



Machine Learning and XAI approaches for Allergy Diagnosis

Ramisetty Kavya^a, Jabez Christopher^{a,*}, Subhrakanta Panda^a, Y. Bakthasingh Lazarus^b

^a Department of Computer Science and Information Systems, BITS Pilani, Hyderabad Campus, Telangana State, India

^b Joyce Clinical Labs, Marthandam, Tamil Nadu, India



ARTICLE INFO

Keywords:

Ensemble classification
Post-hoc explainability
Clinical decision-making
Allergy diagnosis
Healthcare mobile application

ABSTRACT

This work presents a computer-aided framework for allergy diagnosis which is capable of handling comorbidities. The system was developed using datasets collected from allergy testing centers in South India. Intradermal skin test results of 878 patients were recorded and it was observed that the data contained very few samples for comorbid conditions. Modified data sampling techniques were applied to handle this data imbalance for improving the efficiency of the learning algorithms. The algorithms were cross-validated to choose the optimal trained model for multi-label classification. The transparency of the machine learning models was ensured using post-hoc explainable artificial intelligence approaches. The system was tested by verifying the performance of a trained random forest model on the test data. The training and validation accuracy rate of the decision tree, support vector machine and random forest are 81.62, 81.04 and 83.07 respectively. During evaluation, random forest achieved a rate of 86.39 accuracy overall, and 75% sensitivity for the comorbid Rhinitis-Urticaria class. The framework along with all the functionalities were deployed on mobile devices. The average performance of the clinicians before and after using the decision support system were 77.21% and 81.80% respectively. The diagnosis system integrated with mobile applications serves as a source of information whereby junior clinicians can use it to confirm their diagnostic predictions.

1. Introduction

Clinical diagnosis is generally based on the signs and symptoms of a disease [1]. Early diagnosis based on the common signs and symptoms is quite difficult and a challenging task because there are several symptoms which are common to more than one disease. Delay in diagnosis has significant impact on the severity of the symptoms which leads to life-threatening diseases. Allergies are one among the diseases that are difficult to diagnose in its early stages because they share common symptoms with several other medical disorders.

Allergy is an inflammatory disease that results in hypersensitive reactions in the immune system when allergy-causing substances present in the environment either enters or comes in direct contact with the body. The allergy-causing substances known as allergens are not harmful, but trigger the immune system to react adversely [2]. Various allergic diseases like allergic rhinitis, asthma, urticaria, atopic dermatitis, and many others are present in many parts of the world. The type of allergy, the signs and symptoms, and their manifestation differ geographically. More than 25% of the Indian population [3] and nearly 30% of the global population [4] are suffering from various allergic

diseases [5,6]. Due to rapid urbanization and change in life style, an exponential rise in allergic diseases over the next two decades is expected.

Allergic rhinitis is an airway allergic disease, which involves the entire respiratory tract [7]. Though it is not a life-threatening disease; it is often under-diagnosed which leads to complicated problems like asthma and sinusitis. Severe allergic rhinitis mainly affects children's mental health, learning at schools, and quality of life [8]. Urticaria is a commonly occurring skin allergy with a broad spectrum of symptoms, including angioedema or the sudden development of wheals on the skin [9]. It has a significant impact on patients' physical appearance and social life, and affects daily activities, due to constant itching.

Allergic diseases can be diagnosed based on physical examination, clinical history, and knowledge about local food habits and environmental conditions. If the presence of allergic symptoms is detected, then sensitization can be confirmed by performing either *in vivo* tests (intradermal or epicutaneous skin test) or *in vitro* assays (immunoglobulin E antibody serology, basophil activation or mediator release) [10]. The type of allergy test to be performed is chosen based on patient-history, suspected allergen triggers, and operational issues like cost, time, and

* Corresponding author.

E-mail address: jabezc@hyderabad.bits-pilani.ac.in (J. Christopher).

risk associated. Generally, intradermal skin test is considered as a gold standard for diagnosing the presence and the cause of allergy [10]. But the vagueness and uncertainty involved in the interpretation of the test's final outcomes makes it less desirable. It is a difficult task for general physicians and junior clinicians to analyze and report the final outcomes of the intradermal skin test in the absence of an immunologist. As of now, the decision of the domain expert (Immunologist) is considered as the gold standard for reporting the final test result; but there is a dearth of allergy specialists for finalizing the test outcomes in several areas. Moreover, it is practically difficult to ensure consistency in administration and reporting of the allergy tests across several medical clinics.

Sometimes patients visit health care centers with comorbidities. Computer-aided systems, and aggregated online medical data or literature resources, are unavailable for junior clinicians to refer during decision-making. It may not be possible to develop a unique clinical support system for all diseases because symptoms, laboratory tests, testing modalities, and equipment calibrations may vary from disease to disease. However, it is possible to combine a few diseases together; clinicians, from their experience and knowledge from medical records, reveal some correlations and relationships between the diseases. For instance, a patient suffering from allergic rhinitis is prone to be affected with asthma [11]. Instead of providing two distinct support systems for allergy rhinitis and asthma, it is desirable to combine and provide a single support system to immunologists. It is advisable to combine those diseases that are treatable by a medical expert.

Computer-aided systems aim at assisting decision-makers in decision-making tasks. Different forms of computer-aided systems for enhancing clinical diagnosis and decision support are reported in literature; these include Clinical Decision Support Systems (CDSS) [12], Diagnosis Support Systems [13], Automated Medical Diagnostics [14], Healthcare Information System [15], Electronic Health Record System [16], Intelligent Healthcare Systems [17], Knowledge-based Systems [18], Medical Expert Systems [19], Evidence-based Systems [20], and many others. These systems aid physicians in providing safe and reliable medical services in various domains such as Cardiology [21], Pulmonology [22], Respirology [23], Hepatology [24], Nephrology [25], Dentistry [26], Ophthalmology [26], Dermatology [27] and many others. Junior clinicians can utilize the outcomes and decisions to support their medication advises in the absence of domain experts. With the appropriate use of CDSS, clinicians can avoid misinterpretation of test outcomes and making incorrect diagnostic decisions that are likely to happen due to negligence, observational errors or sometimes even lack of exposure and domain knowledge. Moreover, the use of CDSS in clinical decision-making tasks enhances the quality of medical services provided.

Clinical Decision Support System (CDSS) developed based on emerging approaches like ML and AI supports to arrive at quick and better decisions. ML approaches identify patterns in complex datasets for generating appropriate predictions [28]. The dataset is divided into train and test; the former is used to train and validate the ML models whereas the latter is used to evaluate the performance of the model. Learning models are categorised into supervised learning and unsupervised learning. The goal of supervised learning is to learn the mapping from input features to the associated output class labels whereas unsupervised learning groups the samples that have similar features. Supervised learning is further divided into classification and regression based on the domain of the outcome-feature (class label). Classification is the task of predicting the class labels of unknown samples by the use of classifiers like k -nearest neighbors [29], support vector machine [30], decision tree [31], multi-layer perceptron [32] and ensemble approaches [33]. The selection of an optimal classifier from these many classifiers to generate appropriate predictions from given data require validation and performance evaluation. This ensures that the classifiers that are used by CDSS provides reliable results and outcomes.

In general, few classification models are considered to be black-box; but the clinicians often require a clear understanding of its' outcomes.

They desire to know the justification of why the system had arrived with such a decision. The need for transparency gradually increases with the increase in the danger of using predictions that are not justifiable [34]. But there is a trade-off between model performance and transparency. In order to avoid this trade-off, eXplainable Artificial Intelligence (XAI) concepts are infused. These concepts ensure that the CDSS provides supportive reasoning and explainable medical knowledge for all the predictions by a classifier. Moreover, this knowledge and reasoning is easy to be understood and to be interpreted by the clinicians. Hence, this work focuses on developing an Allergy Diagnosis Support System (ADSS) using machine learning models, whereby, it is feasible to generate interpretable explanations by maintaining a high level of model performance [34]. The generated explanations would be in a human-understandable format and will increase the trust to believe the predictions given by the support system. In general, IF-THEN rules are used to represent the explanations extracted from medical data to support predicted disease. Since the ADSS is built upon medical data and patient's demographic information, and then validated by medical experts, junior clinicians can trust the predictions and supportive rules provided by the ADSS. Moreover, the use of self-explanatory IF-THEN rules to support multiple disease combinations, may increase the usage of these systems by physicians in diagnostic decision-making tasks.

This work focuses on two aspects which are as follows:

- First, development of CDSS to support diagnosis of multiple combinations of allergic disorders. The focus is to provide understandable human-reasoning behind the predictions given by CDSS. This reasoning will increase the physician's trust on the support systems. Immunologists can now resort to one unified platform for allergy diagnosis instead of referring to multiple disease specific systems.
- The efficiency of CDSS lies in the quality of data; hence the second aspect is about enhancing data quality by handling the class imbalance problem. A dataset is said to be imbalanced if there is at least one class with fewer samples compared to other classes. Class imbalance makes learning algorithms to be biased towards classes with a higher number of samples. Data sampling techniques are used for addressing the class imbalance problem. The work incorporates modifications to existing sampling algorithms to handle class imbalance problems by considering user preferences.

The remainder of this paper is as follows: Section 2 presents related works in existing literature, mainly about allergy diagnosis and prognosis, and clinical decision support systems. Section 3 presents the allergy diagnosis support system framework. Section 4 gives details about the experimental settings, and discusses the results and threads to validity. Section 5 specifies the use of technology in integrating the CDSS with mobile platform. Finally, the conclusion section shreds light on the scope and future directions of this work.

2. Related works

The related works presented in this section are based on two aspects: Allergy Diagnosis and Prognosis, and Clinical Decision Support Systems (CDSS). The former focuses on the various kinds of allergy treatments, including antihistamines and allergy immunotherapy that are frequently adopted by clinicians to diagnose allergic diseases. It also highlights the challenges faced by clinicians who diagnose allergy with traditional approaches and thereby the need for computerized systems to support their decision-making in allergy diagnosis. The latter focuses on the computerized systems like CDSS that are developed to support clinicians while treating allergic patients.

2.1. Allergy Diagnosis and Prognosis

Recto et al. conducted a survey among specialists from dermatology, otorhinolaryngology, and allergology about diagnostic decision factors

and practical considerations while treating allergic rhinitis and urticaria patients with second-generation antihistamines [5]. The survey highlights the need for online tools to diagnose allergic diseases, and personalize the treatment by recommending a suitable antihistamine according to the patient's medical records. Demoly and a group of French allergists [35] have provided a series of online tools, tables that contain common allergy symptoms, and detailed allergy case reports to support general practitioners and house surgeons. Their work not only adds to allergy-literature, but the tools and tables support general physicians to treat mild and moderate allergy which in turn reduces the burden of immunologists.

