for various test size(10%~50%)

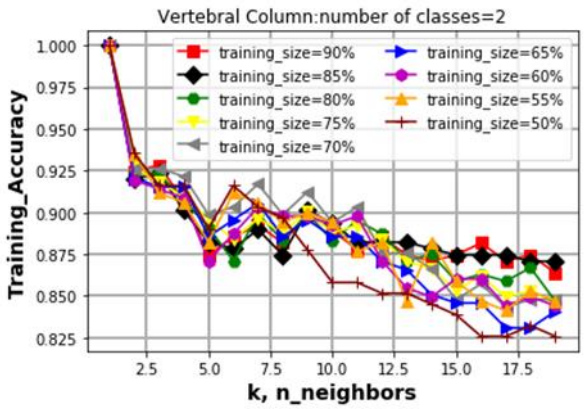


Figure 17 Test Accuracy versus k-neighbors, for various test size(50%~90%)

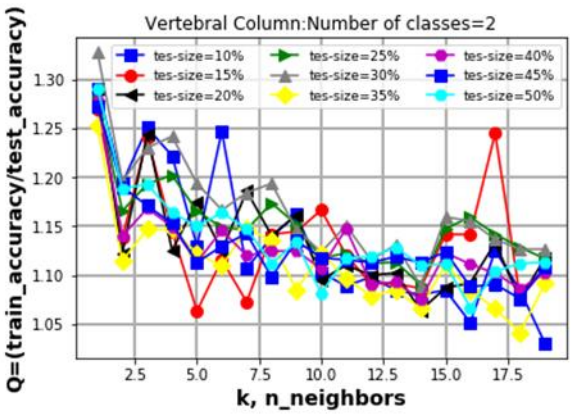


Figure 18 Ratio versus k-neighbors, for various test size (10%~50%)

The ratios of the training size accuracy to the test size accuracy versus *k* under various test sizes [10%~50%], are indicated in Figure 18. The highest value is observed when k=1 in the range of [1.25~1.35]. When k=2, the ratio varies from 1.10 to 1.20, which depends on the test size. By keeping the nearest neighbor *k* constant, the test size or the training size does not affect significantly this ratio. When k>2, Q increases or decreases in the range of [1~1.20].

Table 3 Test set, training set and ratio Q under various ranges of k

k	**1**	**[2-4]**	**[5-10]**	**[11-19]**
Test set-Accuracy	75%~83%	[74%~83%]	[71%~83%]	[70%~84%]
Training set-Accuracy	100%	[90%~95.2%]	[87.5%~92%]	[82.5%~90%]
$Q = \frac{Training\ acuracy}{Test\ Accuracy}$	(1.27~1.37)	[1~1.25]	[1.06~1.25]	[1.03~1.25]

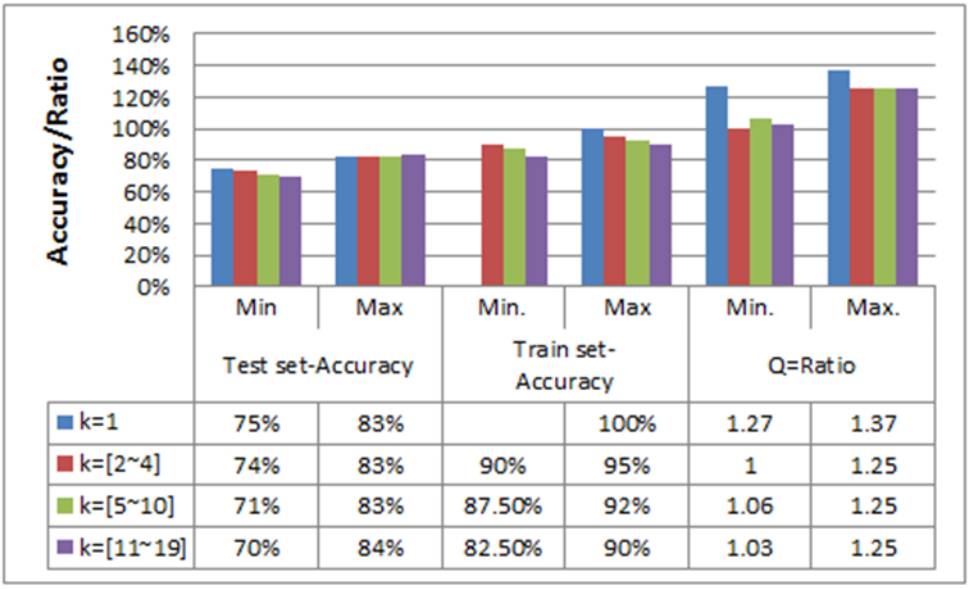


Figure 19 Minmum and maximum of Test, training accuracy and ratio versu k.

