

# Comparative Analysis of KNN Algorithm using Various Normalization Techniques

**Amit Pandey**

Bharati Vidyapeeth's College of Engineering, Information Technology, New Delhi, 110063, India  
E-mail: [pandey17amit@gmail.com](mailto:pandey17amit@gmail.com)

**Achin Jain**

Bharati Vidyapeeth's College of Engineering, Information Technology, New Delhi, 110063, India  
E-mail: [achin.jain@bharatividyaeeeth.edu](mailto:achin.jain@bharatividyaeeeth.edu)

Received: 19 June 2017; Accepted: 10 August 2017; Published: 08 November 2017

**Abstract**—Classification is the technique of identifying and assigning individual quantities to a group or a set. In pattern recognition, K-Nearest Neighbors algorithm is a non-parametric method for classification and regression. The K-Nearest Neighbor (kNN) technique has been widely used in data mining and machine learning because it is simple yet very useful with distinguished performance. Classification is used to predict the labels of test data points after training sample data. Over the past few decades, researchers have proposed many classification methods, but still, KNN (K-Nearest Neighbor) is one of the most popular methods to classify the data set. The input consists of k closest examples in each space, the neighbors are picked up from a set of objects or objects having same properties or value, this can be considered as a training dataset. In this paper, we have used two normalization techniques to classify the IRIS Dataset and measure the accuracy of classification using Cross-Validation method using R-Programming. The two approaches considered in this paper are - Data with Z-Score Normalization and Data with Min-Max Normalization.

**Index Terms**—KNN, Classification, Normalization, Z-Score Normalization, Min-Max Normalization, Cross Validation Method.

## I. INTRODUCTION

In present day scenario, machine learning and artificial intelligence are replacing all the conventional computational techniques and programming languages, most importantly machine learning gives computers the ability to learn without being explicitly programmed. K-Nearest Neighbors algorithm is a non-parametric method for classification and regression. The input consists of k closest examples in each space.

For given N training vectors, the k-nearest neighbor algorithm identifies the k number of nearest neighbors. The closest of neighbors are considered and hence it becomes the class of test element.

In this paper, the comparison study includes the results from normalized and non-normalized data.

Table 1. Data-Set Summary

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
Min 4.300	Min: 2.000	Min: 1.000	Min: 0.100	setosa: 50
1stQu: 5.100	1stQu: 2.800	1stQu: 1.600	1stQu: 0.300	versicolor :50
Median:5.800	Median:3.000	Median:4.350	Median:1.300	virginica: 50

The above data-set shown in table I. demonstrates summary about the uneven maximum and minimum values of each of the 4 parameters. The Sepal length has minimum and a maximum value of 4.300 and 7.900 respectively, Sepal width has minimum and a maximum value of 2.000 and 4.400 respectively, Petal length has minimum and a maximum value of 1.000 and 6.900 respectively and Petal width has minimum and a maximum value of 0.100 and 2.500 respectively. Petal length and petal width have wide and narrow ranges respectively. The features with largest values will have an undue effect on the prediction.

According to the definition kNN uses distance function, so it is a requirement for every feature to be scaled in a similar fashion. To normalize the data in appropriate scale, there are various Normalization techniques proposed by the researchers over the years.

KNN text classification algorithm has three limitations: 1. Lack of difference in weight between samples 2. Complexity of calculation due to the usage of all the training samples for classification, and 3. The performance mainly depends on the training set.

This paper is concerned with the comparative study or analysis of K-Nearest neighbor algorithm under different normalization techniques and different values of K. For the comparative analysis, we have used "IRIS" Dataset.

The Paper is structured as follows: Section I. of the Paper provides a basic introduction about the KNN Algorithm and has also discussed why Normalization is required. Section II. gives a brief description of the related work carried over by various researchers on KNN over the years. Section III. discusses in detail about the KNN Algorithm with a flowchart. We have presented our experimental work results in Section IV. for two Normalization techniques Z-Score and Min-Max. In subsection C. comparative analysis of both the normalization techniques is presented. Finally, section V. concludes the paper.