Hellings et al. summarize the challenges and also provide strategies to improve the care for allergic rhinitis patients by adopting allergy immunotherapy in daily life [36]. The national-wide survey conducted among Ear–Nose–Throat (ENT) specialists revealed the two main reasons for not prescribing immunotherapy: high-cost burdens and scarcity in expert health care professionals. In such scenarios, digital tools like decision support systems may support health care professionals in arriving at decisions about adopting immunotherapy for allergy treatment.

Merging the modality of telemedicine-diagnosis with CDSS, significantly improves the facilitation of diagnosis along with quality for allergy patients. Integrating CDSS, with other relevant and emerging technologies like remote patient monitoring, wearable smart devices, electronic diaries, telemedicine enables the process of diagnosis to overcome the barriers like time and distance. Efficiency of telemedicine in treating allergy and asthma is comparable to personal out-patient visits to a health center [37]. It is convenient for in-patients to have regular appointments with immunologists who offer treatment in different hospitals. But the usage of telemedicine for diagnosis of allergy, helps in addressing the issue of availability of immunologists in rural areas and also the issue of difficulties in getting an immunologist's appointment in urban areas.

2.2. Clinical Decision Support Systems

Bousquet et al. developed a CDSS based on Mobile Airways Sentinel Network (MASK) algorithm which benefits both patients and healthcare professionals for receiving appropriate treatment and in determining the step-up or step-down treatment strategy for controlling allergic rhinitis [38]. Large-scale testing trials are required to explore whether the step-up or step-down treatment recommendations provided by CDSS are practically beneficial or not.

The control of variations in allergic rhinitis symptoms depends on environmental exposure, allergic rhinitis phenotype, and current medication procedure. Visual Analogue Scale (VAS) is used to measure the control of allergic rhinitis [39]. It is embedded into Allergy Diary mobile app for patients to access and self-manage their allergy condition. This application also supports in determining the step-up and step-down allergic treatment strategies based on VAS score, allergen exposure, allergic rhinitis phenotype, and patient history.

Chae et al. have developed a medical decision support system for asthma diagnosis [40]. To find the efficiency of the decision support system, authors conducted a comparative study on three supervised machine learning models, namely, Neural Networks (NN), Case-based Reasoning (CBR), and discriminant analysis. NN achieved 92% accuracy; an expert system tool, ART-IM (Automated Reasoning Tool for Information Management), used for implementing CBR achieved an accuracy of 84%. A list of 14 opinion poll questions, lab tests, and skin prick tests are used for implementing the discriminant analysis model. It achieved an accuracy of 90%. A similar framework of NN, CBR, and linear discriminant is used for allergic rhinitis prediction, which achieved 76%, 62%, and 78% accuracy respectively.

Christopher et al. have worked on developing a decision support systems for predicting allergic disorders in southern states of India (Kerala and Tamilnadu) [41,42]. Intradermal skin tests were performed

on patients who had plausible allergic symptoms. Based on the patient's history, a customized list of 40 clinically relevant allergens were tested on 872 patients who had allergic symptoms. Supervised machine learning approaches, which included a rule-based classification approach, were used to develop and validate the CDSS. Clinical relevance of the CDSS was ensured by local immunologists and also by comparing with the Score for Allergic Rhinitis (SFAR). The junior clinicians were able to classify the test samples with an average accuracy of 75.68%. Their works focused on developing separate and specific systems for each allergy, namely rhinitis and urticaria.

To the best of our knowledge based on literature, all the available computer-aided systems support in diagnosing specific allergic disorders. They do not provide any information related to the other possible allergies. But in the real world there are cases where more than one allergy was reported in the same patient. Even though systems developed to support specific disease achieves high accuracy, physicians are not willing to use multiple systems to treat different patients. Patients approach clinicians for different problems, and it is difficult for clinicians to switch among multiple systems to treat comorbidities. Hence there is a need for developing a standalone system for physicians to treat various combinations of allergies like rhinitis, asthma, urticaria, atomic dermatitis, eczema and other possible comorbidities.

This paper focuses on developing a CDSS which is capable of diagnosing coexisting allergic disorders. The proposed allergy diagnosis support system fulfills the physician's requirement of having one unified platform that can diagnose multiple allergies. Moreover, the proposed system provides reasoning behind the predictions made by machine learning algorithms with the support of XAI approaches. This enables physicians to provide trust-worthy, evidence-based reasoning for their decisions. The following section discusses the framework of the proposed clinical decision support system.

3. System framework

The framework of the proposed Clinical Decision Support System (CDSS) for the diagnosis of multiple coexisting allergies is presented in Fig. 1. Data of allergy patients, including the results of the intradermal skin test, are the inputs to the system. The pre-processor, validator, and

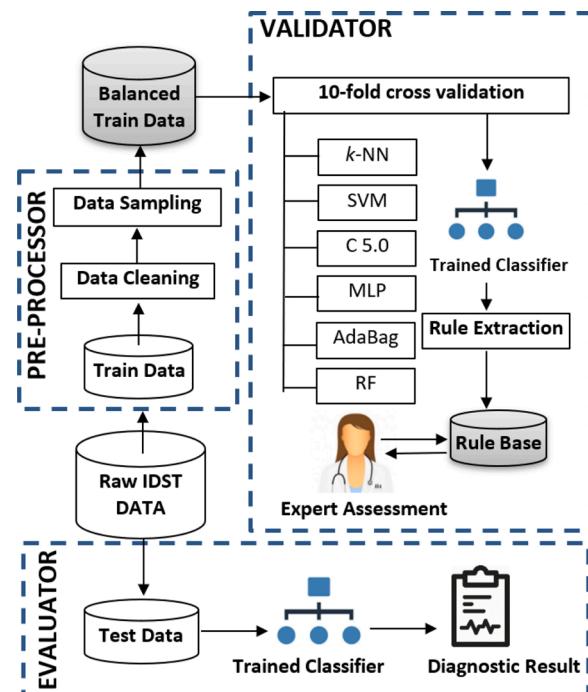


Fig. 1. Framework for allergy diagnosis support system.

evaluator are the three major modules in the allergy diagnosis support system. This section begins with a brief description of all the necessary mathematical representations which supports in understanding the modules of the system.

3.1. Mathematical representations

Let $D \ni \{D_{train}, D_{test}\}$ represents a set of datasets where each element is composed of rows and columns. Both the elements in D contains k distinct attributes A_1, A_2, \dots, A_k as columns and each attribute has a specific domain $A_1 = \{a_{11}, a_{12}, \dots, a_{1p}\}$, $A_2 = \{a_{21}, a_{22}, \dots, a_{2p}\}$, $A_k = \{a_{k1}, a_{k2}, \dots, a_{kp}\}$ where $p \in \mathbb{N}$. An element from a set of class labels represented as $C = \{c_1, c_2, \dots, c_m\}$ is associated with each row in D_{train} . The cardinality of C is represented as m and the cardinality of D_{train} is represented as n . Moreover, the cardinality of a class in D_{train} is represented as n_{ci} . Finally, each instance (row) in D_{train} looks like $\langle A_1 = \{a_{1,1} \dots p\}, A_2 = \{a_{2,1} \dots p\}, \dots, A_k = \{a_{k,1} \dots p\}, C = \{c_{1 \dots m}\}$ whereas an instance from D_{test} looks like $\langle A_1 = \{a_{1,1} \dots p\}, A_2 = \{a_{2,1} \dots p\}, \dots, A_k = \{a_{k,1} \dots p\} \rangle$. All the instances in D_{train} are processed and forwarded to train and validate a set of classification models. A set of classification rules, $R \ni \{r_1, r_2, \dots, r_q\}$ were extracted from the chosen trained classifier. Each rule (r_i) is of the form $(r_i): (Cond \rightarrow Pred)$ where *Cond* (Condition) is a conjunction of attribute-value pairs, represented as $A_1 = \{a_{1,1} \dots p\} \wedge A_2 = \{a_{2,1} \dots p\} \wedge \dots \wedge A_k = \{a_{k,1} \dots p\}$ and *Pred* (Prediction) is the one among m class labels (c_i).

3.2. Data Description

Data of 878 patients were obtained from the Good Samaritan Allergy Testing Centre, Kilpauk, Chennai. The clinical history, physical symptoms and demographic information of 878 patients who visited the clinic are considered to be raw data (D_{raw}). Table 1 presents the complete list of the 91 features in D_{raw} . The distinct features in D_{raw} can be segregated into three disjoint groups namely allergens, symptoms, and personal information. Each group addresses different aspects related to the environmental surroundings, food habits and personal demographics.

Table 1
List of features in D_{raw} .

<i>Allergens (inhalants)</i>				
House dust	Cotton dust	Aspergilus	Road dust	Old paper dust
PS dust				
<i>Allergens (contactants)</i>				
Pollen	Parthenium	Cockroach	Cat dander	Dog fur
<i>Allergens (ingestants)</i>				
Milk (P)	Milk (C)	Curd	Coffee	Tea
Fish1	Fish2	Chicken	Mutton	Egg
Beef	Shark	Crab	Prawns	Gourds
Cabbage	Banana	Beans	Beet root	Brinjal
Radish	Capiscum	Chillie	Cauliflower	Carrot
Broad beans	Corn	Cucumber	Drumstick	Greens
Mango	Coccinia grandis	Cluster beans	Lady's finger	Coriander
Pot root	Mushroom	Brassica oleracea	Onion	Pees
Chayota edulis	Farmer cheese	Potato	Pumpkin	Mentha spicata
Gram	Tomato	Trichosanthes dioica	Plantain stem	Yams
Ragi	Channa	Dhal	Maida	Oats
Garlic	Rice	Wheat	Coconut	Oil
Spices	Ginger	Pepper	Tamarind	Aginomoto
Coco	Horlicks	Boost	Nuts	
<i>Physical symptoms</i>				
Running Nose	Itching	Sneeze	Cough	Wheeze
Headache	Swelling		Red rashes	
<i>Personal information</i>				
Age	Gender		Family history	

- The allergens group contains 80 allergy-causing agents. These are sub-divided into three groups namely inhalants, contactants and ingestants. Inhalants are breathed, ingestants are swallowed and contactants are those that come in contact with skin. The values for these allergens are the observations of the intradermal skin test administered by medical professionals.
- The symptoms group contains physical symptoms that are manifested by the patient. Details about the symptoms are interviewed from patients who visit an allergy clinic. The values for these allergy symptoms are recorded as nominal where *yes* represents the presence of a symptom, and *no* represents the absence of symptoms.
- The personal information group contains demographic and other miscellaneous information.

