IJCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-7, Issue-6, June 2019 E-ISSN: 2347-2693

A Comparative Analysis of Different Machine Learning Classification Algorithms for Predicting Chronic Kidney Disease

Bhawna Sharma[1*], Sheetal Gandotra[2], Utkarsh Sharma[3], Rahul Thakur[4] , Alankar Mahajan[5]

[1,2,3,4,5] Dept. of Computer Engineering, Government College of Engineering and Technology, Jammu and 181122, India

[*]*Corresponding Author: bhawnash@gmail.com,*

DOI: https://doi.org/10.26438/ijcse/v7i6.813 | Available online at: www.ijcseonline.org

Accepted: 12/Jun/2019, Published: 30/Jun/2019

***Abstract*—** Chronic kidney disease (CKD) is a condition characterized by a gradual loss of kidney function over time. It includes risk of cardiovascular disease and end-stage renal disease. In this paper, we use Machine Learning approach for predicting CKD. In this paper, we present a comparative analysis of seven different machine learning algorithms. This study starts with twenty-four parameters in addition to the class attribute and twenty five percent of the data set is used to test the predictions. Algorithms are trained using fivefold cross-validation and performance of the system is assessed using classification accuracy, confusion matrix, specificity and sensitivity.

***Keywords*—** CKD, Machine Learning, Logistic Regression, Support Vector Machine, Random Forest.

I. INTRODUCTION

The interest and inescapability of Machine Learning (ML) is developing. Existing techniques are being enhanced. These accomplishments have prompted the appropriation of machine learning in a few areas, for example, PC vision, therapeutic examination, gaming, web-based life promoting, detecting diseases such as Parkinson's Disease, Intrusion Detection, etc. [1] [2]. In a few situations, machine learning methods provide better results over conventional run-based calculations and even human administrators.

Chronic kidney disease (CKD) is a permanent reduction in kidney function that can progress to end stage renal disease (ESRD), requiring either ongoing dialysis or a kidney transplant to maintain life. CKD also affects how many medications are eliminated from the body [3]. In routine practice, a laboratory serum creatinine value is used to estimate kidney function by incorporating it into a formula to estimate the glomerular filtration rate and establish whether a patient has CKD. It is becoming a major threat in the developing and undeveloped countries. Its main cause for occurrence is diseases like diabetes, high blood-pressure. Other risking circumstances causing chronic kidney disease include heart disease, obesity, and a family history of chronic kidney disease. Its medications, which are dialysis or kidney transplant are very costly and so we need an early detection. In the United States (US), about 117,000 patients developed end-stage renal disease (ESRD) requiring dialysis, while more than 663,000 prevalent patients were on dialysis in 2013. 5.6% of the total medical budget was spent for ERDS in 2012 which is about $28 billion. In India, CKD is widespread among 800 per million populations and ESRD is 150–200 per million populations.

We consider seven machine learning classifiers, namely Logistic Regression, Support Vector Machine, K-nearest Neighbor, Naïve Bayes, Stochastic Gradient Descent Classifier, Decision Trees and Random Forest for predicting CKD. Finally, a set of standard performance metrics is used for estimating the performance of each machine learning and artificial intelligence classifier. The metrics we used included confusion matrix, classification accuracy, specificity and sensitivity.

The rest of the paper is organized as follows. Section-II explains the process workflow and the techniques used for preprocessing the dataset. Section-III gives an overview of the seven machine learning algorithms used in the research. Section-IV gives an overview of the parameters used to evaluate the performance of algorithms. Section-V describes the results obtained after training and testing the algorithms based on the parameters defined in Section-IV. Section-VI draws conclusions.

II. METHODOLOGY

A. Proposed Method

The proposed method compares classification performance of seven different machine learning algorithms namely

Logistic Regression, Support Vector Machine, K-Nearest Neighbor, Naïve Bayes, Stochastic Gradient Descent Classifier, Decision Tree, Random Forest. The proposed process of constructing the predictive models is shown in figure 1.

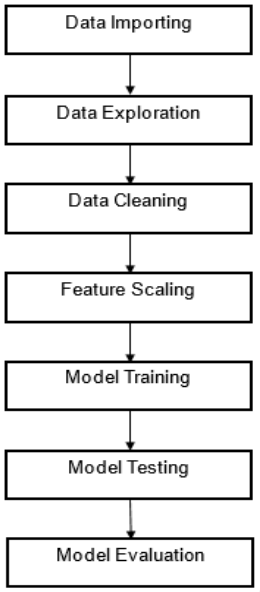

Figure 1. Process of creating the model for predicting CKD.