## II. RELATED WORK

H. V. Jagadish Beng Chin Ooi, Kian-Lee Tan, Cui Yu, Rui Zhang [1], proposed an efficient B+-tree based indexing method called iDistance, for K-nearest neighbor (KNN) search in a high-dimensional metric space. Giuseppe Amato, Fabrizio Falchi, and Claudio Gennaro[2] used kNN classification and landmark recognition techniques to address the problem of monument recognition in images. Ioannis Stamoulias and

Elias S. Manolakos [3] fashioned a mixture of k-nearest-neighbor collateral architectures for FPGAs in the shape of parametrizable soft IP cores. Exhibiting that they can be utilized to solve sizeable classification problems with thousands of training vectors, or thousands of vector attributes using a single FPGA, and accomplish very high output. Xiaohua Liu, Furu Wei Shaodian Zhang and Ming Zhou [4] propose a creative method consisting of three chief elements: (1) normalization of tweets, (2) conjunction of a K-Nearest Neighbours (KNN) allocator with a linear Conditional Random Fields (CRF) model, (3) semi-supervised learning framework. The tweet normalization preprocessing rectifies common ill-formed terms using a global linear model. The KNN- based classifier administers pre-labelling to collect global coarse information across tweets while the CRF model conducts consecutive labeling to capture fine-grained data ciphered in a tweet. The semi-supervised learning plus the gazetteers improve the lack of training data. All-encompassing experiments show the assets of their method over the baselines as well as the effectiveness of normalization, KNN, and semi-supervised learning.

Table 2. Application Areas of Knn.

Application Area	Description	Related Work
Text Mining	Text categorization is the method of identifying the class to which a text document belongs.	[11] Ke-Wei Huang and Zhuolun Li- "A multi-label text classification algorithm for labeling risk factors in SEC form 10-K" [12] Bijalwan, Vishwanath, et al. "KNN based machine learning approach for text and document mining." <i>International Journal of Database Theory and Application</i> 7.1 (2014): 61-70.
Agriculture	In practice, KNN is employed less than other data mining techniques in fields related to agriculture. Some applications include simulation of precipitation and other weather parameters. Another application includes using of satellite imagery for evaluation of forest inventories and estimation of various other forest parameters.	[13]Radnaabazar Chinchuluun, Won Suk Lee, Jevin Bhorania and Panos M. Pardalos "Clustering and Classification Algorithms in Food and Agricultural Applications: A Survey" [14] Reese, Heather, et al. "Applications using estimates of forest parameters derived from satellite and forest inventory data." <i>Computers and Electronics in Agriculture</i> 37.1 (2002): 37-55.
Finance	KNN as a process of scraping out useful patterns and correlations has its own domain in financial modeling. Stock market forecasting including planning investment strategies, uncovering market trends, what stocks to purchase etc. Are some of the crucial financial tasks of KNN. These further include-Credit rating, Money laundering analysis etc.	[15] Hui-Ling Chen, Bo Yang, Gang Wang, Jie Liu, Xin Xu, Su-Jing Wang and Da-You Liu "bankruptcy prediction model based on an adaptive fuzzy k-nearest neighbor method" [16] Tam, Kar Yan, and Melody Y. Kiang. "Managerial applications of neural networks: the case of bank failure predictions." <i>Management science</i> 38.7 (1992): 926-947.
Medicine	KNN finds extensive usage in pharmaceutical industry especially in the detection of oncogenic (cancer cells) growth. This can be done by sorting the given cell data according to the Radius, Symmetry and other parameters. Other applications include the detection of intrusions in computer systems, the prediction of solvent accessibility in protein molecules and analyzing microarray gene expression data.	[17] Bendi Venkata Ramana , M.Surendra Prasad Babu , N. B. Venkateswarlu "A Critical Study of Selected Classification Algorithms for Liver Disease Diagnosis" [18] He, Hongxing, Warwick Graco, and Xin Yao. "Application of genetic algorithm and k-nearest neighbour method in medical fraud detection." <i>Asia-Pacific Conference on Simulated Evolution and Learning</i> . Springer Berlin Heidelberg, 1998. [19] Goshvarpour, Ateke, and Atefeh Goshvarpour. "Radial Basis Function and K-Nearest Neighbor Classifiers for Studying Heart Rate Signals during Meditation." <i>International Journal of Modern Education and Computer Science</i> 4.4 (2012): 43. [20] Ahirwar, Anamika. "Study of techniques used for medical image segmentation and computation of statistical test for region classification of brain MRI." <i>International Journal of Information Technology and Computer Science (IJITCS)</i> 5.5 (2013): 44.