Holdout approach [43] is applied for splitting D_{raw} into two disjoint sets of 8 : 2 proportion; the train dataset (D_{train}) contains 80% of data from D_{raw} and the test dataset (D_{test}) contains 20% of data from D_{raw} . D_{train} is the input data for the pre-processor module; the pre-processor enhances the quality of the data in order to improve the efficiency of the succeeding modules.

3.3. Pre-processor module

Better data leads to better decisions. The quality of D_{train} is the foremost aspect that influences the performance of the allergy diagnosis support system. The presence of insufficient information, missing values, redundant or irrelevant information will degrade the data quality which thereby increases the difficulty level in discovering useful patterns in the validator module. Though data processing takes a considerable amount of time, including this module will improve the overall system performance [44]. Data pre-processor module begins with data cleaning followed by data sampling.

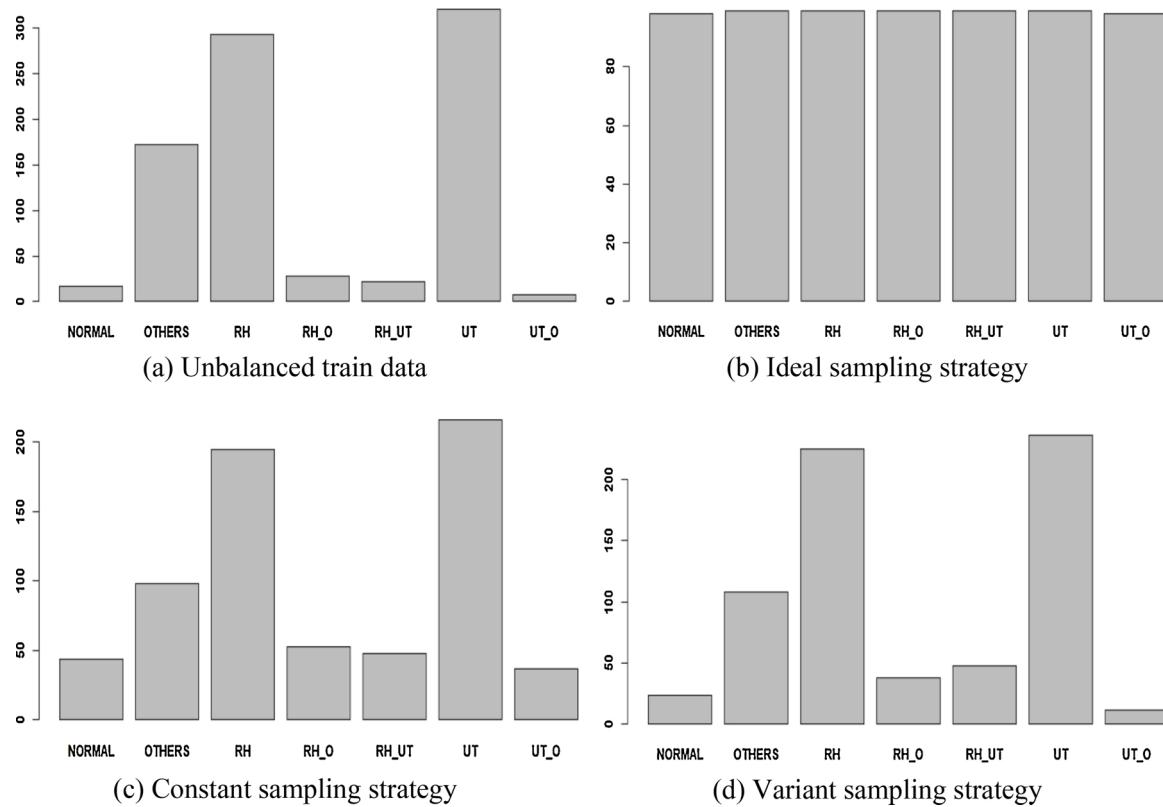
3.3.1. Data cleaning

While dealing with real-time data sources, there is a high chance of missing some attribute-values. Missing values are broadly categorized into three types which are as follows: Missing Completely at Random (MCAR), when there is no relationship between the missing values and the observed values; Missing at Random (MAR), when there is a systematic relationship between the missing values and the observed values; and finally Missing not at Random (MNAR), when the values are not recorded into the dataset intentionally [45].

There are several ways to handle the missing attribute-values based on the reason behind missing those values. In this work, the immunologist and the clinicians were involved in the data cleaning process which includes handling missing data. The reasons behind the data found to be missing were analyzed and appropriate data were imputed based on their experience and context. However, if more than 18–20 values are missing for a patient's record then that record was discarded according to the guidelines of the expert immunologist. After removing samples (records) with missing values there were 691 samples in D_{train} and 169 samples in D_{test} .

3.3.2. Data sampling

After data cleaning, it was observed that the samples present in the D_{train} were not evenly distributed among all the 7 classes; this can be visualized from Fig. 2. Often medical datasets have a large number of samples for some classes, whereas remaining classes have comparatively few samples. Classes with a large number of samples are named as majority classes whereas the remaining classes with smaller number of samples are named as minority classes, and the data is said to be imbalanced [46]. High differences among samples belonging to majority and minority classes, forces the learning algorithms to be more biased towards the majority classes by ignoring the minority classes. It is desirable that the learning algorithms are more generalized towards all the m classes in D_{train} . This generalization can be achieved by resolving

Fig. 2. Distribution of D_{train} .

the class imbalance problem. Many studies in machine learning discuss various approaches to handle the class imbalance problem [47]. The most frequently used approach is data sampling. The two data sampling techniques are over-sampling and under-sampling; the former generates synthetic data which are almost similar to the available instances for increasing the number of instances belonging to minority classes, whereas latter removes some samples from majority classes at random.

Consider a dataset with instances belonging to more than one class; a class with less than desired number of samples (n_s) is said to be a minority. But for another dataset, the same number (n_s) may not be intuitive for a class to be considered as a minority.

Example

If any class in the dataset is identified with less than desired number of samples ($n_s = 25$) then that dataset is said to be imbalanced.

Case I: Consider a hypothetical dataset with 100 samples belonging to 2 classes; positive class and negative class. $n = 100$, $m = 2$ (refer 3.1). The ideal number of samples ($Ideal_s$) in any class in the dataset is supposed to be equal across all classes; $Ideal_s = \frac{n}{m}$. Hence $Ideal_s = 50$. Suppose if the negative class contains less than 25 samples then we can consider this dataset to be imbalanced.

Case II: Consider a hypothetical dataset with 100 samples belonging to 5 classes (c_1, c_2, \dots, c_5). $n = 100$, $m = 5$, $Ideal_s = \frac{n}{m} = 20$. Suppose if c_1 contains 24 samples then we cannot consider this dataset to be imbalanced. Though the number of samples in c_1 is less than n_s , it is greater than the ideal number of samples that is supposed to be in that class ($|c_1| > Ideal_s$). In this case, if n_s is equal to or less than 10 samples for a class then it is reasonable to consider that dataset to be imbalanced.

Hence, a universal constant value for n_s is not applicable to all the datasets in order to consider a class as a minority. For a class to be known as a minority class, the number of samples that needs to be there in that class is dependent on the data distribution, user requirements and domain specifications. However, it is desirable that the user of the data decide the acceptable difference between the number of samples in the majority and minority class, and this thereby reflects on the amount of

sampling that has to be done in order to alleviate class imbalance.

In this work a commonly used sampling algorithm is modified by incorporating an intuitive approach to identify the existence of class imbalance and thereby resolve it by using over-sampling and under-sampling. The proposed approach purely depends on the total number of samples (n) in the dataset, the number of classes (m) and one user-defined parameter (η). Users can give the value ranging from 0 to 1 for (η) based on domain relevance and requirements. η is proportional to n_s which is used to identify whether the class is minority or majority and thereby used in the calculation of amount of sampling ($Cperc$); Desired number of samples (n_s) is computed as follows:

$$n_s(c_i) = \frac{n}{m} - \frac{n-m}{m(\eta_{c_i} + 1)} \quad (1)$$

$$Cperc_{c_i} = \frac{n_s(c_i)}{n_{c_i}} \quad (2)$$

If the user has given $\eta = 0$, then the resultant n_s value is 1, which means there should be at most one instance in order to consider a class as a minority. The existence of a class is not considered if there are no instances belonging to that class. If the user has given $\eta = 1$ then the resultant value for n_s is approximately equal to the 50% of ideal value ($Ideal_s$). It is reasonable to restrict the η value between 0 and 1 because $\eta \geq 0$ implies that there should be at least one instance for each class in a dataset whereas $\eta \leq 1$ implies that the class is with samples less than 50% of $Ideal_s$.

Eqs. (1) and (2) are used to identify whether the class is minority or majority along with amount of sampling that has to be done for each class in D_{train} . There were 691 samples belonging to 7 distinct classes in D_{train} ; $n = 691$, $m = 7$. The number of samples in each class ($|c_i| = n_{c_i}$) is shown in Table 2. Substitute both $n = 691$ and $m = 7$ values in Eq. (1); thereby vary η value to calculate n_s for each class. Table 3 presents the η value along with corresponding n_s value. The class is said to be minority when n_{c_i} value is less than $n_s(c_i)$, otherwise, majority. The n_s values

Table 2List of classes along with number of samples in D_{train} .

Class index	c_i label	$ c_i = n_{c_i}$
c_1	Normal	14
c_2	Others	138
c_3	Rhinitis (RH)	235
c_4	Rhinitis and Others (RH_O)	23
c_5	Rhinitis and Urticaria (RH_UT)	18
c_6	Urticaria (UT)	256
c_7	Urticaria and Others (UT_O)	7
$ C = 7$		$\sum_{i=1}^m n_{c_i} = 691$

Table 3Resultant minority classes after calculating n_s by varying η value.

η	n_s value	Number of minority classes	Class indices
0.16	14	2	c_1, c_7
0.21	17	2	c_1, c_7
0.22	18	3	c_1, c_5, c_7
0.29	22	3	c_1, c_5, c_7
0.3	23	4	c_1, c_4, c_5, c_7
1	49	4	c_1, c_4, c_5, c_7

present in **Table 3** are compared with n_{c_i} value present in **Table 2** to identify the resultant number of minority classes in D_{train} .

As said earlier, if at least one single minority class is identified after calculating n_s value correspondence to η , then the dataset is said to be imbalanced. By analyzing the number of minority classes presented in **Table 3**, there is no change in the resultant minority classes after reaching particular η value. Hence, the resultant number of minority classes for D_{train} is four by ranging the η value between 0.3 and 1; thereby concluded D_{train} as an imbalanced dataset.