In the first step, the dataset is imported. In the second step, the dataset is explored to gather insights. In the third step, the dataset is pre-processed by transforming categorical attributes to binary attributes, handling missing values and removing anomalies. In the fourth step, the features are normalized using Min-Max Scaling. In the fifth step, the models are trained on training data using 5-fold cross validation. In the sixth step, predictions are made using the test data. In the last step, the algorithms are evaluated based on the evaluation parameters defined in Section-IV.

B. Dataset

Our research uses a CKD dataset, which is openly accessible at UCI machine learning laboratory. The CKD dataset consists of 24 attributes (i.e. predictors) in addition to the binary class attribute (target variable). Out of the 24 attributes, 11 are numerical attributes, two categorical with five levels, while the remaining parameters are binary and been coded as zero for abnormal instances and one for normality. In the class attribute, one (1) is coded for presence of CKD and zero (0) represents CKD is not present. This dataset contains 400 observations out of which 150 observations do not have chronic kidney disease (not present / NotCKD) and 250 observations, which have chronic kidney disease (present / CKD). Out of the 400 observations, 300 of them are used for the training of classification algorithms and 100 are used to test the result of these algorithms.

The attributes in the dataset are age, blood pressure, specific gravity, albumin, sugar, red blood cells, pus cell, pus cell clumps, bacteria, blood glucose random, blood urea, serum creatinine, sodium, potassium, hemoglobin, Packed cell volume, white blood cell count, red blood cell count, hypertension, diabetes mellitus, appetite, pedal edema, anemia, and class. The dataset is imported using pandas library in Python language.

C. Data Exploration

We explored the dataset using pandas library and plotted the class attribute with respect to the most prominent attributes that contribute to CKD and gathered some insights from the dataset.

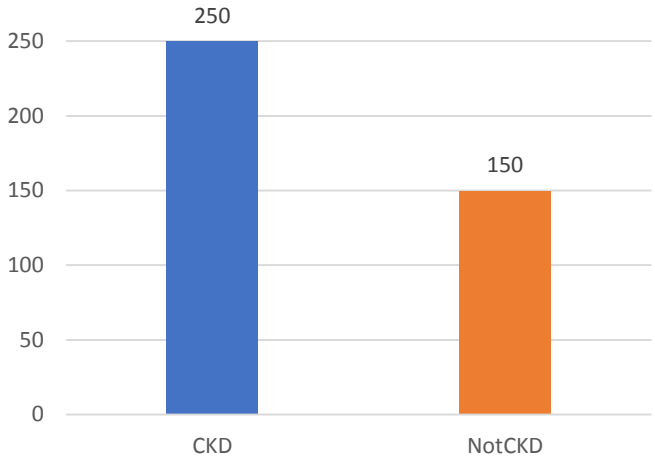

Figure 2. Frequency Distribution Plot of class variable.

Figure 2 shows the frequency distribution of values in the class attribute. It can be seen that there are 250 cases which have Chronic Kidney Disease (CKD) and 150 which don't (Not CKD). This verifies the metadata provided with the dataset.

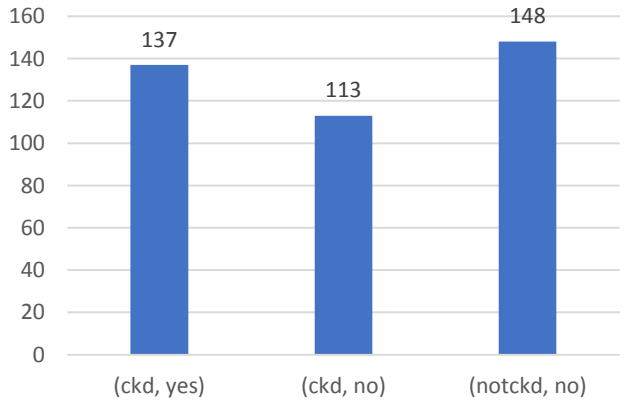

Figure 3. Frequency Distribution Plot of class variable with respect to diabetes.

Figure 3 shows the frequency distribution of class attribute with respect to diabetes attribute. It can be seen that out of the 250 cases that have CKD, 55% (137) of the cases are diabetic.