Hadar Averbuch-Elor Daniel Cohen-Or [5] assume a nearly instantaneous event, such as an interesting moment in a performance captured by the digital cameras and smartphones of the surrounding crowd. The ordering method extracts the K-nearest neighbors (KNN) of each image from a rough all-pairs dissimilarity estimate. The KNN dissimilarities are refined to form a sparse weighted Laplacian, and a spectral analysis then yields a ring angle for each image. Yu-Sheng Lai and Chung-Hsien Wu [6] proposed a method founded on unknown words for meaningful term extraction and selective term selection in text categorization. For significant term extraction, a phrase like unit (PLU)-based probability ratio is proposed to estimate the probability that a word succession is an unknown word. Jinhui Tang, Richang Hong, Shuicheng Yan, Tat-Seng Chua, Guo-Jun Qi, Ramesh Jain [7] maneuver the problem of elucidating a large-scale image corpus by label propagation over noisily tagged web images. To annotate the images more accurately, they proposed a novel kNN-sparse graph-based semi-supervised learning approach for harnessing the labeled and unlabeled data simultaneously. Li Baoli, Lu Qin, Yu Shiwen [8] proposed an improved kNN strategy, in which different numbers of nearest neighbors for different categories are used instead of a fixed number across all categories.

### III. K-NEAREST NEIGHBORS ALGORITHM

K-Nearest Neighbors algorithm falls under the category of lazy learning. The neighbors are picked up from a set of objects or objects having same properties or value, this can be considered as a training dataset. The algorithm uses the Euclidean distance, which is a straight path connecting two points. Before applying KNN algorithm on a dataset, the dataset must be prepared, that means, the dataset's parameters must be scaled down to a normalized scale.

Euclidean distance between points A and B is the length of the line segment connecting them.

The formula for Euclidean distance is given in equation 1 below:

$$d(A, B) = \sqrt{(x_1 - x_2)^2 + (y_2 - y_1)^2} \quad (1)$$

The flowchart shown in figure 1. Shows the stepwise demonstration KNN Algorithm working.

Start with choosing a data set, once done, shuffle the dataset just in case it is in serial order, which would render the whole algorithm pointless.

After getting the desired shuffled dataset, apply normalization to scale the minimum and maximum values of the parameters. Then take a portion of the data as a training model and apply our algorithm to it. Finally, evaluate and analyze the accuracy of the prediction of the algorithm.

