ISSN: 2229-6956 (ONLINE)
DOI: 10.21917/ijsc.2024.0462

NORMALIZATION AND FEATURE SELECTION USING ENSEMBLE METHODS FOR CROP YIELD PREDICTION

A. Chitradevi and N. Tajunisha
Department of Computer Science, Sri Ramakrishna College of Arts and Science for Women, India

Abstract

In machine learning study proposes an ensemble-based strategy for both feature selection and data standardization to enhance model performance and interpretability. To maintain consistency across datasets, it employ average filling and weighted K-means clustering. Weighted K-means assigns distinct values to samples based on their distances to cluster centers, offering a more precise representation of the data distribution. Meanwhile, average filling replaces missing values with the average of corresponding features, ensuring a complete dataset for subsequent analysis. For feature selection, adopt an ensemble approach that combines Random Forest (RF) with Logistic Regression (LR) and ElasticNet. RF captures feature importance through tree-based analysis, while LR and ElasticNet provide additional insights into feature relevance and coefficients. This amalgamation aims to provide a comprehensive understanding of feature importance within the dataset. Principal Component Analysis (PCA) is employed to reduce dataset complexity while preserving key properties, facilitating more effective feature selection. By identifying orthogonal components that best explain data variation, PCA enables efficient representation and feature selection. In the final stage, Support Vector Machines (SVM) are utilized for categorization. SVM, a powerful classification method, establishes strong decision boundaries that optimize the gap between classes. Leveraging the selected features, the SVM model effectively categorizes new instances.

Keywords:

Dataset Normalization, Feature Selection, Weighted K-Means Clustering, Decision Tree Regressor, Random Forest

1. INTRODUCTION

The performance and interpretability of prediction models in machine learning may be greatly improved by preprocessing processes like dataset normalization and feature selection [1]. Normalization ensures that characteristics are on analogous scales, thereby reducing the impact of variables with greater magnitudes [2]. Feature selection, on the other hand, seeks to identify the most pertinent predictors by removing irrelevant or redundant features, thereby enhancing model efficiency and interpretability [3].

Normalization techniques such as standardization and min-max scaling have been extensively utilized historically [4]. Nonetheless, recent advancements in ensemble methods have demonstrated optimistic results in overcoming the limitations of conventional approaches [5]. Ensemble methods predict by combining multiple models or algorithms, capitalizing on the strengths of each component to improve overall performance [6]. Ensemble methods provide the potential for more robust and accurate preprocessing in the context of normalization and feature selection [7].

This describes an ensemble approach that incorporates multiple normalization and feature selection techniques [8]. We propose a novel method for normalizing datasets that combines weighted K-means clustering, average filling, and Decision Tree Regressor [9]. This combination permits adaptive normalization by allocating various weights to data points in accordance with their proximity to cluster centroids [10]. In addition, the average infill technique manages absent values, ensuring that valuable data is preserved during the normalization procedure [11]. The Decision Tree Regressor identifies nonlinear relationships, thereby enhancing the normalization process [12].

In addition, ensemble feature selection is introduced by integrating Random Forest (RF), Logistic Regression (LR), and ElasticNet with Principal Component Analysis (PCA). Using the assets of multiple algorithms, this ensemble approach seeks to identify the most informative features [13]. While LR and ElasticNet use regularization techniques to select meaningful predictors, RF provides a robust feature ranking based on variable importance [14]. PCA reduces dimensionality while preserving important features, thereby improving model efficiency [15].

Our proposed method provides a comprehensive paradigm for dataset normalization and feature selection by integrating these ensemble methods [16]. The combination of weighted K-means clustering, average filling, and Decision Tree Regressor guarantees a more precise and trustworthy normalization procedure [17]. The ensemble feature selection approach using RF, LR, ElasticNet, and PCA guarantees the retention of only the most informative features, thereby enhancing model interpretability and minimizing over fitting [18]. In the subsequent sections, we will delve into the specifics of our proposed method for normalizing datasets and selecting features using ensemble methods [19-21]. Experimental results on a variety of datasets will demonstrate the efficacy and efficiency of our method, emphasizing its potential for enhancing predictive modelling tasks in real-world applications [22]. The results of this study will help farmers make informed decisions by enabling them to anticipate the yield of their crop before planting it in the field. Farmer assistance in optimising agricultural yield requires prompt guidance on projecting future crop yield and analysis [25]. Agricultural yields for maize and potatoes from several sources, together with weather information. Support Vector Regressor, Polynomial Regression, and Random Forest were used to analyse the gathered data. Temperature and rainfall were employed as predictors [26]. A novel machine learning-based agricultural decision support system in this study. Our primary goal was to determine how climate change would affect the productivity of agricultural crops in East African nations. To provide farmers and decision-makers with a crop yield prediction system, we combined data from many sources, including the climate, crop productivity, and pesticide use [30].

1.1 MOTIVATION

Normalization and feature selection play vital roles in improving the performance and interpretability of machine learning models. However, there is no foolproof strategy for doing these errands, and several approaches may provide contrasting outcomes depending on the dataset and the situation at hand. Therefore, there is a need for innovative approaches that can leverage the strengths of multiple techniques to achieve optimal normalization and feature selection outcomes. To overcome the difficulties of dataset normalization and feature selection, we present a multi-method ensemble-based technique in this research. By integrating weighted K-means clustering and average filling techniques for dataset normalization, we aim to capture the underlying data distribution more accurately and handle missing values effectively. This approach can lead to a more comprehensive and representative dataset for subsequent analysis. For feature selection, we employ an ensemble approach that combines Random Forest (RF) with Logistic Regression (LR) and ElasticNet. RF is known for its ability to assess feature importance through tree-based analysis, while LR and ElasticNet provide additional insights into feature relevance and coefficients. By combining these techniques, we can gain a deeper understanding of the importance and contribution of each feature in the dataset.