Based on the simulation results summarized in Table 3 (represented in Figure 19), this model applied on the dataset of vertebral column is able to make prediction from the training set to the test for the whole range of k under various test sizes. In contrast to the test set, when k=1, the model performs well on the training set with the highest accuracy 100% and the ratio Q (k=1) in the range of [1.27~1.37] which depends on the test size. When k varies from 2 to 4, the training accuracy decreases in the range of [86%~92.5%]. But, the test size accuracy remains approximately the same in the range of [70%~83%]. This is indicated by a lower ratio Q (k=2-4) = [1~1.25] in comparison with Q (k=1) = [1.27~1.37]. Increasing k from 5 to 10, does not affect significantly the test set and the ratio Q (k=5-10) = [1~1.25], which means that the training set accuracy decreases. But the minimum and the maximum magnitude of the training set decreases. For k>10, the range of the accuracy for the training set and the test set are [82.5%~90%] and [70%~84%], respectively. The best performance for this model and this dataset is observed when k=2 and the test size 20% or 35%, with the trainig set and the test set accuracy larger than 90% and 80%, respectively. We see that our model is about 80% accurate, which might still be acceptable. Our results strongly agree with Handayani's results [5] but could not agree less with the Vijayalakshmi's results [4].

5. Conclusion

In this investigation, we build a statistical machine learning model based on supervised learning algorithms, applied to data set that contains two label information classes. GUI has been developed using KNN classifier to improve the efficiency of the diagnostic of pathology on the column vertebral. The working system was tested successfully, which diagnoses and recognizes the pathology on real data. The experimental results show a high accuracy, larger than 90% for the training and larger than 80 % for the test set. The class label extraction of the test data succeeds in 83%, 82% (k=2,9 test-size=35%), and 84% when (k=5,6, test size=15%), (k=2,9 test-size=35%), and(k=16, 19, test size=10%). While the training set shows a higher accuracy for all the training size 100% when k=1, larger than 91% when k=2, and larger than 90% when k=3, 4. This model works well on the training set, but does not perform badly on the test set. But, still, it is good, which might still be acceptable that can learn from the measurements of six (06) input variables whose features are known. The test size combined with the CKNN method can be used to control the accuracy rate. Thus, we

can predict the pathology on vertebral column for new six (06) input dataset with a higher accuracy. This application is faster which can reduce the heavy physician workloads and diagnostic time to make rapid and an effective decision.

Compliance with ethical standards

Acknowledgments

Acknowledgments must be inserted here.

Disclosure of conflict of interest

If two or more authors have contributed in the manuscript, the conflict of interest statement must be inserted here.

References

[1] Duda RO, Hart PE. Pattern Classification and Scene Analysis. John Wiley & Sons. 1973.

[2] Indyk P, Motwani R. Approximate nearest neighbors: towards removing the curse of dimensionality. In STOC. Proceedings of the thirtieth annual ACM symposium on Theory of computing.1998: 604–613.

[3] Teknomo K. K-Nearest Neighbors Tutorial. 2006. . Available from https://people.revoledu.com/kardi/tutorial/KNN/

[4] Vijayalakshmi G V, Mohan Kumar M. Diagnosis of Vertebral Column Pathologies using kNN Classifier. International Journal of Engineering Research & Technology (IJERT). ISSN: 2019; 7(8):2278-0181.

[5] Handayani I. Application of K-Nearest Neighbor Algorithm on Classification of Disk Hernia and Spondylolisthesis in Vertebral Column. Indonesian Journal of Information Systems (IJIS). 2019; 2(1):57-66

[6] Rocha Neto A. R. and Barreto, G. A. . 'On the Application of Ensembles of Classifiers to the Diagnosis of Pathologies of the Vertebral Column: A Comparative Analysis', IEEE Latin America Transactions. 2009; 7(4):487-496. Available from https://archive.ics.uci.edu/ml/datasets/Vertebral+Column

[7] Sergides IG, McCombe Peter F, White G , Sabarul M., William R. Sears. Lumbo-pelvic lordosis and the pelvic radius technique in the assessment of spinal sagittal balance: strengths and caveats. European Spine Journal. 2011; 20 (Suppl 5): S591–S601.