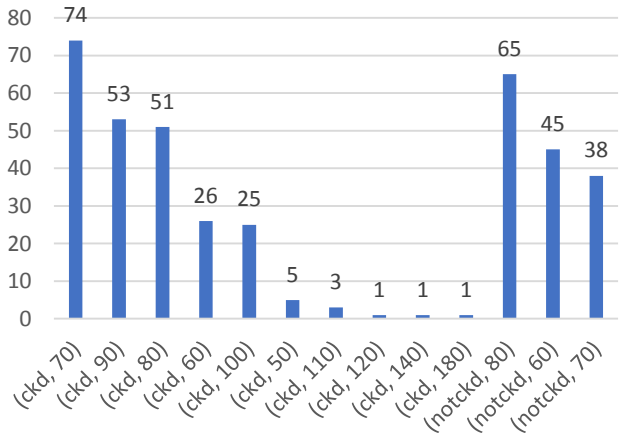

Figure 4. Frequency Distribution Plot of class variable with respect to blood pressure.

Figure 4 shows the frequency distribution of class attribute with respect to blood pressure attribute. It can be seen that Majority of the CKD cases (about 70%) have Blood Pressure greater than 70 mmHg, depicting that these values are of Diastolic Blood Pressure (As Diastolic Blood Pressure greater than 70 is a sign of High Blood Pressure and High Blood Pressure is one of the major factors of CKD).

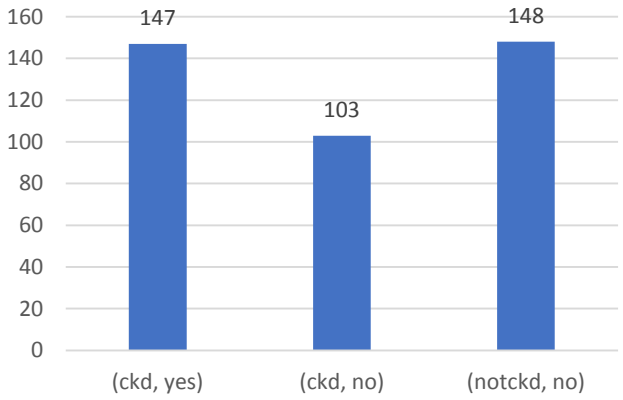

Figure 5. Frequency Distribution Plot of class variable with respect to hypertension.

Figure 5 shows the frequency distribution of class attribute with respect to hypertension. It can be seen that out of the 250 cases that have CKD, 59% (147) of the total CKD cases also have Hypertension.

D. Finding and Removing Anomalies

We analyzed the data for unique values in every attribute and there were certain anomalies such as – 'ckd\t' in class attribute, '\t?' in rc (red blood cell count) and pcv (packed cell volume) attributes, '\t800' in wc (white blood cell count) attribute and so on. These anomalies needed to be removed before the data is fed to different algorithms because the algorithms don't accept string data as input. These anomalies were removed using the following regular expression:

```
for i in ['pcv','wc','rc']:
df[i] = df[i].str.extract('(\d+.\d+|\d+)').astype(float)
```

E. Handling Missing Data

When the data collected is real world data, it will contain missing values. This brings more change in the prediction accuracy. Sometimes these missing values can be simply deleted or ignored if they are not large in number [4]. It is the simplest way to handle the missing data but it is not considered healthy for the model as the missing value can be an important attribute contributing to the disease. The missing values can also be replaced by zero this will not bring any change as whole, but this method cannot be much yielding. In our case, the missing values in the numerical features which contain continuous floating and integer values are replaced by their respective median. The missing values in the nominal features are replaced by zero (0).

F. Feature Scaling

It is the process of normalizing or standardizing the features present in the dataset. The features with different ranges of numerical values can effect the final result, as most of the machine learning techniques focus on the use of magnitude to determine the result. We have used the min-max scaling technique to scale the features. This scaling brings the value between 0 and 1.

$$x' = \frac{x - min(x)}{max(x) - min(x)} \quad (1)$$

G. K-fold cross validation

Cross validation is used to check how well the model is trained without considering the test data. In k-fold cross validation, the dataset is divided into k-folds. Anyone of the fold can be considered as test set and the remaining folds are considered as training set. This process goes on until all the folds are taken as test sets. We have used 5-fold cross validation in our machine learning models.