2. BACKGROUND

Araque et al. [1] proposes a two-dimensional taxonomy to categorize ensembles of classifiers and features, combining traditional hand-crafted features and automatically retrieved embedding features. Multiple ensembles' classification results are measured against a deep learning benchmark. Conţiu and Groza Elghazel and Aussem [4] present a framework called RCE for feature selection in unsupervised learning using an ensemble of clustering methods. The framework demonstrates the capability to build meaningful clusters with fewer features, improving clustering accuracy on various datasets. Gaikwad and Thool [6] propose the ensemble bagging approach for network-wide anomaly detection in intrusion detection systems. The Bagging ensemble with the REPTree basis classifier is evaluated and compared to standard machine learning methods. Laradji et al. [8] focus on software defect prediction and explore feature selection strategies using ensemble learning. They find that greedy forward selection outperforms correlation-based methods and demonstrate the effectiveness of ensemble learning with duplicated features and skewed datasets. Prusa et al. [11] to take advantage of ensemble learners while coping with high dimensionality, you may mix bagging and boosting with feature selection. The proposed Select-Boost approach outperforms Select-Bagging and individual feature selection methods in sentiment analysis tasks. Safiyari and Javidan [13] concentrate on predicting survival rates for lung cancer patients using ensemble learning. They preprocess the data, employ correlation-based feature selection, and reduce the dimensionality before training the ensemble models. Tan et al. [16] industrial electricity demand forecasting: present a deep ensemble learning model. In terms of accuracy metrics, their state-of-the-art models are surpassed by their hybrid ensemble approach and improved loss function. Verma et al. [18] explore the use of ensemble methods, including Bagging, AdaBoost, and Gradient Boosting classifiers, for predicting skin diseases. They employ various machine learning classification techniques and achieve superior accuracy using Gradient Boosting with feature selection. Yekkala et al. [20] investigate the combination of Particle Swarm Optimization (PSO) and ensemble classifiers for cardiac issue prediction. PSO is used for feature selection, and Bagged Tree Ensemble Classifier significantly improves accuracy for precise prognoses. Bharadiya et al. [22] Remote sensing can be a fast, economical, and efficient way to monitor, evaluate, and estimate crop yield. An extensive evaluation of the application of DL approaches for remote sensing data-based agricultural production forecasting has been carried out in the study. Oikonomidis et al. [24] proposed approaches are the XGBoost as a single model, the XGBoost with scaling, the XGBoost combined with scaling and feature selection methods, the hybrids CNN-XGBoost, CNN-DNN, CNN-RNN, and CNN-LSTM models. Olofintuyi et al. [27] In comparison to statistical models, LSTM models are better suited for yield prediction when dealing with time series data. Additionally, the total model's performance and resilience can be improved by using RNN to extract yield features and CNN to extract climatic features. Lastly, a reliable reference for yield prediction is anticipated from the model. Seireg et al. [28] study, LGBM, GBR, XGBoost, and Ridge were consuming the assembling and falling techniques. Pham et al. [29] constructed a framework for evaluating several feature dimension reduction methods, specifically FS, FX, and FSX, in the process of creating crop yield prediction models based on machine learning techniques.

Table.1. Comparison for existing works

Method	Advantage	Limitation
Deep learning [1]	Experiments utilizing the proposed models on a range of public datasets demonstrate that the integrated models beat the deep learning baseline in terms of F1-Score, showing the strategy's effectiveness.	First, the manual extraction of features in surface approaches requires domain expertise and can be time-consuming and subjective.
Random Cluster Ensemble [5]	The advantage of the proposed Random Cluster Ensemble (RCE) method for feature selection in unsupervised learning is its ability to estimate feature importance and select relevant features effectively.	One limitation of the Random Cluster Ensemble (RCE) method for feature selection in unsupervised learning is that it relies on the assumption that the selected features are representative of the underlying data distribution.
Ensemble Learning [12]	Using Select-Boost in conjunction with feature selection is an effective strategy for tweet sentiment classification since it overcomes two major obstacles: low-quality data and a large data set.	Using ensemble learning approaches like Select-Boost in combination with feature selection for tweet sentiment categorization may be time-consuming and resource-intensive.
Extra tree	The advantage of the proposed study's	One limitation of the proposed study is that it

classifier [19]	approach, which combines six different data mining classification techniques with ensemble methods (Bagging, AdaBoost, and Gradient Boosting), is that it leverages the strengths of multiple algorithms to improve the prediction accuracy.	does not provide a comprehensive analysis or comparison of the individual data mining classification techniques used.

1.2 PROBLEM DEFINITION

In machine learning pipelines, normalizing datasets and selecting relevant features using ensemble techniques is a common challenge that this research attempts to solve. While feature selection seeks to discover the most informative and discriminatory characteristics for constructing strong models, normalization is crucial to guarantee that the data is uniform and similar across features. This research intends to offer a comprehensive solution to these difficulties by proposing an ensemble-based approach that combines methods such as weighted K-means clustering, average filling, RF, LR, ElasticNet, PCA, and SVM. The used normalization methods provide a true picture of the data distribution and deal with missing values, while the ensemble feature selection strategy use a number of strategies to zero down on the most important features while simultaneously decreasing the dataset's dimensionality. The ultimate goal of the work is to use these methods to improve the precision and interpretability of machine learning models across a variety of settings.

3. MATERIALS AND METHOD

We provide a thorough strategy for dataset normalization, ensemble feature selection, and classification using SVM in this section. To normalize the dataset, we employ weighted K-means clustering to capture the data distribution accurately, average filling to handle missing values, and a decision tree regressor to address outliers. For feature selection, we utilize an ensemble approach combining RF with Logistic Regression (LR) and ElasticNet. This ensemble method allows us to obtain a comprehensive understanding of feature importance. To further enhance feature selection, PCA is applied to reduce the dimensionality while preserving important characteristics. Finally, for classification, we employ SVM, a powerful algorithm that maximizes the margin between different classes, resulting in robust decision boundaries. We want to increase the performance and interpretability of our classification model by combining these strategies.

1.3 DATASET COLLECTION

The dataset available at the provided Kaggle link is titled "Crop Production in India." It is a comprehensive collection of agricultural data that focuses on crop production in various states of India. https://www.kaggle.com/datasets/abhinand05/crop-production-in-india The dataset encompasses information from the years 1997 to 2015 and includes details such as crop type, crop area, yield, and production for different crops across different districts and states in India. This dataset is valuable for analyzing and understanding crop production patterns, trends, and variations in different regions of India over time.

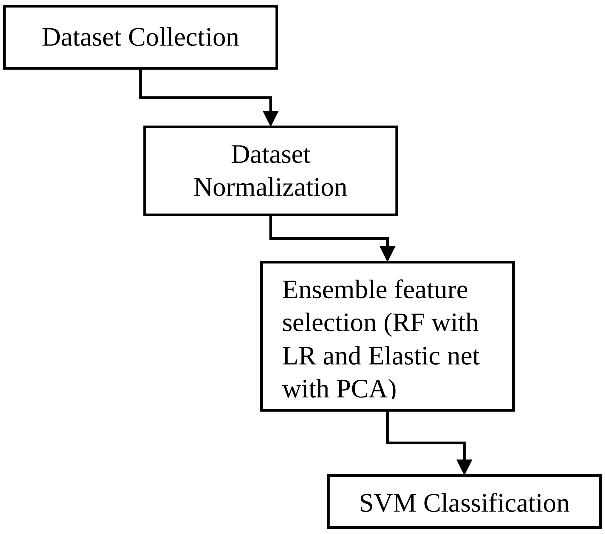
Dataset Collection
Dataset Normalization
Ensemble feature selection (RF with LR and Elastic net with PCA)
SVM Classification

Fig.1. Overall Flow Diagram

1.4 DATASET NORMALIZATION USING WEIGHTED K-MEANS CLUSTERING, AVERAGE FILLING, DECISION TREE REGRESSOR

After data collection, the collected dataset undergoes several normalization techniques. Weighted K-means clustering is employed to identify clusters of similar data points, with weights assigned based on their relevance. This allows the algorithm to prioritize certain data points during normalization. Average filling is applied to handle missing or incomplete data, replacing missing values with the average value of available data for each feature. This ensures the dataset remains complete and maintains statistical properties.