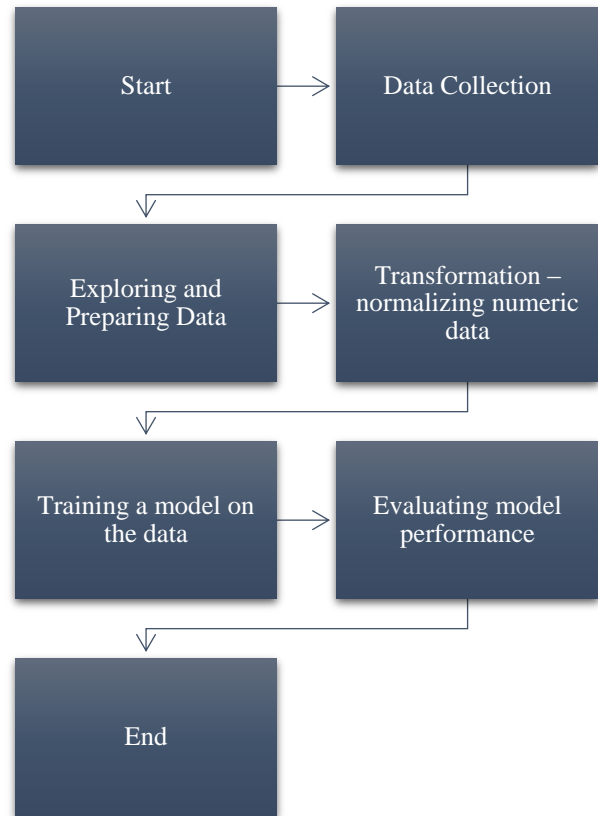


Fig.1. Flowchart of kNN Algorithm

### IV. EXPERIMENTAL WORK

This article primarily focuses on the comparative study of various values of K in K-Nearest Neighbors algorithm applied on a given dataset using two normalization techniques, namely- Z-score normalization and Min-Max normalization. The accuracy of the predicted results of each of the two techniques has been compared. Also, we have computed the average prediction efficiency of K-nearest neighbor algorithm using the two normalization techniques and concluded the one technique with the highest efficiency.

#### A. Dataset Summary

For the experimental work, we have used IRIS Dataset comprising of 3 classes with 50 instances in each class. In the dataset, each class refers to a different type of Iris plant. One class is linearly separable from the other 2, the latter are not linearly separable from each other. Following are the attributes in the IRIS Dataset:

1. sepal length in cm
2. sepal width in cm
3. petal length in cm
4. petal width in cm
5. class:
  - a. Iris Setosa
  - b. Iris Versicolour
  - c. Iris Virginica

Class attribute of the iris plant is used for the prediction measure. Table III. Shows sample of IRIS Dataset.

Table 3. Sample of ‘Iris’ Dataset

	Sepal length	Sepal width	Petal length	Petal width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa

Summary of the IRIS Dataset which shows the various statistical parameters are shown in Table IV.

Table 4. Summary of ‘Iris’ Dataset

Sepal Length	Sepal Width	Petal Length	Petal Width	Species
Min: 4.300	Min: 2.000	Min: 1.000	Min: 0.100	setosa: 50
1stQu: 5.100	1stQu: 2.800	1stQu: 1.600	1stQu: 0.300	versicolor: 50
Median: 5.800	Median: 3.000	Median: 4.350	Median: 1.300	virginica: 50
Mean: 5.843	Mean: 3.057	Mean: 3.758	Mean: 1.199	
3rdQu: 6.400	3rdQu: 3.300	3rdQu: 5.100	3rdQu: 1.800	
Max: 7.900	Max: 4.400	Max: 6.900	Max: 2.500	

The above data given contains four parameters of 50 species of each Setosa, Versicolor, Virginica. The above gives an account of minimum value, 2nd, 3rd quartile and the maximum value of each parameter.

**B. Normalization**

Normalization is the process of adjusting values measured on a different scale to a common scale. Normalization allows comparison of corresponding values of different data-sets. Without normalization, our data would be unscaled and hence highly intricate to calculate and compare with other parameters. There are many Normalization techniques – Feature scaling, Standardized moment, Coefficient of variation, Studentized residual, Student’s t-statistic, Standard Score. In this paper, we have focused on two normalization techniques i.e., Z-Score and Min-Max Normalization technique.

**1. Standard Score (Z-score)**

In [9] the Z-Score method or standard score method is presented, which normalizes each score to its number of standard deviations that it is distant from the mean score. Equation 2. shows basic Formula of Z-Score Normalization.

$$\frac{x-\mu}{\sigma} = \frac{x-\text{mean}}{\text{standarddeviation}} \tag{2}$$

This formula rescales each of a feature’s value in terms of how many standard deviations they fall above or below the mean value. The resulting value is called a z-score. The z-scores fall in an unbounded range of negative and positive numbers. Unlike the normalized values, they have no predefined minimum and maximum.

Following is the Z-Score Normalization Code is written in R-Programming to apply on the dataset.

```
normalize <-function(x){return((x-mean(x))/(sd(x)))}
iris_n <- as.data.frame(lapply(iris[,c(1,2,3,4)],
normalize))
iris_train <-iris_n[1:129,]
iris_test <-iris_n[130:150,]
iris_train_target <-iris[1:129,5]
```

Table V. shows IRIS Dataset before applying Z-Score Normalization technique.

Table 5. Sample of Dataset before Z-Score Normalization

S.No	Sepal Length	Sepal Width	Petal Length	Petal Width
1	5.1	3.5	1.4	0.2
2	4.9	3	1.4	0.2
3	4.7	3.2	1.3	0.2
4	4.6	3.1	1.5	0.2
5	5	3.6	1.4	0.2
6	5.4	3.9	1.7	0.4

Table VI. shows the Dataset after application of Z-Score Normalization.