III. MACHINE LEARNING ALGORITHMS

A. Logistic Regression

Logistic Regression (LR) is a type of linear regression model [5]. LR computes the distribution between the example X

and Boolean class label Y by P(X|Y). Logistic regression classifies Boolean class label Y as follows:

$$P(Y=1/X) = \frac{1}{1 + exp(w_0 + \sum_{i=1}^{n} w_i X_i)} \quad (2)$$

$$P(Y=0/X) = \frac{1}{1+exp(w_0 + \sum_{i=1}^{n} w_i X_i)} \quad (3)$$

B. Support Vector Machine

For the classification problem, the Support Vector Machine (SVM) is the popular data mining method used to predict the category of data [6]. The main idea of SVM is to find the optimal hyperplane between data of two classes in the training data. SVM finds the hyperplane by solving the optimization problem.

$$maxQ(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j d_i d_j x_i^T x_j \quad (4)$$

where $0 \leq \alpha_i \leq C$ for I = 1, 2, …., n.

SVM uses the decision function $f(x)$ defined in the form of kernel function for calculating the output as

$$f(x)=sign\left[\sum_{i=1}^{l} \alpha_i d_i K(x,x_i)+b\right] \quad (5)$$

where $K(x, x_i)$ is the kernel function.

C. K-Nearest Neighbors

K-nearest neighbors (KNN) is the classification method for classifying unknown examples by searching the closest data in pattern space [7]. KNN predicts the class by using the Euclidean distance defined as follows:

$$\boldsymbol{d(x,y)} = \sqrt{\sum_{i=1}^{k} (x_i - y_i)^2} \quad (6)$$

The Euclidean distance d(x, y) is used to measure the distance for finding the k closest examples in the pattern space. The class of the unknown example is identified by a majority voting from its neighbours.

D. Naïve Bayes

Naïve Bayes (NB) are probabilistic classifiers, which are based on Bayes Theorem. In Naïve Bayes, each value is marked independent of other values and features. Each value contributes independently to the probability. The higher the probabilistic value, the higher are the chances of data point belonging to that class or category. Naïve Bayes algorithm uses the concept of Maximum Likelihood for prediction. This algorithm is fast and can be used for making real time predictions such as sentiment analysis. For example, a characteristic item may be seen as an apple in case it is red, round and around 3 sneaks in broadness. Despite whether these features depend on each other or upon the nearness of substitute features, these properties openly add to the probability that this common item is an apple and that is the reason it is known as 'Naïve' [8].

E. SGD Classifier

It is a Logistic Regression Classifier based on Stochastic Gradient Descent Optimization. Stochastic gradient descent (SGD) in contrast performs a parameter update for each training example x(i) and label y(i) as follows.

$$\theta = \theta - \eta \cdot \nabla_\theta J(\theta; x^i; y^i) \quad (7)$$

F. Decision Tree

Decision tree (DT) is the classification method frequently used in data mining task [9]. A decision tree is a structure that includes a root node, branches, and leaf nodes. It divides the data into classes based on the attribute value found in training sample. A Decision Tree Classifier generates the output as a binary tree like structure called a decision tree, in which each branch node represents a choice between a number of alternatives, and each leaf node represents a classification or decision. A Decision Tree model contains rules to predict the target variable. This algorithm scales well, even where there are varying numbers of training observations and large number of attributes [10].

G. Random Forest

Random Forest (RF) is a variant of ensemble classifier consisting of a collection of tree-structured classifiers h (x, yk), which is defined as multiple tree predictors yk such that each tree relies upon the estimations of an arbitrary vector inspected independently and with a similar distribution for all trees in the forest. The randomization is done by random selection of input attributes for producing individual base decision trees [11]. Random forests become different in a way from other methods that a modified tree learning algorithm is utilized that chooses the differentiable candidate in the learning procedure, a random subset of the features. The cause for doing this is the relationship of the trees in a standard bootstrap sample. For example, if one or a couple of features are extreme indicators for the response variable (target output), these features will be chosen in a considerable lot of the decision trees, reasoning them to end up correlated.

IV. EVALUATION PARAMETERS

A. Confusion Matrix

It is a performance measurement for machine learning classification problem where output can be two or more classes. It is a table with 4 different combinations of predicted and actual values [12].

Table 1. Confusion Matrix (CM).

	Predicted Negative	Predicted Positive
Negative cases	TN	FP
Positive cases	FN	TP

Then we may define some evaluation measures.

$$\text{(A) Accuracy} = \frac{TN + TP}{TN + TP + FN + FP}$$

$$\text{(R) Recall, Sensitivity} = \frac{TP}{TP + FN}$$

$$\text{(S) Specificity} = \frac{TN}{TN + FP}$$

V. RESULTS AND DISCUSSION

All machine learning algorithms are trained and tested by the proposed method explained in Section-II and are compared based on the defined evaluation parameters. The confusion matrix of each algorithm is shown in Table 2.