1.4.1 K-means Clustering:

One of the most basic unsupervised learning algorithms, K-Means, takes on the ever-present problem of grouping. Assuming k clusters, the method provides a fast and easy way to divide data into distinct groups.

The main idea is to find k cluster centroids. It's critical to carefully place these centroids since each one produces a different result. As a result, it is best to place them as far away as feasible. The next step is to determine the centroid of the data set for each point. When all initial concerns have been addressed, the first phase is over. New nuclei, or centroids, for the clusters generated in the previous phase need the computation of k. After these k new centroids have been found, the same data points will need to be re-bound to the closest new centroid again. We seem to have set off a recursive process. This loop might lead to the discovery that the positions of the k centroids gradually shift until no more adjustments need to be made. That is to say, no longer do centroids shift.

Finally, the purpose of this approach is to reduce the square of the mistake. The point of view

$$W(S,C)=\sum_{k=1}^{K}\sum_{i\in S_k}\left\|y_i-c_k\right\|^2 \tag{1}$$

Specifically, S is a K-cluster partition of the M-dimensional feature space into non-empty, non-overlapping clusters S_k, where each entity set represented by vectors y_i is linked with a centroid c_k k=1. The algorithm consists of the following operations:

Step 1: Set k nodes in the space defined by the items to be clustered. These are the first centers of each group.

Step 2: Put things where their centroid is closest to.

Step 3: After everything has a home, we can redo the k-center point calculations.

Step 4: Keep doing Steps 2 and 3 until the axes of the centroids stop shifting.

1.4.2 Weighted k-means:

The weighted k-means method has many benefits over the original k-means algorithm, including improved resilience against outliers, higher quality clusters, and quicker convergence. It has seen extensive usage in image segmentation, data mining, and bioinformatics, among other applications.

A k-partitioning technique takes a collection of n items, $D = x_1, x_2, x_n$, and a positive number K, and divides it into precisely K distinct subsets, $D_1, \in D_k, D_k$. Clustering theory states that objects with similar properties are more closely connected to one another than to all other objects. The difficulty of deciding may be reduced by developing a cost function that evaluates the success of clustering for each subset of the dataset. The characteristic of each gene is shown here as an integer. So, the amount of characteristics an object possesses may be represented as a row vector of real numbers of length d. For the sake of argument, let's assume that all of the data in the dataset is complete and that each item has the same amount of qualities. Let there be n objects in the set $x_i \in D_k$ to symbolise. For the sake of brevity, we shall abbreviate the j^{th} property of x_i as x_{ij}. A D attribute matrix for an object set is denoted by $X = (x_{ij})$.

$$j_G(\Delta) = \sum_{k=1}^{K} \sum_{x_i \in D_k} (x_i - m_k) G (x_i - m_k) \tag{2}$$

$$m_k = \frac{1}{n_k} \sum_{x_i \in D_k} x_i \tag{3}$$

G is a positive symmetric weighted matrix, where n_k and m_k are the means and the size of D_k, respectively. A symmetric positive matrix G^* meeting Eq.(4) is sought via the weighted k-means approach such that the desired subset, indicated by *.

$$j_G(\Delta^*) = \min_{\Delta} j_G(\Delta) \tag{4}$$

When $j_g(\Delta^*)$ is computed by multiplying a partition by a weighted matrix G, the output might vary. Thus, it is necessary to normalise the weighted matrix. The G determinant is assumed to be 1 in this investigation.

$$(\det(G)) = 1 \tag{5}$$

Condition (5) is met by virtue of the fact that $G = I$ in (5), and the cost function and optimum goal of a typical k-means algorithm are defined by Eq.(6) and Eq.(7), respectively.

Let us say that a collection of data, denoted by $X = x_1, \ldots x_n$, exists in a d-dimensional Euclidean space R^d. The k-means approach seeks to minimise an objective function to partition a data set X into a desired number of clusters, k:

$$P(U,Z) = \sum_{i=1}^{n} \sum_{l=1}^{k} U_{il} \sum_{j=1}^{d} d\left(X_{ij}, Z_{lj}\right) \tag{6}$$

to which $U_{il} \sum_{j=l}^{d} d(X_{ij}, Z_{lj})$, where signifies that the ith data point X_i is part of the U_i^{th} cluster, and $[U]$ is a $[n * k]$ partition matrix, are binary variables. If Z is a collection of k-vectors representing the cluster centers, then the distance between the ith data point and the l^{th} cluster center on the j^{th} variable is denoted by $d(X_{ij}, Z_{lj})$. The following conditions must be true to reduce $P(U,Z)$:

$$Z_{lj} = \frac{\sum_{i=1}^{n} U_{il} x_{ij}}{\sum_{i=1}^{n} U_{il}} \quad \text{for } 1 \le l \le k \text{ and } 1 \le j \le d \tag{7}$$

$$\begin{cases} u_{il} = 1 \quad if \sum_{j=1}^{d} d\left(X_{ij}, Z_{lj}\right) \le \sum_{j=1}^{d} d\left(X_{ij}, Z_{li}\right) \text{ for } 1 \le t \le k \\ u_{it} = 0 \quad \text{for } t = 1 \end{cases} \tag{8}$$

Using Equations and, the k-means method may be seen as a recursive approximation of Picard's fixed point. The k-means technique first operates from the perspective $P(U,Z)$, in which each variable is given equal weight. There might be a great deal of noise in the data due to uncontrolled variables in the gathering process. that maybe a solution can be found using weighted k-means. One possible representation of the weights for m variables.

$$P(U,Z,W) = \sum_{i=1}^{n} \sum_{l=1}^{k} U_{il} \sum_{j=1}^{d} W_j^d d\left(X_{ij}, Z_{lj}\right) \tag{9}$$

The equation was derived by Yang and Wu, who had previously used the idea of adding an exponential distance weight to an equation. Taking into account the spatial restrictions used in FCM, an Equation is analysed.

1.4.3 Average Filling:

Three hundred individual stems were labelled in each plot for consistency. During the time between anthesis and harvest, thirty randomly tagged spikes were harvested from each individual plant between 9:00 and 11:00 h, once every five to seven days. All of the spikes were sniped off at the base and stored in a sealed plastic bag with a label. To stop any more evaporation, the bags were put on ice within the foam container. After that, we placed each sample in a labelled paper bag and dried it at 105 degrees Celsius for 30 minutes (to de-enzyme it) and then at 75 degrees Celsius until the weight was consistent. By hand, the grain was meticulously threshed off the spikes and separated into individual grains and the rest of the spike. Each plot's aggregate grain weight was measured. We counted hundreds of grains at random, mixed them well, and then weighed them three times. Spike weight as a fraction of grain weight:

$$GPS(\%) = \frac{TGW_{dry}}{TSW_{dry}} \times 100\% \tag{10}$$

Grain percentage of spike weight (on a dryweight basis) is denoted as GPS(%), where TGW_{dry} and TSW_{dry} are measured in grams, respectively.

$$SMC(\%) = \frac{TSW_{fresh} - TSW_{dry}}{TSW_{fresh}} \times 100\% \tag{11}$$

Grain weight (GW) with time for the evaluated winter wheat under varying water and fertilizer availability was best fit by a sigmoid growth function:

$$GW = GW_{\max}\left(1+\frac{t_e - t}{t_e - t_m}\right)\left(\frac{t}{t_e}\right)\frac{t_e}{t_e - t_m} \text{ if } 0 \le t \le t_e \quad (12)$$

Grain weight (*mg*), days since an *ds*, and plants produced (*n*) are entered into the equation. Grain weight reaches its maximum value, GW_{max}, at *te*, the end of growth, and the maximum filling rate emerges at tm.