Table 6. Post Z-Score Normalization Result

S. No.	Sepal length	Sepal width	Petal length	Petal width
1	-1.25996	0.09788	-1.22246	-1.311051
2	1.63836	-0.13154	1.15675	0.525644
3	0.18919	-0.13154	0.59027	0.788031
4	-0.89766	0.78616	-1.27910	-1.311052
5	2.24216	-1.04924	1.77987	1.443993
6	-0.53537	1.93331	-1.39240	-1.048667

Table 7. Accuracy Of The Predication

K	Z-score normalization
1	85.71%
13	85.71%
50	100.00%
100	42.85%

The above is the table showing the values of K vs the accuracies of the prediction using Z-score normalization. We get the maximum accuracy at k=50 and minimum accuracy at k=100.

The graph below is a pictorial representation of the prediction accuracy on applying Z-score normalization.

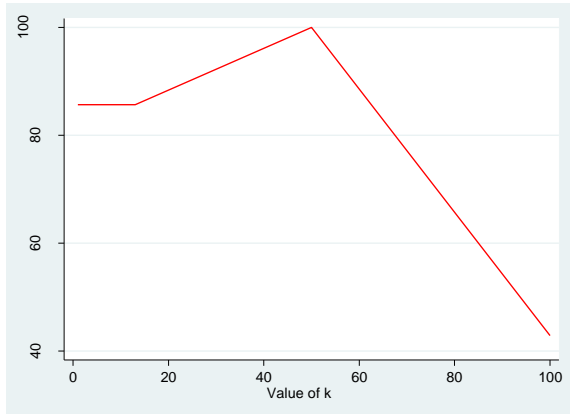


Fig.2. Z-score Normalization Accuracy vs. K

2. Min-Max Normalization (Feature Scaling)

The Min-Max method [10] transforms a feature such that all of its values fall in the domain [0, 1].

Equation 3. shows basic Formula of Min-Max Normalization

$$X_{new} = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (3)$$

Normalized feature values can be interpreted as indicating how far, from 0 percent to 100 percent, the original value fell along the range between the original minimum and maximum.

Table VIII. shows the dataset before application of Min-Max Normalization.

Table 8. Original Dataset (Without Normalization)

	Sepal length	Sepal width	Petal length	Petal width
1	5.1	3.5	1.4	0.2
2	4.9	3	1.4	0.2
3	4.7	3.2	1.3	0.2
4	4.6	3.1	1.5	0.2
5	5	3.6	1.4	0.2
6	5.4	3.9	1.7	0.4

Following is the snippet of code executed to get the desired result.

```
normalize <-function(x){return((x-
min(x))/(max(x)-min(x)))}
iris_n <- as.data.frame(lapply(iris[,c(1,2,3,4)],
normalize))
iris_train <-iris_n[1:129,]
iris_test <-iris_n[130:150,]
iris_train_target <-iris[1:129,5]
```

Table IX. shows the dataset after application of Min-Max Normalization to the original dataset.

Table 9. Post Min-Max Normalization Result

	Sepal length	Sepal width	Petal length	Petal width
1	0.1388889	0.4583333	0.1016949	0.0416667
2	0.8055556	0.4166667	0.8135593	0.625
3	0.4722222	0.4166667	0.6440678	0.7083333
4	0.2222222	0.5833333	0.0847458	0.0416667
5	0.9444444	0.25	1	0.9166667
6	0.3055556	0.7916667	0.0508475	0.125

Table 10. Accuracy of the Predication

K	min/max normalization
1	100.00%
13	95.23%
50	90.47%
100	66.67%

Again, like shown in Z-score normalization, the above is the table showing the values of K vs the accuracies of the prediction using Min-Max normalization. We get the maximum accuracy at k=1 and minimum accuracy at k=100.