The optimal way to overcome the learning bias caused by class imbalance problem is utilising sampling algorithms for providing a balanced train dataset to the learning algorithms. Though many approaches are available in the literature for sampling [48], Synthetic Minority Over-sampling Technique (SMOTE) is the widely used well-known algorithms for over-sampling and under-sampling simultaneously [49]. Hence, SMOTE is used to over-sample the four minority classes and under-sample the remaining three majority classes to make the D_{train} as balanced data. The pseudo-code of SMOTE algorithm for multi-class along with proposed modifications is shown in Algorithm 1: η -SMOTE.

η -SMOTE: Synthetic Minority Oversampling Technique – used to create synthetic samples – is better than re-sampling [49]. Both over-sampling and under-sampling is applied until the difference between the samples belonging to majority and minority classes arrive at desired percentages. In this work, three distinct strategies namely, ideal, constant and variant are considered to calculate the amount of sampling ($Cperc$) that is to be applied on D_{train} . First, in ideal strategy, equal distribution of samples to all the classes in a dataset is maintained ($n_s = \frac{n}{m}$). Second, in constant strategy, if the user has given the η value for minority class then the calculated n_s value using Eq. (1) is applied to minority classes for over-sampling, whereas n_s value obtained by substituting $1 - \eta$ value is for under-sampling the majority samples. Third, in variant strategy, the user can give the separate η value to each class present in the dataset based on domain-requirements and priorities. n_s value for each class(c_i) has to be calculated by substituting the corresponding η value. Once after calculating the $n_s(c_i)$ values according to the followed strategy, subtract n_s value from n_{c_i} ; if the resultant value is negative then perform over-sampling else perform under-sampling. The class distribution of D_{train} after applying three distinct strategies

can be visualized from Fig. 2.

Algorithm 1: η -SMOTE

```

Process: Synthetic Minority Over Sampling Technique
Input: Train data with  $n$  samples and  $m$  classes ( $c_i$ )
          Sampling strategy (stgy) :{Constant,Variant}
          User defined parameter ( $\eta$ ) :  $0 \leq \eta \leq 1$ 
Output: Balanced train dataset
Step 1: Compute the desired number of samples for each class ( $n_s(c_i)$ )
  if stgy == Constant then
    if  $c_i == Minority$  then
       $n_s(c_i) \leftarrow \left( \frac{n}{m} \right) - \left( \frac{n - m}{m(\eta + 1)} \right)$ ; //  $1 \leq i \leq m$ 
    end
  else
     $n_s(c_i) \leftarrow \left( \frac{n}{m} \right) - \left( \frac{n - m}{m((1 - \eta) + 1)} \right)$ 
  end
  end
  else if stgy == Variant then
    foreach  $c_i$  in Train data do
       $n_s(c_i) \leftarrow \left( \frac{n}{m} \right) - \left( \frac{n - m}{m(\eta_i + 1)} \right)$ 
    end
  end
Step 2: Compute the amount of sampling for each class ( $Cperc(c_i)$ )
  foreach  $c_i$  in Train data do
     $Cperc(c_i) \leftarrow \frac{n_s(c_i)}{n_{c_i}}$ ; //  $n_{c_i}$  = no of samples belongs to  $c_i$  in train data
    if  $Cperc(c_i) > 1$  then
      for  $j \leftarrow 1$  to  $(n_s(c_i) - n_{c_i})$  do
        Compute  $k$  nearest neighbours for  $j$  using dist; //  $k = 5$ 
        Choose a random neighbour among  $k$ ; // dist = HVDM or HEMOM
        Compute difference (diff)
        Generate a random number (gap) between 0 to 1
        Synthetic Sample =  $i + gap \times diff$ ; // Over-Sample
      end
    end
    else if  $Cperc(c_i) < 1$  then
      Remove  $(n_{c_i} - n_s(c_i))$  samples from  $c_i$  at random; // Under-Sample
    end
    else if  $Cperc(c_i) = 1$  then
      Samples in  $c_i$  remains unchanged
    end
  end

```

3.4. Validator module

The outcome of the pre-processor module is balanced train data that serves as input for training and validating six classification models. The main aim of the validator module is to compare the classification model predictions with a real-world dataset, for assessing its predictive power. This module permits to establish the degree of confidence of the model which supports in choosing the trained classifier [50]. Thereafter the evaluator module assesses the model's adequacy to the needs of the decision-maker.

The six classifiers are broadly divided into five methodologies: distance-based approaches which include similarities and inter-class boundaries, tree construction approaches, neural networks and finally hybrid and ensemble approaches.

Similarities: “similar instances belongs to similar class” is the fundamental principle behind the classification model based on the similarities [51]. This method is commonly known as k -Nearest Neighbors (k -NN) [29], where k signifies the number of nearest neighbors. Appropriate distance metric is used to identify the k -nearest neighbors of a test instance and then predict it as the frequently occurring class label. The choice of k -value and the distance measure significantly impact the performance of k -NN. Distance metrics like Euclidean, manhattan, mahalanobis, and many others are available for estimating the similarity

between instances. Heuristics, and trial and error are the widely preferred ways to choose the optimal k value.

Inter-class boundaries: Support Vector Machine (SVM) is a well-known binary classifier developed based on the inter-class boundaries [30]. Let f_1, f_2, \dots, f_m be the set of binary classifiers for m classes respectively. Each f_i learns how to differentiate c_i from the remaining classes and predicts whether the test instance, x_i belongs to c_i or not. Each f_i trains its hyperplane with maximal margins and predicts the test instance.

The few ways to improve the prediction accuracy of SVM include changing the kernel function and tuning the cost parameter. Changing kernel function transforms the data into a higher dimensional space to convert non-separable data into linearly separable data. Linear, radial, and polynomial are the commonly used kernel functions. In this work, radial basis kernel function is used to validate SVM's performance on the balanced train data. The cost function uses the default value specified by the caret package as in [52].

Tree construction approaches: Decision tree, a widely used classification model, is formed based on the set of hierarchical decisions on features to divide data into smaller subsets [31]. If a subset contains instances that belong to the same class, then the subset is called a pure partition and can no more be divided. If there exist some subsets with a mixture of classes, then either stop splitting and accept the impure subsets or recursively split the remaining subtree.

Iterative Dichotomizer (ID3), Classification and Regression Trees (CART), C5.0, and others are some traditional decision tree induction algorithms. Among all these decision tree algorithms, C5.0 algorithm with Information Gain (IG) as the split criteria is used in this work to construct the decision tree with the balanced train data (D_{train}). The feature with highest IG is selected as the split-attribute.

Neural networks: A Neural network comprises of fully or partially connected input, hidden, and output layers. A perceptron is the basic form of a neural network with only input and output layers [32]. The input layer receives features from data and transmits to the output layer without performing any computation. Only the output layer will do mathematical computations on received inputs and classify accordingly. The singed linear function is the activation function used by perceptron for classification. The weights associated with input nodes and the bias are adjusted using optimization approaches. While training, if the predicted class label mismatches with the original class label, then weights associated with input nodes are updated based on the error value. One single training record undergoes multiple iterations, and each iteration is termed as an epoch. This process of iteratively updating weights continues until both labels match, or if maximum epoch is reached; in the latter, the prediction label by the end of the last epoch is considered.

Hybrid and ensemble approaches: Classification methods that tend to aggregate multiple classifiers for generating a single predictive model is known as ensemble learning [53]. Bagging and boosting are the well-known ensemble approaches [54].

Bagging and boosting: Let t be the number of trials; for each trial, balanced D_{train} is sampled with replacement (bootstrap samples) so that few instances from balanced train data appear more than once, and few instances may not appear at all. A classifier (M_t) is generated at the end of each trial. Bagging aggregates all M_t classifiers in order to form resultant bagged classification model (M^*) is known as bootstrap aggregation. Let x be the test instance that needs to be classified; then each M_t will predict the class and M^* considers the majority voted class label as the predicted class label for x .

Boosting assigns distinct weights to each instance at each trial. The weight of the instance differs from trial to trial. Similar to bagging, each M_t will predict the test instance and the majority voted class is considered as a predicted class label for the test instance. The validator module uses the combination of both bagging and adaptive boosting (AdaBoost) named as Bagged AdaBoost (AdaBag) [55] is used in the validator module.

Random forest: Another popular method of ensemble learning is

random forest that uses a collection of decision trees for prediction [33]. Random forest works better for large datasets compared to most classifiers. Each tree in the forest is generated by using a feature subset and a bootstrapped dataset. Two-thirds of the samples (in-bag samples) from balanced train data are used to model the random forest, whereas remaining samples (out-bag samples) are used to perform internal cross-validation to know the performance of random forest.

The two parameters that needs to be defined in order to construct a random forest are as follows: first, number of decision trees to be there in the forest and second, number of features selected for splitting the decision trees while growing the forest. Though the number of trees in forest has very less impact on performance, the forest with high number of trees is robust. Let V_x is a vector associated with an instance (x) where each element in V_x represents a predicted class by each decision tree in random forest. If random forest is modelled with q decision trees, then V_x contains q elements. If V_x contains c_i as majority voted class then x is predicted as c_i . A decision rule can be interpreted as a path from the root node to the leaf node in DT. Extracting decision rules from a single decision tree is quite simple and easy.

3.4.1. *k-fold cross validation*

Among all classifiers (k -NN, SVM, decision tree, neural network, bagged adaboost, and random forest), an optimal and suitable classification approach needs to be selected. One of the standard procedures to evaluate and compare a set of classification models by applying multiple data splits is *k*-fold Cross Validation [56]. This technique will randomly partition the balanced train dataset into k equal size disjoint subsets. A set of classifier models are trained and tested (validated) repeatedly for k times. Each time, iteratively $k - 1$ subsets or folds are used for model training, and the remaining fold is used for model validation. For conducting an unbiased test, the samples present inside the folds remain constant. Validation involves parameter tuning too; it is the selection of appropriate values for model parameters which will have an impact on the performance of the classifier.

Based on the model performance and model-interpretability the optimal trained classifier is used for the extraction of rules. In the allergy diagnosis support system, appropriate reasoning for prediction is required by the immunologists. Since the ML classifiers' interpretability is not sufficient to meet the criteria imposed by immunologists, a recent XAI approach known as post-hoc explainability [34] is formulated and applied to the classifier in order to explain the system generated decisions. The main purpose of imposing post-hoc explainability technique is to communicate the understandable information about how the already trained and validated classifier is generating decisions to the given inputs. The two shallow ML classifiers that are considered in this study are support vector machine and random forest which requires the post-hoc explainability technique in order to explain the system generated decision predictions. The rules extracted from random forest are understandable and concise. Moreover, random forest classifier would provide clear explanations on simplification. Hence it is chosen for rules extraction.