1.4.4 Decision Tree Regression:

For each class, we construct a regression tree for use in the soft classification process (see Fig.1). Each regression tree's feature vector is comprised of pixel intensity values over several bands, while the target vector is comprised of each pixel's known class proportions (referred to as soft reference data). The anticipated class proportions are derived after feeding the intensity values into each regression tree. The following demonstrates the method for constructing the regression trees from the training dataset,

```
do
pixels' intensity readings over many bands as independent factors;
Put in a pixel's known class-$i$ percentage as the dependent variable;
build the regression tree $i$ for class $i$;
for class $i$ =1,. . ., $M$
where $M$ is the number of classes.
```

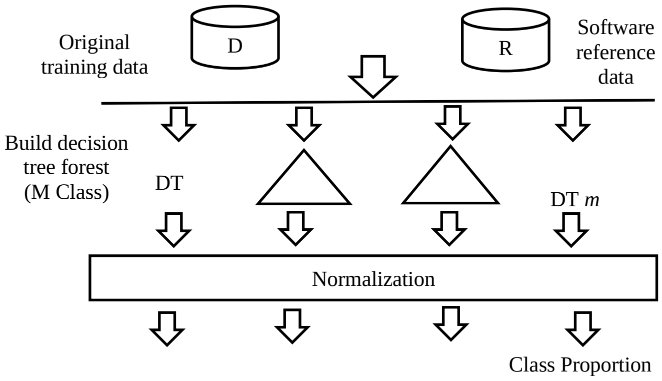

Fig.2. Soft categorization of remote sensing data using a decision tree regression technique

```
do
Pixel intensity values across many frequency bands may be inputted;
class $i$: conduct the $i$ regression tree;
The percentage of pixels that belong to class $i$ is the result of the $i^{th}$ regression tree node;
for class $i$ =1,. . ., $M$.
```

Soft classification outputs for a pixel are commonly scaled from 0 to 1 so that they more accurately represent the class proportions inside a pixel region on ground. Therefore, *DT*(*i*), where *i* = 1,..., *M*, stands for the projected class proportions by tree *i*, and the normalisation of these proportions is as,

$$p(i) = \frac{DT(i)}{\sum_i DT(i)} \; i{=}1,\ldots,M \quad (13)$$

The accuracy of the sorting is assessed once it has been done. Traditional error matrix based measures are normally reserved for examining the correctness of a hard classification, whereas a fuzzy error matrix based measure may be used to evaluate the efficacy of a soft classification. We evaluate DTR-based soft categorization in this work by means of the latter two metrics.

By recursively partitioning space in such a way that samples with similar labels are clustered together, a decision tree may be built from a collection of training vectors x_i R^n, where *i* = 1, 2, 3,..., *n*.

Put *Q* in place of the data at the m^{th} node. Separate the information into q_{left} and q_{right} categories for each split = (*j*, t_m) combination of a feature *j* and a threshold t_m.

$$q_{left}(\theta)=(x,y)|x_j <= t_m \quad (14)$$

$$q_{right}(\theta)=q\backslash q_{left}(\theta) \quad (15)$$

Different impurity functions $H(q_{left}(\theta))$ are used to compute the impurity at m for different problem types (regression vs. classification).

$$G(q,\theta) = \frac{n_{left}}{n_m} H\left(q_{left}(\theta)\right) + \frac{n_{right}}{n_m} H\left(q_{right}(\theta)\right) \quad (16)$$

Select the parameters that minimizes the impurity:

$$\theta^* = \text{argmin}_\theta\, G(Q,\theta) \quad (17)$$

Repeat until N_m minsample or N_m = 1 is attained to get the maximum depth where q_{left} and q_{right} are subsets. To minimize the L1 error, the median values at the terminal nodes of the spline are used, and the L2 error is minimized by using the mean values at the terminal nodes of the spline.

Standard Deviation:

$$y_m = \frac{1}{N_m}\sum_{i\in N_m} y_i \quad (18)$$

$$H(X_m) = \frac{1}{N_m}\sum_{i\in N_m} \left(y_i - y_m\right)^2 \quad (19)$$

Absolute Mean of Error:

$$y_m = \frac{1}{N_m}\sum_{i\in N_m} y_i \quad (20)$$

$$H(X_m) = \frac{1}{N_m}\sum_{i\in N_m} \left|y_i - y_m\right| \quad (21)$$

where X_m represents the X_m^{th} node's training data

1.5 ENSEMBLE FEATURE SELECTION USING RF WITH LR AND ELASTICNET WITH PCA

After data normalization, the most salient features may be isolated using ensemble feature selection techniques. In order to achieve a more reliable and complete feature subset, it is necessary to combine different feature selection approaches. Each feature's significance is calculated using a combination of

Logistic Regression (LR) and Random Forest (RF). RF measures the impact of features on model accuracy, while LR provides insights into the individual effects of features on the target variable. By combining the results from both algorithms, a more reliable ranking of feature importance is obtained.

1.5.1 Random Forest:

The rationale is that if a feature is truly important for prediction, permuting its values should significantly degrade the model's performance. Random Forest's feature significance scores have several potential applications in the context of feature selection. Selecting the attributes with the highest significance ratings is a frequent strategy. This helps in identifying the most relevant features for the prediction task. The number of features to select can be determined based on domain knowledge, trial and error, or using techniques like cross-validation.

Another approach is to set a threshold for feature importance and select all features whose importance exceeds the threshold. This allows for a more flexible selection criterion, where the threshold can be adjusted based on the desired number of features or the desired level of feature relevance. Random Forest's feature selection process provides several benefits. It can handle interactions and non-linear relationships between features, making it effective in identifying both individual and combined feature importance. It is also robust to noisy or redundant features, as it considers the collective performance of all trees in the forest. Additionally, Random Forest can handle both numerical and categorical features, which is advantageous when dealing with diverse datasets.

The convergence theorem, generalized error, and unconventional estimations form the foundation of Random Forest. Following is the formula for a random forest:

$$\{h(X,\emptyset_k),\ k=1,2,\ldots,K\} \tag{22}$$

The X-dimensional collection of sample-condition attributes, the k-dimensional baseline classifier parameter, and the s-dimensional sample size:

$$T=\{(x_i,y_i),x_i\in X,\ y_i\in Y,\ i=1,2,\ldots,N\} \tag{23}$$

A collection of M-dimensional attribute vectors is denoted by X, whereas Y is the determining factor.