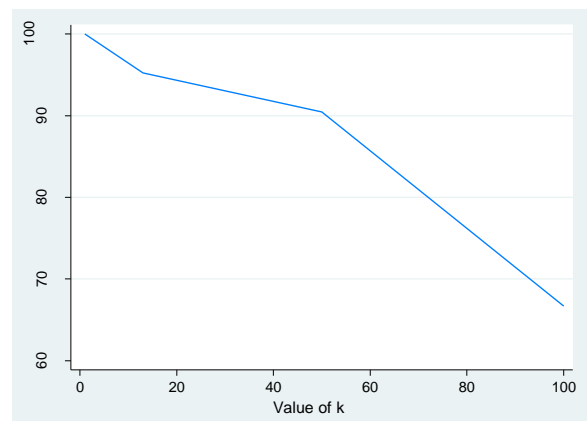


Fig.3. Min-Max Normalization Accuracy vs. K

The graph above is a pictorial representation of the prediction accuracy on applying Min-Max normalization.

C. Comparison between Different Normalization Techniques with Different Values of k

Table 11. Accuracy of the Predication (Min-Max Vs Z-Score)

K	Min-max normalization	Z-score normalization
1	100.00%	85.71%
13	95.23%	85.71%
50	90.47%	100.00%
100	66.67%	42.85%

In this paper, we have analyzed that for different values of K, different normalization techniques give different accuracies. The KNN algorithm has predicted 3



species into their respective categories, under different values of K and normalization, the accuracy of predictions have varied. Conventionally, the best K value is taken as the square root of the total number of observations. As shown in the table above, for K=1, the min/max normalization technique gives us 100% accuracy whereas Z-score normalization gives 85.71%, for K=13, the accuracy is 95.23% for Min-Max normalization and again 85.71% for Z-score, at K=50, the Min-Max gives 90.47% accuracy whereas Z-score normalization gives a good 100% accuracy, at K=100, Min-Max gives 66.67% accuracy whereas Z-score gives 42.85% accuracy.

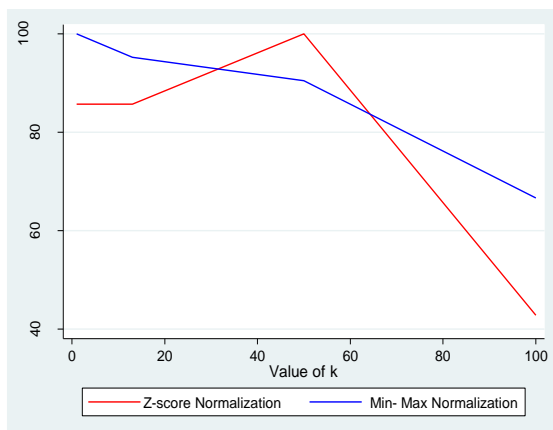


Fig.4. Comparison of Z-Score and Min-Max Normalization Technique

## V. CONCLUSION

In this paper, we have compared two normalization techniques, namely- Min-Max normalization and Z-score normalization for different values of K.

The average accuracy came out to be 88.0925% for Min-Max normalization 78.5675% for Z-score normalization.