3.4.2. *Rule extraction*

A random forest is a collection of trees. Each path from the root node to a leaf node in a tree is interpreted in the form of IF-THEN rule, where the IF part is the antecedent (condition), and the THEN part is the consequent (Prediction). Combining all the rules extracted from each decision tree results in a set of rules. These rules are stored in a rule base for expert assessment [57]. It is validated by a panel of immunologists. Usually, clinical assessment adds more relevance than mere statistical analysis. Experts suggestions often include additional factors that are overlooked by the trained model. It is quite challenging for the CDSS to exactly represent the expert knowledge in the form of classification models. Moreover, for some sections (trees or rules) there may be no modification applicable; but the expert assessment enables us to know the clinical significance of the diagnostic results yielded by the CDSS.

Table 4

Rules extracted from random forest based on variant sampling strategy.

Rule num	Condition	Pred
1	Tea in ('NR') & gram in ('0','HR','MR','NR') & runningnose in ('no') & redrashes in ('yes')	UT
2	Cottondust in ('HR','MR') & beans in ('LR','MR') & runningnose in ('no') & cough in ('no')	UT
3	Drumstick in ('0','MR','NR') & runningnose in ('yes') & wheezeBlocks in ('no') & swelling in ('yes')	RH_UT
4	Drumstick in ('LR') & tamarind in ('LR','MR','NR') & sneeze in ('no') & swelling in ('yes')	UT
5	Ginger in ('HR','LR','MR','NR') & tamarind in ('0') & sneeze in ('no') & itching in ('yes')	UT
6	Chowchow in ('LR','MR','NR') & sneeze in ('no') & redrashes in ('yes')	UT
7	Housedust in ('HR','NR') & aspergilus in ('LR') & beans in ('LR','MR','NR') & redrashes in ('no')	RH
8	Crab in ('LR') & sneeze in ('yes') & itching in ('no')	RH
9	Pollen in ('HR','NR') & fishB in ('LR','MR','NR') & redrashes in ('yes')	UT
10	Cockroach in ('MR') & crab in ('0','LR') & sneeze in ('yes') & swelling in ('yes')	RH_UT
11	Cabbage in ('0','LR') & ginger in ('HR','LR','MR','NR') & redrashes in ('yes')	UT
12	Parthenium in ('HR','LR') & potato in ('0') & garlic in ('0','LR') & runningnose in ('yes')	RH
13	Sneeze in ('no') & wheezeBlocks in ('no') & itching in ('no') & Age in ('G1','G2','G3')	OTHERS
14	MilkP in ('HR','LR','MR','NR') & peas in ('LR','MR','NR') & maida in ('0','LR') & tamarind in ('0') & cough in ('yes')	RH
15	Pollen in ('HR') & radish in ('LR','MR','NR') & itching in ('no')	RH
16	Mutton in ('HR','LR','MR','NR') & fishA in ('0','LR') & beetroot in ('MR','NR') & cough in ('yes') & redrashes in ('no')	RH
17	Radish in ('LR','MR') & runningnose in ('no') & sneeze in ('no')	UT
18	Banana in ('0') & vazpooThandu in ('NR') & itching in ('yes') & redrashes in ('no')	OTHERS
19	Housedust in ('HR','LR','NR') & curd in ('MR','NR') & tamarind in ('0') & wheezeBlocks in ('no')	UT
20	fishA in ('MR','NR') & crab in ('0','LR') & avaraikai in ('HR','LR','MR','NR') & gourds in ('0','MR') & runningnose in ('yes')	RH
21	Gram in ('0','HR','MR','NR') & tamarind in ('0') & cough in ('yes') & headache in ('yes')	RH
22	Cottondust in ('MR') & greens in ('0','MR') & cough in ('yes') & redrashes in ('no') & FHistory in ('yes')	RH
23	Prawns in ('HR','LR','MR','NR') & beetroot in ('NR') & brinjal in ('0','LR') & garlic in ('0') & sneeze in ('no')	OTHERS
24	Curd in ('MR','NR') & gram in ('LR') & sneeze in ('no') & itching in ('yes')	UT
25	Lfinger in ('0') & runningnose in ('no') & sneeze in ('no') & wheezeBlocks in ('yes') & headache in ('no')	OTHERS
26	Aspergilus in ('LR','MR','NR') & cockroach in ('HR','LR','MR') & prawns in ('HR','LR','MR') & gourds in ('0') & sneeze in ('yes') & itching in ('no')	RH
27	Cockroach in ('MR') & fishA in ('0','LR') & drumstick in ('LR') & runningnose in ('yes') & swelling in ('no')	RH
28	Kovaikai in ('0','LR') & ginger in ('HR','LR','MR','NR') & redrashes in ('yes')	UT
29	Cauliflower in ('LR','MR','NR') & gourds in ('LR','NR') & paneer in ('0','LR') & sneeze in ('yes') & redrashes in ('no')	RH
30	Lfinger in ('MR','NR') & tomato in ('0','LR') & runningnose in ('yes') & itching in ('no')	RH
31	Cockroach in ('LR','NR') & garlic in ('0') & cough in ('no') & headache in ('no') & swelling in ('no')	OTHERS
32	Aspergilus in ('HR','LR','MR') & wheezeBlocks in ('no') & swelling in ('yes')	UT
33	Cockroach in ('HR') & egg in ('0','LR','MR') & cough in ('yes') & redrashes in ('no')	RH
34	Pollen in ('LR','MR','NR') & egg in ('HR','NR') & potroot in ('MR','NR') & sneeze in ('no') & FHistory in ('yes')	OTHERS
35	Gourds in ('0') & vazpooThandu in ('0') & yams in ('MR','NR') & garlic in ('MR','NR') & redrashes in ('no')	RH_O
36	Cockroach in ('HR','LR') & fishB in ('LR','MR','NR') & garlic in ('LR','MR','NR') & itching in ('no')	RH
37	Tea in ('0','MR','NR') & egg in ('0','LR','MR') & runningnose in ('no') & wheezeBlocks in ('no') & redrashes in ('no')	OTHERS
38	Pollen in ('HR','LR','MR') & fishA in ('0','LR') & sneeze in ('yes') & itching in ('yes')	RH_UT
39	Beetroot in ('LR','MR','NR') & sneeze in ('yes') & headache in ('yes') & Age in ('G2','G4','G5')	RH
40	Curd in ('MR','NR') & tea in ('0','LR') & dhal in ('0','MR','NR') & tamarind in ('0') & wheezeBlocks in ('no')	RH
41	fishB in ('0') & garlic in ('LR','MR','NR') & runningnose in ('yes') & itching in ('no')	RH
42	Yams in ('MR','NR') & runningnose in ('yes') & FHistory in ('no')	RH_O
43	Tea in ('0','LR') & prawns in ('0','HR','NR') & channa in ('0','HR','NR') & maida in ('0','NR') & wheezeBlocks in ('no') & itching in ('no')	OTHERS
44	Cottondust in ('MR') & sneeze in ('yes') & swelling in ('no') & FHistory in ('no') & Sex in ('m')	RH
45	Prawns in ('0','HR','MR','NR') & cauliflower in ('MR','NR') & yams in ('0') & wheezeBlocks in ('no')	UT
46	Cauliflower in ('NR') & runningnose in ('no') & itching in ('no')	OTHERS
47	Beef in ('MR','NR') & carrot in ('0','MR','NR') & peas in ('LR','MR','NR') & wheezeBlocks in ('yes')	RH_O
48	Aspergilus in ('HR','LR','MR') & fishB in ('0','LR','NR') & gram in ('0','HR','MR','NR') & runningnose in ('yes')	RH
49	Curd in ('MR','NR') & wheezeBlocks in ('no')	UT
50	Cottondust in ('LR','MR','NR') & parthenium in ('LR','NR') & runningnose in ('yes') & redrashes in ('no')	RH
51	Curd in ('0') & sneeze in ('no') & redrashes in ('no')	OTHERS
52	Tea in ('0','MR','NR') & cough in ('no') & redrashes in ('no')	OTHERS
53	Tea in ('0','MR') & fishA in ('MR','NR') & wheezeBlocks in ('yes') & itching in ('yes') & swelling in ('yes')	UT_O
54	Cabbage in ('0','LR') & sneeze in ('no') & redrashes in ('yes')	UT
55	Housedust in ('HR','MR') & MilkC in ('0') & greens in ('0')	RH

Table 4 presents the sample rules extracted from random forest based on variant sampling strategy and this model generates rules for all the m classes.

3.5. Evaluator module

Evaluator module aims at assessing the optimal model's adequacy to the needs of the final users [50]. This module evaluates the CDSS by analyzing the performance of the trained classifier (random forest) on new instances (D_{test}). The test instance is associated with a vector where each element in the vector is the predicted class label generated by one tree in the random forest. Each tree will give a prediction to test instance, and the random forest will consider the majority class in the vector as the final prediction to the test instance. The final prediction

(diagnostic result) is made available to the clinicians.

Among many performance measures that are available in literature, accuracy, kappa, sensitivity, specificity, and area under the curve (AUC) value are selected for this study. All these measures are different in their evaluation focus [58]. Accuracy estimates the efficiency of a classifier to predict the instances correctly. Accuracy is the appropriate measure for binary class problems compared to the multi-class problems because it is unclear in representing the split down of accuracy among more than two classes. Kappa is similar to classifier accuracy except that kappa measure normalizes at the baseline of random chance present in a dataset. Though kappa is an underused statistical measure, it is appropriate for multi-class imbalanced datasets compared to accuracy. Sensitivity, also known as recall, measures the ability of a class to predict instances belongs to that class correctly. Whereas specificity measures the ability of a

class to reject instance that does not belong to that class. A Receiver Operating Characteristic (ROC) graph is used to select the classifier based on the visualizations of the classifier's performance. These graphs are widely used in diagnostic systems to describe the trade-off between true positive rate (y-axis) and false negative rate (x-axis). The point (0, 1) represents the flawless classification whereas the point towards the north-east always resembles the best compared to other points in the graph. The curve that reaches the peak true positive rate without any variation in direction is comparatively better than the curve with too many variations [59]. Area under the ROC curve value (AUC value), represents the ability to discriminate one class with remaining classes.