Random forest generalization mistakes are as follows:

$$PE^{*def}=P(x,y)\ (av_k\ I(h(X,\emptyset_k)=Y)\text{-max}_{j=y}\ av_k\ I(h(X,\emptyset_k)=j)<0) \tag{24}$$

It quantifies how wrong a random forest is in classifying a specific dataset. The following convergence theorem existed during the period K:

$$PE^{*}\xrightarrow{a,s}P_{X,Y}(P_o(I(h(X,\emptyset_k)=j)<0\text{-max}_{j=y}\ P_o(h(X,\emptyset_k)=j)<0) \tag{25}$$

The generalized error limits of random forests are obtained by combining Hoeffding's inequality and Chebyshev's inequality with Eq.(26):

$$PE^{*}\le\frac{p\left(1-s^{2}\right)}{s^{2}} \tag{26}$$

where s is the basic classifier's accuracy and p is the correlation between the two.

1.5.2 Logistic Regression:

The probabilities of occurrences that may be classified into two groups are predicted using the statistical method of Logistic Regression. This supervised learning method is used in many fields, including machine learning, statistics, and even medical research. While linear regression is used to predict continuous values, logistic regression is used to estimate the probability of a binary outcome based on a collection of continuous predictor factors and a binary target variable. Any real number may be converted into a probability value between zero and one using the sigmoid function.

Maximum likelihood estimate is used to fit a logistic function to the training data, which is how the logistic regression algorithm gets its desired results. The approach optimizes the logistic function's parameters (coefficients) during training such that the discrepancy between the probabilities predicted by the function and the actual binary labels in the training data is as little as possible. Optimization methods like gradient descent are often used for this purpose.

Applying the learnt coefficients to the input variables and determining the appropriate probabilities is how the logistic regression model is put to use for prediction once it has been trained. A decision threshold may be used to classify the data into two sets with different probabilities. If the threshold is set to 0.5, for instance, only cases with projected probability more than 0.5 are assigned to one class, while those with probabilities less than 0.5 are assigned to the other class. Logistic Regression has several advantages, including simplicity, interpretability (coefficients can be directly interpreted as the impact of the input variables on the probability), and efficiency in training and prediction. It accepts both numeric and categorical data, and may be expanded to deal with multi-class classification issues using methods like one-vs-rest or multinomial logistic regression.

Simple (two-variable) regression and multiple regression are both subsets of the basic single-equation linear regression model, which may be written mathematically as:

$$Y=a+\sum_{i=1}^{k}b_i x_i+u \tag{27}$$

where Y is the result, $Y=x_1,x_2,x_i,\ldots,x_k$ are the k independent variables, a and b_i are regression coefficients standing in for the model parameters for a given population, and u is a stochastic disturbance-term standing in for the effect of unspecified independent variables and/or a totally random element in the specified relationship.

1.5.3 RF with LR:

Ensemble feature selection is a strategy for determining the most significant qualities or features by combining the capabilities of the Random Forest (RF) and Logistic Regression (LR) algorithms. This method seeks to improve the accuracy and robustness of feature selection by using the distinct benefits of each algorithm.

Random Forest assesses feature relevance by calculating the influence of each feature on total model accuracy. Random Forest Feature Selection are the Train a random forest model with the training dataset T using K decision trees. And calculate the feature importance scores for each feature in the random forest model. Then rank the features based on their importance scores in descending order.

Logistic Regression, on the other hand, predicts the link between continuous predictors and a binary result. Logistic

Regression Feature Selection are to select the top N features from the ranked feature list obtained from the random forest. And train a logistic regression model using the selected features and the training dataset *T*.

The outputs of Random Forest and Logistic Regression are merged in ensemble feature selection to produce a thorough rating of feature value. A more accurate and robust evaluation of feature relevance is produced by integrating the rankings or significance scores from both algorithms.

1.6 ELASTICNET

For the ElasticNet, 'loss + penalty' is the objective function:

$$\min_{\beta_0,\beta}\frac{1}{N}\sum_{i=1}^{N} w_i I\left(y_i,\beta_0+\beta^T x_i\right)+\gamma\left(\frac{(1-\alpha)\|\beta\|^2}{2}+\alpha\|\beta\|_1\right) \quad (28)$$

The symbol w_i stands for the observational value. $I_l(y)$ stands for the influence on the negative logarithm of the likelihood of making an observation. The regularization parameter l (whose functional form is model-specific) calculates the shrinkage, 2 is the L2-norm of, 1 is *t*, and the ElasticNet penalty sets the weights for ridge and lasso regression. Because no one part is more crucial than any other,

$$\sum_{i=1}^{N} w_i = N \quad (29)$$

$$w_i = M/n_i \quad (30)$$

where the sum of the records in the batch of which *i* is a member is denoted by n_i.

1.6.1 *Principal Component Analysis:*

Since the standard PCA method incorporates all training images in the eigenspace calculation, it does not account for class differentiation. Finding the eigenvector might be a challenging intermediary step if there are a lot of training photos or if the picture dimensions are high. This is because updating a conventional PCA model with more training photos requires recalculating the eigenspace, eigenvalues, and feature vectors for each image, which is a very inefficient use of computational resources. The training process in Superior PCA has been considerably simplified by the adoption of a novel training and projection technique. In order to build an eigensubspace and a set of feature parameters, Superior PCA first filters through the training photographs and categorizes the persons inside them. Choose the subject whose eigensubspace best approximates the test image.

Step 1: Let the training set of all images X can be described as

$$X=\{X_1,X_2,X_3\ldots X_L\} \quad (31)$$

Step 2: Compute the mean vector of all training images of i^{th} person.

$$X_I=\frac{1}{N_I}\sum_{k=1}^{N_i} X_k^i \ \left(i=1,2,\ldots,l\right) \quad (32)$$

Step 3: Compute the covariance of the training set of the i^{th} person

$$S_{x_i}=\frac{1}{N_I}\sum_{k=1}^{N_i}\left(X_k^i-X_i\right) \quad (33)$$

Step 4: 4. Compute Matrix X_i S_{m} largest eigenvalues I_j^u, where j = 1,2...,m

1.6.2 *ElasticNet with Principal Component Analysis:*

By integrating ElasticNet and PCA into a single ensemble technique, feature selection performance is enhanced. In order to better understand a dataset, principal component analysis may be used to reduce the number of dimensions used to describe it by finding a smaller collection of orthogonal axes, or principle components, that capture the bulk of the variance. ElasticNet, on the other hand, is a regularization technique that performs feature selection by shrinking the coefficients of less relevant features while encouraging sparsity.

ElasticNet with Principal Component Analysis

Input:

- Dataset X consisting of n samples and m features: $X = [x_1, x_2, ..., x_n]$, where x_i is a m-dimensional feature vector.
- Number of desired principal components to retain after PCA: k.
- ElasticNet regularization parameters: α and λ.
- Threshold for feature selection: θ.