Table 2. Confusion Matrices of all algorithms.

Model	Not CKD	CKD
Logistic Regression	38 (TN)	0 (FP)
	0 (FN)	62 (TP)
Support Vector Machine	36 (TN)	2 (FP)
	2 (FN)	60 (TP)
K-Nearest Neighbour	38 (TN)	0(FP)
	2 (FN)	60 (TP)
Naïve Bayes	38 (TN)	0 (FP)
	3 (FN)	59 (TP)
Stochastic Gradient Descent Classifier	38 (TN)	0 (FP)
	0 (FN)	62 (TP)
Decision Trees	38 (TN)	0 (FP)
	3 (FN)	59 (TP)
Random Forest	38 (TN)	0 (FP)
	0 (FN)	62 (TP)

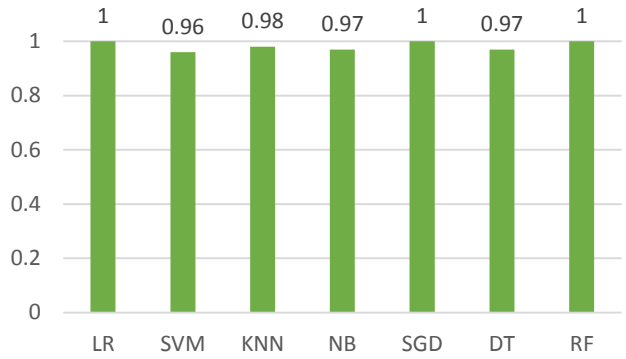

Figure 6. The classification accuracy of algorithms.

Figure 6 shows the accuracy of seven classifiers. From the results, it can be seen that the Logistic Regression (LR), Random Forest (RF) and SGD classifier have the highest accuracy than the others (1.0) while Decision Tree (DT), SVM classifier, Naive Bayes (NB) and KNN have an accuracy of 0.97, 0.96, 0.97 and 0.98 respectively.

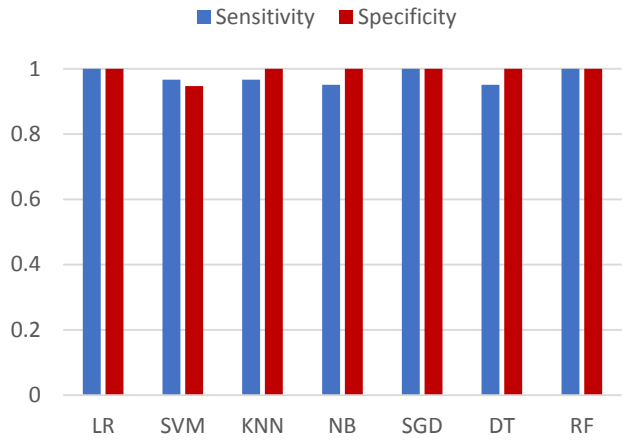

Figure 7. Sensitivity and Specificity of algorithms.

Figure 7 shows the sensitivity and specificity of the seven classifiers. From the results, it can be seen that the Logistic Regression, SGD classifier and Random Forest have the highest sensitivity (1.0) than the others. In case of specificity, Logistic Regression, KNN, Naïve Bayes, SGD Classifier, Decision Tree and Random Forest have a specificity of 1.0 and SVM has a specificity of 0.947.

VI. CONCLUSION

We have trained seven different machine learning algorithms to predict the presence of chronic kidney disease. Of all the other models compared, Logistic regression, SGD Classifier and Random forest provide the best results. These have

surpassed other classifiers and are able to detect the chronic kidney disease more precisely. If these models are trained using a varied and extensive range of attributes, they may result in more accurate predictions. The results would be more assuring if more observations are gathered which results in an increase in the size of dataset. Hospitals and diagnostic centres can use this for faster and digitized analysis for predicting chronic kidney disease.

REFERENCES

[1] Chaitanya Gupte and Shruti Gadewar, "Diagnosis of Parkinson's Disease using Acoustic Analysis of Voice", International Journal of Scientific Research in Network Security and Communication, Vol.**5**, Issue.**3**, pp.14-18, **2017**.