In a multi-class problem, sensitivity, specificity, and AUC value are calculated using one-vs-all approach; select one class and consider all the remaining classes as other class and then calculate similar to a binary classification problem. The performance of a multi-class classifier can be summarized using *Confusion Matrix*. Let c_i denote the columns of original class labels, whereas \hat{c}_i is the rows of predicted class labels by classifier and n represents the total number of elements in the confusion matrix. Let $S_{\hat{c}_i}$ represent the total number of elements in \hat{c}_i row and S_{c_i} represent the total number of elements in c_i column. The diagonal elements, D in the confusion matrix are predicted correctly by the classifier and the non-diagonal elements, ND in the confusion matrix are mis-predicted by the classifier. $D_{(c_i, \hat{c}_i)}$ represents the diagonal element (True Positive) for c_i and \hat{c}_i . Let TN_{c_i} is the sub-matrix (True Negative) formed by removing the entire row under \hat{c}_i and column under c_i and S_{TN} represent the total number of elements present in TN_{c_i} sub-matrix. The difference between D and S_{c_i} represented as S_{FN} (False Negative) and S_{FP} (False Positive) represents the difference between D and $S_{\hat{c}_i}$. The previously mentioned performance measures were computed directly from the confusion matrix.

$$\text{Accuracy} = \frac{\sum_{i=1}^m D_{(c_i, \hat{c}_i)}}{n}$$

$$\text{Kappa} = \frac{\sum_{i=1}^m D_{(c_i, \hat{c}_i)} - EA}{n - EA}$$

$$EA = \frac{\sum_{i=1}^m (S_{c_i} \times S_{\hat{c}_i})}{n}$$

$$\text{Error} = \frac{\sum_{i=1}^m ND_{(c_i, \hat{c}_i)}}{n}$$

$$\text{Sensitivity of a class, } c_i = \frac{D_{(c_i, \hat{c}_i)}}{S_{c_i}}$$

$$\text{Specificity of a class, } c_i = \frac{S_{TN}}{S_{TN} + S_{FP}}$$

4. Experiments and results

Patients with symptoms like running nose, sneezing, cough, wheezing, red rashes, itching, and headache visit a general physician before approaching an immunologist. If the symptoms are due to allergy then the regular treatment procedures provided by general physicians may not control the allergic manifestations. Based on the type and severity of the symptoms, the patients are advised to visit an allergy testing center.

4.1. Experimental settings

In this study, medical records of 878 allergy patients are used. They were obtained over a period of six months from Good Samaritan Kilpauk Allergy Testing Centre, Chennai, Tamil Nadu, India. Due to diversity in environmental conditions, food habits, lifestyle and symptom-

manifestations the components of the allergy test are tailored to the people residing in southern parts of India. When a patient visits the allergy clinic, the initial diagnosis starts with an interaction with the patient by the healthcare workers or immunologist. The presence, duration and severity of physical allergic symptoms along with certain demographic information and family background information about the patient are recorded.

According to the individual's food habits and lifestyle, a customized list of allergens were tested using intradermal skin test procedure. As a prerequisite for testing, it is required that the patient should not have taken any anti-allergy drugs or related medication for three days. Tests were conducted by injecting 0.01 ml of the selected allergen from a patient-specific customized list of allergens using sterile, disposal, plastic 1ml tuberculin syringe on upper half of the forearm. If the patient is allergic to an injected allergen, then a wheal-and-flare reaction appears at the point of the allergen-prick; else no reaction will appear on the tested area. An hourly-based consecutive reading is taken to finalize the test result. The test results constitute the raw IDST data (D_{raw}) which is the input to the CDSS.

The CDSS is designed, developed and tested using the following tools and packages: All experiments were implemented in RStudio (Version-1.2.5033). UBL package [60] was used to implement the sampling algorithm (SMOTE), Caret package [52] to streamline the training process of classification models and ggplot2 package [61] to visualize the data and performance results. inTrees [57] and devtools [62] packages were used to extract rules from random forest. Xtable [63] function was used to return and display the extracted rules.

The developed CDSS was deployed in the mobile platform using 'Visual Studio Code' Editor with build-in Java Script IntelliSense for scripting, and 'Expo' framework for React applications generation and code simulation.

4.2. Results and discussion

This section presents the observations, findings, and also some intermediate results of the various modules.

There are 7 classes as shown in Table 2; two classes for specific allergic disorders namely 'urticaria' and 'rhinitis'; one class for other allergies such as asthma, eczema, drug allergies which are not in the scope of this study; three classes for representing the allergy comorbidities, and finally a class 'normal' which represents the patients with no symptoms of allergy. From Fig. 3 the effect of physical signs and symptoms on all the 7 classes can be analyzed. These signs and symptoms are recorded as nominal (yes/no) attributes. The symptoms like running nose, cough, sneeze and wheeze are frequent in patients suffering from rhinitis whereas symptoms like itching, swelling and red rashes are present in patients with urticaria.

The amount of sampling that needs to be applied on a class depends on both the sampling strategy and the number of samples belonging to that class. Amount of sampling or the sampling percentage resembles whether the class is over-sampled or under-sampled. Both the sampling strategy and the sampling technique significantly impact the performance of the classifier. Tables 5 and 6 present the effect of the three distinct amounts of sampling strategies on the performance of classification models. Moreover, from Fig. 4 we can observe that random forest scores higher than all other classification approaches before sampling; however other classification approaches yield comparable results after using sampling techniques. Without applying sampling approaches, random forest achieves 84.52% followed by SVM with 82.92% accuracy. After applying SMOTE variant sampling strategy, SVM achieves 81.04% and C5.0 achieves 81.62% whereas random forest achieves 83.07% accuracy.

In the case of allergy diagnosis, high sensitivity is important because leaving the allergy patient without providing appropriate treatment is not advisable. It is not harmful to give allergy treatment to the non-allergic patient, so specificity value to identify non-allergic patients

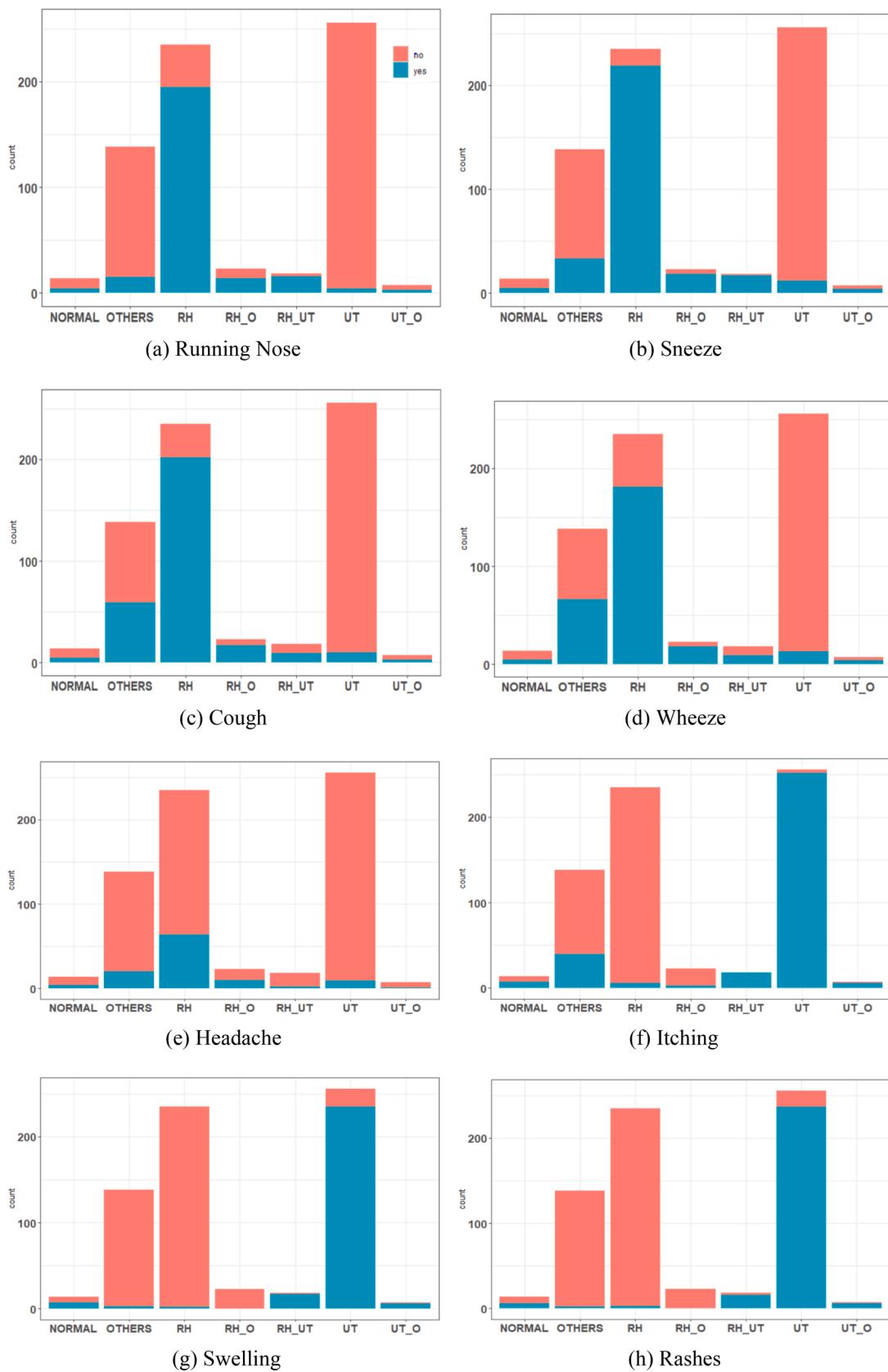
**Fig. 3.** Allergic symptoms effect on allergic diseases.

Table 5
Accuracy of classifiers on D_{train} .

	Unbalanced	Ideal	Constant	Variant
K-NN	76.12	73.23	79.31	75.83
SVM	82.92	88.13	79.45	81.04
C5.0	82.78	81.33	81.04	81.62
NN	74.96	64.25	65.41	72.65
AdaBag	82.20	57.74	73.66	79.74
RF	84.52	84.37	81.48	83.07

Table 6
Kappa of classifiers on D_{train} .

	Unbalanced	Ideal	Constant	Variant
K-NN	64.00	68.77	73.40	66.00
SVM	74.74	86.16	72.87	73.42
C5.0	74.77	78.22	75.54	74.96
NN	63.91	58.30	55.79	62.65
AdaBag	74.06	50.69	65.06	71.69
RF	77.46	81.77	75.84	76.65

correctly is not as important as sensitivity for not leaving a single allergic patient. Classifying the patient with combinations of allergies into individual allergy class is not desired. The real purpose of the system fails if an allergic patient is predicted as normal. By applying the one-vs-all approach, the sensitivity, specificity, and AUC values of a random forest classifier for each class in D_{train} is presented in Table 7. The sensitivity and specificity values of the remaining classifiers are presented in the Appendix.