Process:

1) Perform PCA on the dataset X to obtain the principal components.
    a) Compute the mean of X: $\mu = (1/n) * \Sigma(x_i)$
    b) Center the dataset by subtracting the mean:
    $$X_c = X - \mu$$
    c) Compute the covariance matrix of
    $$X_c\text{: }\Sigma = (1/n) * X_t^T * X_c$$
    d) Compute the eigenvectors and eigenvalues of Σ
    e) Arrange the eigenvectors by their eigenvalues, decreasing order.
    f) Identify the best k eigenvectors to use as PCs.
2) To reduce the dimensionality of the dataset X, we may use the k primary components to perform the transformation.
    a) Compute the projection matrix: $P = [v_1, v_2, ..., v_k]$, where v_i is the i^{th} eigenvector
    b) Transform the dataset: $X_t = X_c * P$
3) Apply ElasticNet to the transformed dataset X_t for feature selection.
    c) Initialize the coefficient vector $\beta = 0$
    a) Iterate until convergence:
    b) Update β using the ElasticNet optimization objective:
    β = argmin $(1/n)*||Y\text{-}X_t * \beta||^2 + \lambda*[(1\text{-}\alpha)/2*||\beta||^2+\alpha*||\beta||_1]$ (34)

where Y is the target variable vector

    c) Set coefficients smaller than a threshold θ to zero to enforce sparsity
4) Select the features corresponding to the non-zero coefficients in the final β as the selected features.

Output: Selected features subset.

1.7 CLASSIFICATION USING SVM

In the parameter space, the SVM is a nonlinear classifier because the mapping from the input pattern space to the high dimensional feature space is nonlinear. The optimization problem presented by SVM training is quadratic in nature. To optimize the distance between the hyper plane and the closest point, one must solve the quadratic optimization problem of establishing a hyper plane with the coefficients $w^T x + b = 0$. Here, w is the vector of hyper plane coefficients, and b is the bias factor. It turns out that with huge margins of separation, the only thing that matters is how close the points are to each other. The kernel function is used to calculate this kind of similarity. There is no universally accepted procedure for selecting an appropriate kernel function for a given situation.

$$D=\{(x^1,y^1),\ldots(x^1,y^1)\},xI\ R^n,\ yI\ \{1,-1\} \tag{35}$$

If the distance between the vectors nearest to the hyperplane is largest, then the separation produced by the hyperplane is best. A canonical hyperplane is a hyperplane with parameters w and b such that and only if these constraints hold,

$$\min_i |<w,x^{\mathrm{i}}> +b|=1 \tag{36}$$

Training errors may be kept to a minimum while profit maximization is still possible by adjusting the regularization value ‘C’. This is referred to as a “soft margin.” Therefore, a kernel function and a regularization parameter are needed to develop a support vector machine.

There are times when the SVM will not use a linear boundary, but instead will project the input vector x onto a high dimensional feature space z. The SVM may create an ideal hyper plane for classifying features in this higher-dimensional space if a non-linear mapping is used.

An inner product in feature space has an equivalent kernel in input space,

$$K(x,x')=<j(x),j(x')> \tag{37}$$

Non-linear modeling is where a polynomial mapping comes in,

$$K(x,x') = <x,x'>^d \tag{38}$$

where d is the polynomial degree. There has been a lot of focus on radial basis functions, often using a Gaussian of the type,

$$K(x,x) = \mathrm{e}^{-\frac{\|x-x'\|^2}{2\sigma^2}} \tag{39}$$

4. RESULTS AND DISCUSSION

After applying ensemble feature selection and SVM classification to the dataset, the findings and analysis are presented in the results and discussion section. The purpose of this part is to shed light on the efficiency and usefulness of the suggested method, as well as to address the implications and limits of the obtained findings.

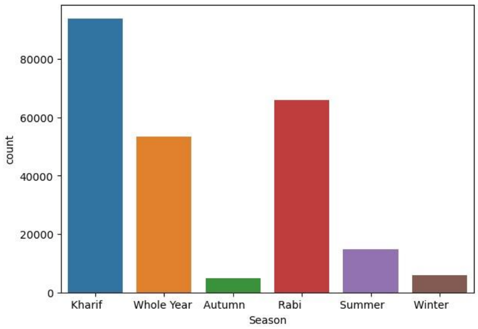

Fig.3. Plot Graph

The plot graph may show in Fig.3. The x-axis represents the season, while the y-axis displays the count X-axis: The x-axis represents the seasons of the year. It typically includes the four seasons: spring, summer, autumn (fall), and winter. Each season is usually depicted as a discrete point or label along the x-axis. Y-axis: The y-axis represents the count or quantity of something being measured or observed. The specific count being represented on the y-axis will depend on the context of your plot. Here are a few examples:

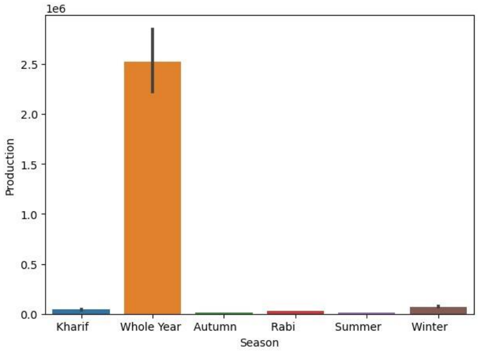

Fig.4. Production Analysis

A production analysis is shown in Fig.4. Season is shown along the x axis, while output is shown along the y axis. The Fig.4’s x-axis depicts each of the year’s four distinct seasons. Seasons including spring, summer, fall, and winter are often covered. Each season is often denoted by a discrete dot or label located along the x-axis. Y-axis: The output is shown along the y-axis in Fig.4.

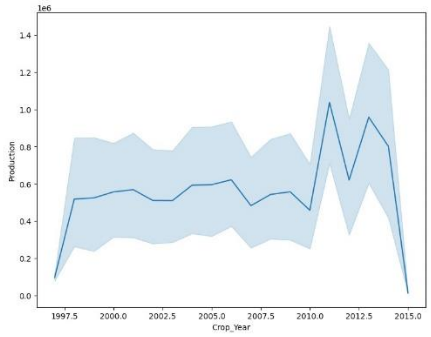

Fig.5. Crop Year-wise Production Analysis

The Fig.5 depicts a Crop Year-wise Production Analysis, with the X-axis representing the crop year and the Y-axis displaying the production levels. The goal of this plot is to study and comprehend production fluctuations over crop years.

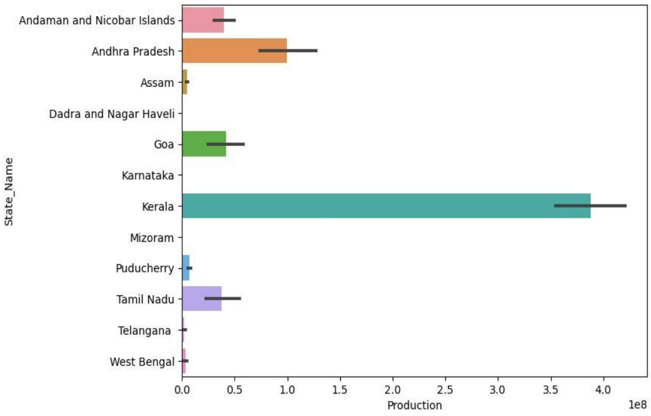

Fig.6. Feature Selection

The Fig.6 depicts feature selection, with the X-axis representing the production variable and the Y-axis displaying the names of distinct states. The goal of this graphic is to show the link between the production variable and the different states.