[2] Pallvi Dehariya, "An Artificial Immune System and Neural Network to Improve the Detection Rate in Intrusion Detection System", International Journal of Scientific Research in Network Security and Communication, Vol.**4**, Issue.**1**, pp.1-4, **2016**.

[3] Antje Erler, Martin Beyer, Juliana J. Petersen, Kristina Saal, Thomas Rath, Justine Rochon, Walter E. Haefeli and Ferdinand M. Gerlach, "How to improve drug dosing for patients with renal impairment in primary care – a cluster-randomized controlled trial", BMC Family Practice, Vol.**13**, Issue.**1**, Article.**91**, pp.1-8, **2012**.

[4] S. Venkata Lakshmi, M. K. Meena and N. S. Kiruthika, "Diagnosis of Chronic Kidney Disease using Random Forest Algorithms", International Journal of Research in Engineering, Science and Management, Vol.**2**, Issue.**3**, pp.559-562, **2019**.

[5] R. Xi, N. Lin and Y. Chen, "Compression and Aggregation for Logistic Regression Analysis in Data Cubes", IEEE Transactions on Knowledge and Data Engineering, Vol.**21**, Issue.**4**, pp.479-492, **2009**.

[6] R. G. Brereton, and G. R. Lloyd, "Support Vector Machines for classification and regression", Analyst, Vol.**135**, Issue.**2**, pp.230-267, **2010**.

[7] Galit Shmueli, Nitin R. Patel and Peter C. Bruce, "Data Mining for Business Intelligence: Concepts, Techniques, and Applications in Microsoft Office Excel with XLMiner", Wiley Publishing, pp.**250-268**, **2010**.

[8] Afzal Ahmad, Mohammad Asif and Shaikh Rohan Ali, "Review Paper on Shallow Learning and Deep Learning Methods for Network Security", International Journal of Scientific Research in Computer Science and Engineering, Vol.**6**, Issue.**5**, pp.45-54, **2018**.

[9] J. Ross Quinlan, "C4.5: Programs for Machine Learning by J. Ross Quinlan.", Morgan Kaufmann Publishers, Inc., pp.**17-26**, **1993**.

[10] Deepika Mallampati, "An Efficient Spam Filtering using Supervised Machine Learning Techniques", International Journal of Scientific Research in Computer Science and Engineering, Vol.**6**, Issue.**2**, pp.33-37, **2018**.

[11] M. S. Anbarasi and V. Janani, "Ensemble classifier with Random Forest algorithm to deal with imbalanced healthcare data", In International Conference on Information Communication and Embedded Systems (ICICES), Chennai, India, pp.**1–7**, **2017**.

[12] Hanyu Zhang, Che-Lun Hung, William Cheng-Chung Chu, Ping-Fang Chiu and Chuan Yi Tang, "Chronic Kidney Disease Survival Prediction with Artificial Neural Networks", In IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Madrid, Spain, pp.**1-6**, **2018**.

Authors Profile

Bhawna Sharma completed B.Tech. in CSE from Pondicherry Engineering College, Poncicherry, in 1996 and M.S. Software Systems from BITS Pilani in the year 2003. She has done Ph.D. in Computer Science & IT from University of Jammu, J&K in 2015 and is currently working as Associate Professor in Department of Computer Engineering, Government College of Engineering & Technology, Jammu, J&K since 2001. She is a life member of IE(India), CSI and ISC. She has published many research papers in reputed International & National journals and conferences. Her areas of interest include Formal Languages & Automata Theory, Software Systems, Soft Computing, Computer Networks and Big Data Analytics. She has more than 18 years of teaching experience.

Sheetal Gandotra obtained B.E. in Computer Engineering from Pune University in 1996 and M.E. in Computer Science & Engineering from Panjab University in the year 2005. She is currently working as Associate Professor in Department of Computer Engineering, Government College of Engineering & Technology, Jammu, J&K since 2001. She has published many research papers in reputed International & National journals and Conferences. Her areas of interest include Image Processing, Data Structures and Operating Systems. She has more than 18 years of teaching experience.

Utkarsh Sharma is currently pursuing B.E in Computer Engineering from Government College of Engineering and Technology, Jammu, J&K.

Rahul Thakur is currently pursuing B.E in Computer Engineering from Government College of Engineering and Technology, Jammu, J&K.

Alankar Mahajan is currently pursuing B.E in Computer Engineering from Government College of Engineering and Technology, Jammu, J&K .