Random forest achieves a sensitivity of 0.9538, 0.9259 and 0.9166 for RH, UT and RH_UT respectively for constant sampling strategy, and 0.9244, 0.9322 and 0.9791 respectively for variant sampling strategy. The sensitivity of RH and UT is high compared to the comorbidity class RH_UT in constant sampling strategy. The focus of this work is to improve the sensitivity of comorbidity classes rather than the usual individual disease classes. This is ensured by assigning appropriate parameter values to the sampling strategy; the η value to each class in the variant sampling strategy are 0.11, 0.44, 0.11, 0.18, 0.44, 0.25, 0.05 for c_1 to c_7 respectively. Consequently, the relative improvement in the sensitivity between RH, UT and RH_UT is observed. A higher sensitivity rate is achieved for comorbidity class than the individual classes in variant strategy.

Fig. 5 represents the ROC of random forest classifier on D_{train} . These

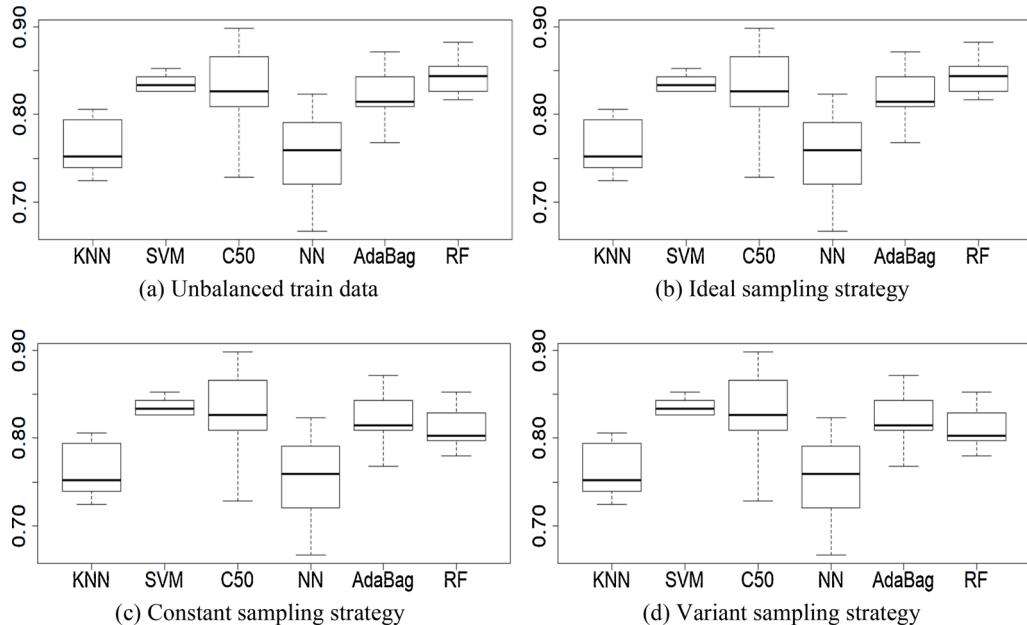
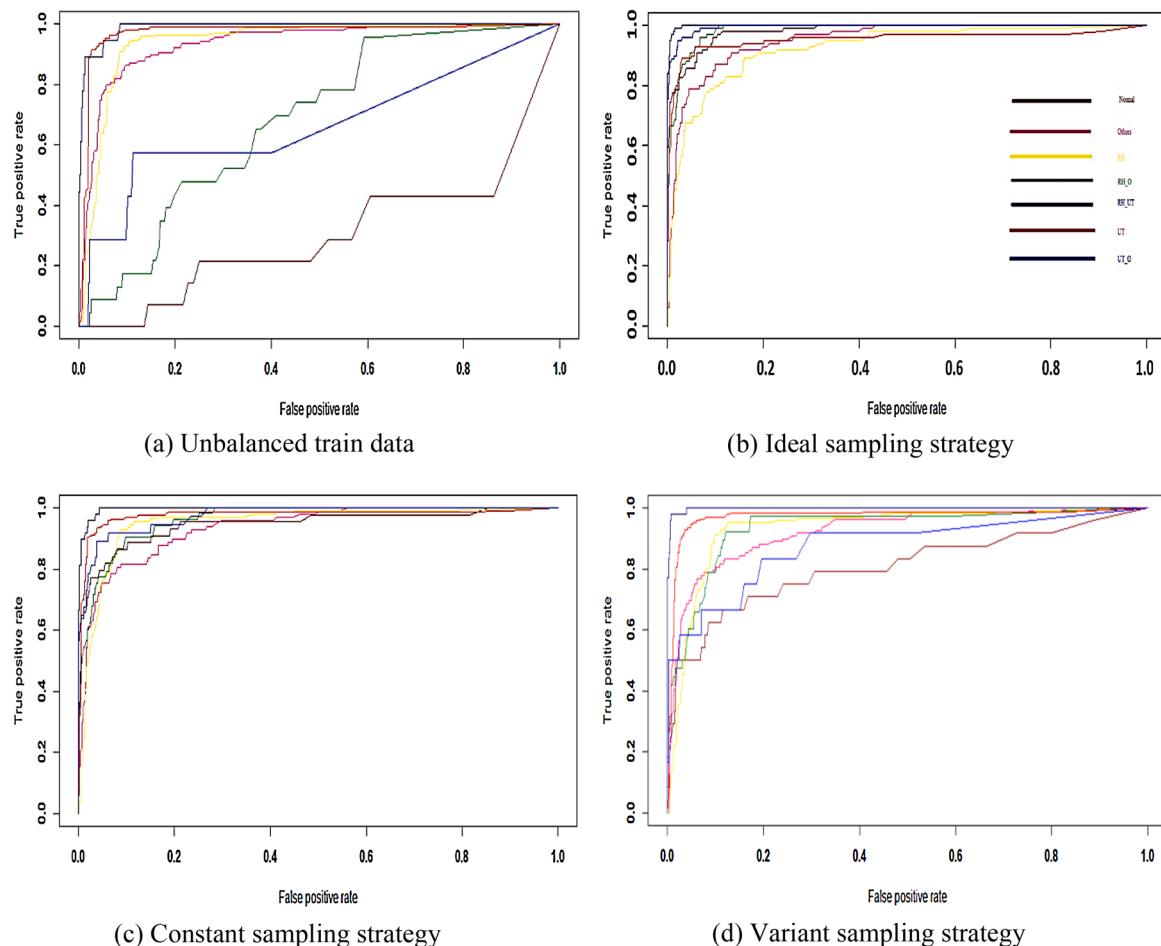


Fig. 4. Box plot of classifier performance on D_{train} .

Table 7
Sensitivity, specificity, AUC values of a random forest classifier on D_{train} .

Random Forest D_{train}		NORMAL	OTHERS	RH	RH_O	RH_UT	UT	UT_O
Unbalanced	Sensitivity	0	0.7971	0.9489	0	0.7777	0.9258	0
	Specificity	1	0.9367	0.8882	1	0.9925	0.9632	1
	AUC value	0.3050	0.9369	0.9420	0.6830	0.9898	0.9712	0.6639
Ideal	Sensitivity	0.7959	0.7879	0.6767	0.8889	0.9899	0.8586	0.9082
	Specificity	0.9747	0.9595	0.9611	0.9595	0.9616	0.9780	0.9933
	AUC value	0.9815	0.9518	0.9250	0.9858	0.9988	0.9543	0.9952
Constant	Sensitivity	0.3636	0.7551	0.9538	0.3962	0.9166	0.9259	0.5945
	Specificity	0.9969	0.9427	0.8790	0.9968	0.9891	0.9579	0.9954
	AUC value	0.9458	0.9359	0.9508	0.9624	0.9965	0.9754	0.9748
Variant	Sensitivity	0.1250	0.7500	0.9244	0.3157	0.9791	0.9322	0.2500
	Specificity	0.9970	0.9520	0.8798	0.9969	0.9891	0.9449	0.1666
	AUC value	0.8072	0.9228	0.9350	0.9357	0.9982	0.9730	0.8757

Fig. 5. ROC of the random forest classifier on D_{train} .

figures support in deciding the choice of the random forest classifier based on True Positive Rate of the minority class across all four sampling strategies. In variant sampling, values are given based on the domain requirement, where by the chance of missing out comorbid allergic conditions are minimized. From the curves it can be inferred that class c_5 (RH_UT) has a better True Positive Rate for random forest classifier trained with variant strategy. The True Positive Rate of remaining classes followed almost the same decreasing order of values. Hence, random forest classifier developed using the Variant sampling strategy is chosen as a trained classifier and used in the evaluator module over the test set (D_{test}) to generate the diagnostic result and to analyze the performance of the CDSS.

The random forest (variant strategy) achieves an accuracy of 86.39% for the samples in the D_{test} . The same set of samples is forwarded to four clinicians in order to evaluate their performance. The results of evaluation of the four clinicians on D_{test} before and after using the CDSS is presented in Table 8. The performance of clinicians after using allergy diagnosis support system is increased by 4.59% compared to the performance of clinicians before using CDSS.

Table 8
Performance of clinicians on D_{test} .

Clinician ID	Without using CDSS	Using CDSS
Clinician #1	80.47	83.43
Clinician #2	78.10	81.65
Clinician #3	73.96	78.10
Clinician #4	76.33	84.02
Average	77.21	81.80

4.3. Threads to validity

The CDSS developed and deployed in this work is tailored for south Indian regions (Tamil Nadu and Kerala). However, allergens differ based on environment, food habits, geographical locations, and personal interests. The same CDSS would not be suitable or efficient when deployed for other parts of India; however, training the same framework with appropriate and relevant data will yield better systems.

The validity of the results also depends on the class distribution of the data. This depends on the season during which the allergy samples are observed and recorded. The samples recorded during the summer season (April-June) are more likely to have skin related allergic disorders such as urticaria whereas samples recorded during winter and rainy season (October-December) are more likely to contain airway allergic diseases such as rhinitis and asthma.

Variations in performance measures are observed while training classification models. This happens due to slight variations in class boundaries as well as skewed data distribution for the class label. In order to validate and develop a generalized classifier with minimal out-of-sample error and minimal variations in performance measures, k -fold cross validation and sampling strategies are used in the validator module. Using stratified sampling and incurring more computational time by repeated validation may yield significant robust results.