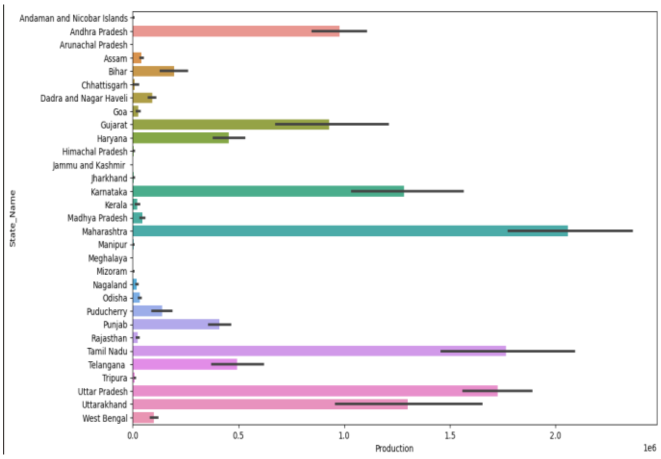

Fig.7. State-wise Analysis of Production

The Fig.7 depicts a State-wise Analysis of Production, with the X-axis representing the production variable and the Y-axis displaying the names of distinct states. The goal of this graphic is to evaluate and compare production levels among states.

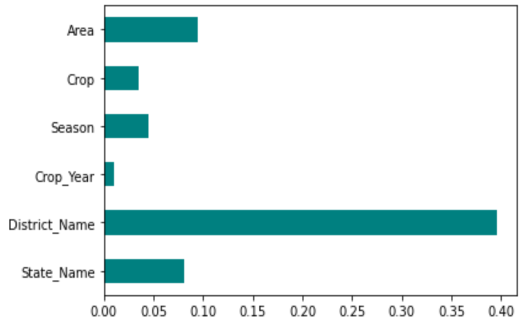

Fig.8. Important Feature selection

As can be seen in Fig.8, the best features have been chosen using the ensemble feature selection. The features and reviews as a whole. There has had a significant effect on the title. A global accuracy of 0.91 percent has been reached.

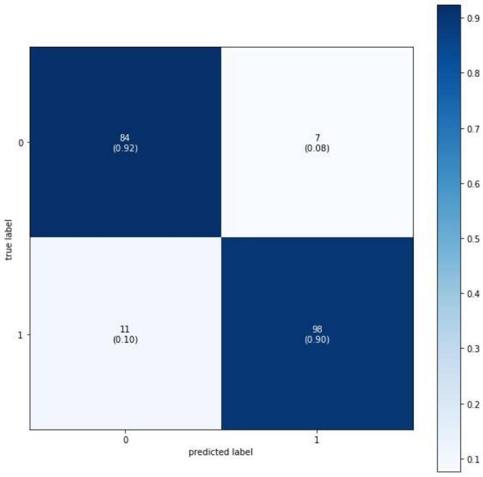

Fig.9. Confusion matrix

The Fig.9 shows confusion metrics TP, FP, TN, and FN values are represented in Fig.9. The predicted class for TP is 84, TN is 98, and FP and FN are 7 and 11.

Table.2. Performance Metrics Comparison

	K-means	Random forest	Proposed method
Accuracy	0.84	0.88	0.91
Precision	0.86	0.90	0.93
Recall	0.81	0.85	0.90
F-measure	0.83	0.87	0.92

The Table.2 reveals that the suggested technique outperformed Random Forest (0.88) and K-means (0.84), respectively. In terms of accurately categorizing examples, this demonstrates that the suggested technique excels above the other two methods. The suggested technique attained the maximum precision of 0.93, where precision quantifies the percentage of accurately predicted positive cases relative to all anticipated positive instances. The highest accuracy was achieved by Random Forest (0.9), while the lowest was achieved by K-means (0.86). As a result, it seems that the suggested strategy made more accurate positive predictions. The suggested strategy achieved a recall of 0.90, where recall is defined as the fraction of true positive events that were properly predicted. The recall for Random Forest was 0.85, whereas the recall for K-means was 0.81. Since the suggested technique has a better recall score, it is more likely to correctly identify true positives. The F-measure is a fair measurement of both accuracy and recall since it is the harmonic mean of the two. With an F-measure of 0.92, the suggested technique outperformed Random Forest (0.87), K-means (0.83), and everything else. These results show that the suggested strategy improved upon the trade-off between accuracy and recall. The suggested technique achieved higher levels of accuracy, precision, recall, and F-measure than both K-means and Random Forest. It showed improvements in classification accuracy, positive prediction accuracy, positive instance capture, and precision-to-recall ratio. These results demonstrate the superiority of the suggested strategy over the other two approaches when applied to classification tasks.

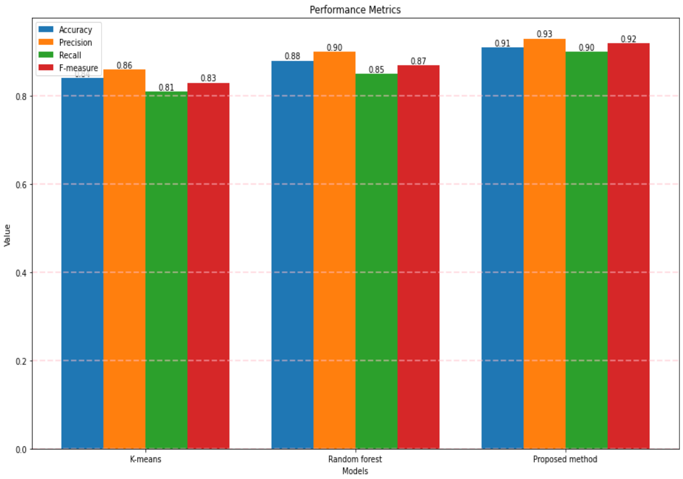

Fig.10. Performance Metrics Comparison

The Fig.10 shows performance metrics comparison the x axis shows models and y axis shows value.

5. CONCLUSIONS

In conclusion, this study proposes novel method on ensemble-based approach that combines weighted K-means clustering, average filling, Random Forest, Logistic Regression, Elasticnet, Principal Component Analysis, and Support Vector Machines for dataset normalization, feature selection, and classification. By employing these techniques, we aim to improve the performance and interpretability of machine learning models. The weighted K-means clustering and average filling techniques ensure a more accurate representation of the dataset by considering sample weights and handling missing values appropriately. This normalization step prepares the dataset for subsequent analysis. The ensemble feature selection approach, combining Random Forest, Logistic Regression, and Elasticnet, allows us to identify the most relevant features by considering their importance, relevance, and coefficients. This helps in reducing dimensionality and eliminating noise or redundant information. Principal Component Analysis further enhances the feature selection process by reducing the dimensionality of the dataset while preserving its important characteristics. Finally, SVM used for categorization.

REFERENCES

[1] O. Araque and C.A. Iglesias, “Enhancing Deep Learning Sentiment Analysis with Ensemble Techniques in Social Applications”, *Expert Systems with Applications*, Vol. 77, pp. 236-246, 2017.