## REFERENCES

- [1] Jagadish, Hosagrahar V., et al. "iDistance: An adaptive B+-tree based indexing method for nearest neighbor search." *ACM Transactions on Database Systems (TODS)* 30.2 (2005): 364-397.
- [2] Amato, Giuseppe, Fabrizio Falchi, and Claudio Gennaro. "Fast image classification for monument recognition." *Journal on Computing and Cultural Heritage (JOCCH)* 8.4 (2015): 18.
- [3] Stamoulias, Ioannis, and Elias S. Manolakos. "Parallel architectures for the kNN classifier--design of soft IP cores and FPGA implementations." *ACM Transactions on Embedded Computing Systems (TECS)* 13.2 (2013): 22.
- [4] Liu, Xiaohua, et al. "Named entity recognition for tweets." *ACM Transactions on Intelligent Systems and Technology (TIST)* 4.1 (2013): 3.
- [5] Averbuch-Elor, Hadar, and Daniel Cohen-Or. "RingIt: Ring-Ordering Casual Photos of a Temporal Event." *ACM Trans. Graph.* 34.3 (2015): 33.
- [6] Lai, Yu-Sheng, and Chung-Hsien Wu. "Meaningful term extraction and discriminative term selection in text categorization via unknown-word methodology." *ACM Transactions on Asian Language Information Processing (TALIP)* 1.1 (2002): 34-64.
- [7] Tang, Jinhui, et al. "Image annotation by k nn-sparse graph-based label propagation over noisily tagged web images." *ACM Transactions on Intelligent Systems and Technology (TIST)* 2.2 (2011): 14.
- [8] Baoli, Li, Lu Qin, and Yu Shiwen. "An adaptive k-nearest neighbor text categorization strategy." *ACM Transactions on Asian Language Information Processing (TALIP)* 3.4 (2004): 215-226.
- [9] M. Montague and J. A. Aslam. Relevance score normalization for metasearch. In *Proceedings of the ACM CIKM*, pages 427-433. ACM, 2001.
- [10] Forman, George, Martin Scholz, and Shyamsundar Rajaram. "Feature shaping for linear SVM classifiers." *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining.* ACM, 2009.
- [11] Huang, Ke-Wei, and Zhuolun Li. "A multilabel text classification algorithm for labeling risk factors in SEC form 10-K." *ACM Transactions on Management Information Systems (TMIS)* 2.3 (2011): 18.
- [12] Bijalwan, Vishwanath, et al. "KNN based machine learning approach for text and document mining." *International Journal of Database Theory and Application* 7.1 (2014): 61-70.
- [13] Chinchuluun, Radnaabazar, et al. "Clustering and classification algorithms in food and agricultural applications: a survey." *Advances in modeling agricultural systems.* Springer US, 2009. 433-454.
- [14] Reese, Heather, et al. "Applications using estimates of forest parameters derived from satellite and forest inventory data." *Computers and Electronics in Agriculture* 37.1 (2002): 37-55.s
- [15] Chen, Hui-Ling, et al. "A novel bankruptcy prediction model based on an adaptive fuzzy k-nearest neighbor method." *Knowledge-Based Systems* 24.8 (2011): 1348-1359.
- [16] Tam, Kar Yan, and Melody Y. Kiang. "Managerial applications of neural networks: the case of bank failure predictions." *Management science* 38.7 (1992): 926-947.
- [17] Ramana, Bendi Venkata, M. Surendra Prasad Babu, and N. B. Venkateswarlu. "A critical study of selected classification algorithms for liver disease diagnosis." *International Journal of Database Management Systems* 3.2 (2011): 101-114.
- [18] He, Hongxing, Warwick Graco, and Xin Yao. "Application of genetic algorithm and k-nearest neighbour method in medical fraud detection." *Asia-Pacific Conference on Simulated Evolution and Learning.* Springer Berlin Heidelberg, 1998.
- [19] Goshvarpour, Ateke, and Atefeh Goshvarpour. "Radial Basis Function and K-Nearest Neighbor Classifiers for Studying Heart Rate Signals during Meditation." *International Journal of Modern Education and Computer Science* 4.4 (2012): 43.
- [20] Ahirwar, Anamika. "Study of techniques used for medical image segmentation and computation of statistical test for region classification of brain MRI." *International Journal of Information Technology and Computer Science (IJITCS)* 5.5 (2013): 44.

**Authors' Profiles**

**Amit Pandey** is a student at Bharati Vidyapeeth's College of Engineering, New Delhi (Affiliated to Guru Gobind Singh Indraprastha University, Delhi) pursuing the bachelor of technology in Information Technology. His area of interest includes Machine Learning.



**Achin Jain**, working as Assistant Professor in Bharati Vidyapeeth's College of Engineering, New Delhi (Affiliated to Guru Gobind Singh Indraprastha University, Delhi) in the Information Technology Department. He received his M.Tech (Computer Science and Technology) & B.Tech (Information Technology) from the Guru Gobind Singh Indraprastha University, Delhi. He has rich experience teaching B.Tech students and has published more than 6 Research Papers in International Journals and Conferences. His area of interest includes Web Usage Mining, Web Attacks.

**How to cite this paper:** Amit Pandey, Achin Jain, "Comparative Analysis of KNN Algorithm using Various Normalization Techniques", International Journal of Computer Network and Information Security(IJCNIS), Vol.9, No.11, pp.36-42, 2017.DOI: 10.5815/ijcnis.2017.11.04