[2] R. Banerjee and M. Singh, “Efficient Genomic Selection using Ensemble Learning and Ensemble Feature Reduction”, *Journal of Crop Science and Biotechnology*, Vol. 45, No. 1, pp. 1-112, 2020.

[3] S. Contiu and A. Groza, “Improving Remote Sensing Crop Classification by Argumentation-Based Conflict Resolution in Ensemble Learning”, *Expert Systems with Applications*, Vol. 64, pp. 269-286, 2016.

[4] H. Elghazel and A. Aussem, “Unsupervised Feature Selection with Ensemble Learning”, *Machine Learning*, Vol. 98, No. 1-2), pp. 157-180, 2013.

[5] N. Fayyazifar and N. Samadiani, “Parkinson’s Disease Detection using Ensemble Techniques and Genetic Algorithm”, *Proceedings of International Conference on Artificial Intelligence and Signal Processing*, pp. 1-4, 2017.

[6] D.P. Gaikwad and R.C. Thool, “Intrusion Detection System using Bagging Ensemble Method of Machine Learning”, *Proceedings of International Conference on Computing Communication Control and Automation*, pp. 1-6, 2015.

[7] I. Kaur and A. Kaur, “A Novel Four-Way Approach Designed with Ensemble Feature Selection for Code Smell Detection”, *IEEE Access*, Vol. 9, pp. 8695-8707, 2021.

[8] I.H. Laradji, M. Alshayeb and L. Ghouti, “Software Defect Prediction using Ensemble Learning on Selected Features”, *Information and Software Technology*, Vol. 58, pp. 388-402, 2015.

[9] A. Moghimi, C. Yang and P.M. Marchetto, “Ensemble Feature Selection for Plant Phenotyping: A Journey from Hyperspectral to Multispectral Imaging”, *IEEE Access*, Vol. 8, 1-13, 2018.

[10] B.T. Pham and I. Prakash, “Coupling RBF Neural Network with Ensemble Learning Techniques for Landslide Susceptibility Mapping”, *Catena*, Vol. 195, pp. 104805-104814, 2020.

[11] J.D. Prusa and A. Napolitano, “Using Feature Selection in Combination with Ensemble Learning Techniques to Improve Tweet Sentiment Classification Performance”, *Proceedings of International Conference on Tools with Artificial Intelligence*, pp. 1-8, 2015.

[12] A. Rai, “Optimizing a New Intrusion Detection System Using Ensemble Methods and Deep Neural Network”, *Proceedings of International Conference on Trends in Electronics and Informatics*, pp. 1-5, 2020.

[13] A. Safiyari and R. Javidan, “Predicting Lung Cancer Survivability using Ensemble Learning Methods”, *Proceedings of International Conference on Tools with Artificial Intelligence*, pp. 1-6, 2017.

[14] V. Shorewala, “Early Detection of Coronary Heart Disease using Ensemble Techniques”, *Informatics in Medicine Unlocked*, Vol. 67, No. 2, pp. 100655-100665, 2021.

[15] S. Tajik, S. Ayoubi and M. Zeraatpisheh, “Digital Mapping of Soil Organic Carbon using Ensemble Learning Model in Mollisols of Hyrcanian Forests, Northern Iran”, *Proceedings of International Conference on Geoderma Regional*, pp. 1-12, 2020.

[16] M. Tan and F. He, “Ultra-Short-Term Industrial Power Demand Forecasting using LSTM based Hybrid Ensemble Learning”, *IEEE Transactions on Power Systems*, Vol. 88, 1-9, 2020.

[17] Z. Tang, W. Cai and C. Han, “An Object-Based Approach for Mapping Crop Coverage using Multiscale Weighted and Machine Learning Methods”, *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, Vol. 76, No. 2, pp. 1-14, 2020.

[18] A.K. Verma, S. Pal and S. Kumar, “Comparison of Skin Disease Prediction by Feature Selection using Ensemble

Data Mining Techniques”, *Informatics in Medicine Unlocked*, Vol. 65, No. 2, 100202-100209, 2019.

[19] B. Weng and W. Martinez, “Predicting Short-Term Stock Prices using Ensemble Methods and Online Data Sources”, *Expert Systems with Applications*, Vol. 112, pp. 258-273, 2018.

[20] I. Yekkala, S. Dixit and M.A. Jabbar, “Prediction of Heart Disease using Ensemble Learning and Particle Swarm Optimization”, *Proceedings of International Conference on Smart Technologies for Smart Nation*, pp. 1-4, 2017.

[21] Y. Zheng, Y. Li and B. Wei, “Feature Selection with Ensemble Learning Based on Improved Dempster-Shafer Evidence Fusion”, *IEEE Access*, Vol. 8, 1-9, 2019.

[22] J.P. Bharadiya and M. Reddy, “Forecasting of Crop Yield using Remote Sensing Data, Agrarian Factors and Machine Learning Approaches”, *Journal of Engineering Research and Reports*, Vol. 24, No. 12, pp. 29-44, 2023.

[23] Z. Zhou, Z. Wu and Y. Qiao, “Comparison of Ensemble Strategies in Online NIR for Monitoring the Extraction Process of Pericarpium Citri Reticulatae Based on Different Variable Selections”, *Planta Medica*, Vol. 82, No. 1-2, pp. 154-162, 2015.

[24] A. Oikonomidis and A. Kassahun, “Hybrid Deep Learning-Based Models for Crop Yield Prediction”, *Applied Artificial Intelligence*, Vol. 36, No. 1, pp. 2031822-2031829, 2022.

[25] B. Panigrahi and M. Sujatha, “A Machine Learning-Based Comparative Approach to Predict the Crop Yield using Supervised Learning with Regression Models”, *Procedia Computer Science*, Vol. 218, pp. 2684-2693, 2023.

[26] M. Kuradusenge, K. Mtonga, A. Mukasine and A. Uwamahoro, “Crop Yield Prediction using Machine Learning Models: Case of Irish Potato and Maize”, *Agriculture*, Vol. 13, No. 1, pp. 225-237, 2023.

[27] S.S. Olofintuyi and D. Olanike, “An Ensemble Deep Learning Approach for Predicting Cocoa Yield”, *Heliyon*, Vol. 9, No. 4, pp. 1-13, 2023.

[28] H.R. Seireg and A. Elmahalawy, “Ensemble Machine Learning Techniques using Computer Simulation Data for Wild Blueberry Yield Prediction”, *IEEE Access*, Vol. 10, pp. 64671-64687, 2022.

[29] H.T. Pham and M. Kuhn, “Evaluation of Three Feature Dimension Reduction Techniques for Machine Learning-Based Crop Yield Prediction Models”, *Sensors*, Vol. 22, No. 17, pp. 6609-6615, 2022.

[30] R. Aworka, F.K. Mutombo, C.L.M. Kimpolo and M. Krichen, “Agricultural Decision System based on Advanced Machine Learning Models for Yield Prediction: Case of East African Countries”, *Smart Agricultural Technology*, Vol. 2, pp. 1-13, 2022.