5. CDSS-integrated mobile application

With the increase in computational power and ubiquity of mobile applications, the global usage of technology in edge devices is expanding opportunities to enhance relevant advancements in proprietary

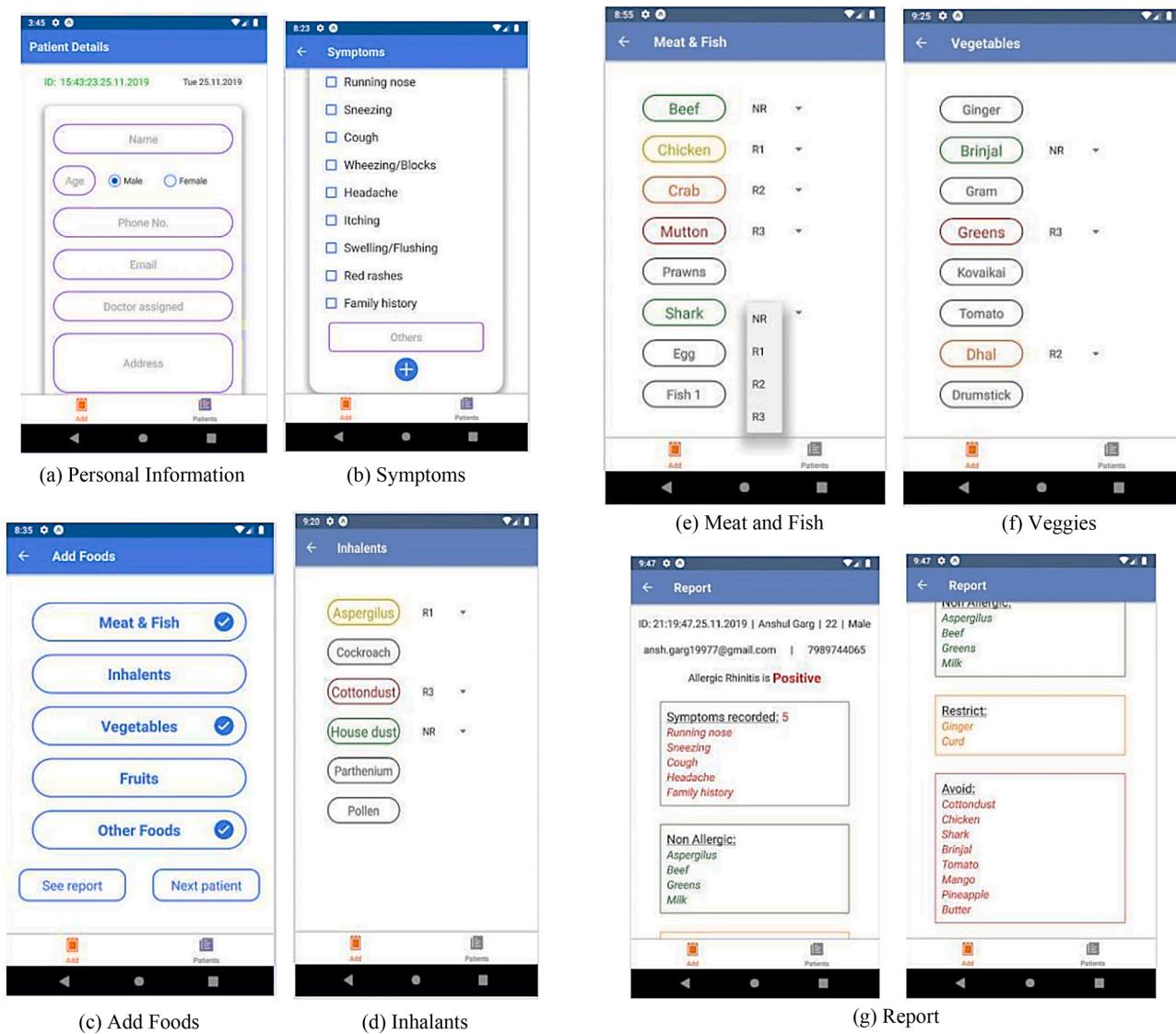


Fig. 6. Screenshots of the CDSS deployed mobile application.

applications. Edge devices such as smartphones are now preferred while developing the mobile applications that are integrated with CDSS to encounter the needs of physicians. This simplifies maintenance of medical records, manages time and schedules in initial patient-physician interaction and also supports physicians in clinical decision-making tasks. CDSS deployed in mobile applications are designed in such a way that the services can be provided with very limited resources over the web(internet). Moreover, in recent times physicians and house surgeons are also progressively adapting mobile applications (apps) to facilitate the real-time decision support at the point of care. In CDSS enabled health applications, much importance is given to interpretable results along with personalized treatment suggestions. The substantial potential of the mobile application to integrate the data from multiple sources like medical observations, environmental conditions, regional allergens, expert knowledge and scientific literature results in refined decisions. This assists clinicians in affirming their decisions and in providing patients with personalized diagnosis.

In this work, the developed CDSS is deployed in mobile platforms using a smartphone app. The platform also supports diagnosis of multiple coexisting allergies and also provides decisions with interpretable reasoning and justification. Moreover, the app interface and functionalities contribute in furnishing the layout of treatment suggestions. These treatment recommendations present the various non-allergic and

allergic agents (Inhalants, Contactants and Ingestants) that need to be either restricted or avoided by the patient. The recommendations and the report (diagnostic result) are automatically generated and made available to the patients so that they can understand their allergy condition. This enables the patient to plan and organize their lifestyle, food habits and other regular activities accordingly. Moreover, these diagnostic results along with treatment suggestions serve as guidelines for physicians in providing their decisions with confidence.

The screenshots of the mobile application are shown in Fig. 6. When a patient visits an allergy clinic, the clinician interacts with the patient and records the personal information in using the patient details form as shown in Fig. 6a. Then the clinician navigates to the symptoms form and records the physical symptoms based on his personal interaction with the patient. On further discussion with the patient, the clinician gets to know about the environmental conditions and the food habits of the patient. Based on the patients' response the list of allergens is selected using the add foods form. The values for list of inhalants, contactants and ingestants are entered based on the observations of the intradermal skin test results. The related forms for all of the above activities are shown in Fig. 6–f. If any patient-specific symptoms or allergen is not listed in the forms the clinician can manually enter the patients symptoms. Then the CDSS framework that is integrated with the mobile application will process the recorded data and generate the personalized

report. The report provides the diagnostic result and also a suggested treatment plan. The layout of personalized report is presented in Fig. 6g.

6. Conclusion and future directions

The machine learning based framework developed for allergy diagnosis intends to assist junior clinicians and house surgeons at allergy centers in their decision-making tasks. The focus is to aid the clinicians in dealing with comorbid allergic conditions rather than easily diagnosable primary allergies. The data cleaning module and the modified sampling approaches in the data sampling module enhances the quality of the intradermal test data. These pre-processing steps catalyze the performance of the learning algorithms. The cross-validation approach ensures that the learning algorithms do not over-fit the training data. Ensemble classification approaches perform better than traditional approaches. Random forest with constant strategy sampling achieves better sensitivity compared to all other cases; however, in variant strategy, difference in value among the classes ensures better sensitivity of RH_UT class. Post-hoc explainability technique of explainable artificial intelligence is considered for extraction of interpretable rules represented by the random forest classifier. The rules are presented as simple condition-prediction (IF-THEN) rule format. These rules are also made available to an immunologist for checking their clinical relevance. For better usability and portability, the developed CDSS is deployed in a mobile platform as a smartphone app. Physicians can easily access it at the point of care. From the patients' perspective, the reports and treatment guidelines are easily accessible on installation of the application.

The allergy diagnosis support system can be made more efficient with the use of meta-heuristics data-processing techniques. Apart from cleaning and sampling, data transformation approaches such as feature selection can be incorporated. Novel learning algorithms and responsible AI approaches can be merged for ensuring ethical, secure and transparent use of technology. Incorporating prognosis details, results of treatment and patient feedback will add more relevance to the system. This may serve as a source of knowledge and reference for junior clinicians, and even experts.

Authors' contribution

Ramisetty Kavya: developed software, performed the statistical analysis, wrote the paper; Jabez Christopher: conceived and designed the analysis, collected the data, performed the analysis, wrote the paper; Subhrakanta Panda: performed the analysis, wrote the paper; Bakthasinh Lazarus: conceived and designed the analysis, contributed the data, performed the clinical analysis.

Acknowledgments

The authors would like to thank Dr. George Moses for providing data and conceptual guidance for this work. The authors thank Dr. Julie Dayalan and the clinicians at the Good Samaritan Lab Services and Allergy Testing Centre, Kilpauk, Chennai, India, for their support during the data collection and pre-processing stages. The authors also thank Dr. Praylin and the clinicians at the Joyce Clinical Lab Services, Marthandam, Kanyakumari district for their contribution to the evaluation and testing phases of the CDSS. Android application development and some clinical aspects of this work were partially funded by DST-SERB start-up research grant FILE NO: SRG/2019/001801.

Declaration of Competing Interest

The authors report no declarations of interest.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the

online version, at <https://doi.org/10.1016/j.bspc.2021.102681>.

References

- [1] L.S. King, Signs and symptoms, *JAMA* 206 (5) (1968) 1063–1065.
- [2] S.J. Galli, M. Tsai, A.M. Piliponsky, The development of allergic inflammation, *Nature* 454 (7203) (2008) 445–454.
- [3] K. Bhattacharya, G. Sircar, A. Dasgupta, S.G. Bhattacharya, Spectrum of allergens and allergen biology in India, *Int. Arch. Allergy Immunol.* 177 (3) (2018) 219–237.
- [4] K. Kulthanan, S. Chusakul, M.T. Recto, M. Gabriel, D.C.-W. Aw, N. Prepagan, A. Wong, J.L. Leong, H. Foong, V.T. Quang, et al., Economic burden of the inadequate management of allergic rhinitis and urticaria in Asian countries based on the ga2len model, *Allergy Asthma Immunol. Res.* 10 (4) (2018) 370–378.
- [5] M.T. Recto, M.T. Gabriel, K. Kulthanan, P. Tantilipikorn, D.C.-W. Aw, T.H. Lee, S. Mutusamy, N.T. Hao, V.T. Quang, G.W. Canonica, et al., Selecting optimal second-generation antihistamines for allergic rhinitis and urticaria in Asia, *Clin. Mol. Allergy* 15 (1) (2017) 19.
- [6] B. Aggarwal, S. Shantakumar, D. Hinds, A. Mulgirigama, Asia-pacific survey of physicians on asthma and allergic rhinitis (aspair): physician beliefs and practices about diagnosis, assessment, and treatment of coexistent disease, *J. Asthma Allergy* 11 (2018) 293.
- [7] P. Small, P.K. Keith, H. Kim, Allergic rhinitis, *Allergy Asthma Clin. Immunol.* 14 (2) (2018) 51.
- [8] M.S. Blaiss, E. Hammerby, S. Robinson, T. Kennedy-Martin, S. Buchs, The burden of allergic rhinitis and allergic rhinoconjunctivitis on adolescents: a literature review, *Ann. Allergy Asthma Immunol.* 121 (1) (2018) 43–52.
- [9] S.J. Lee, E.K. Ha, H.M. Jee, K.S. Lee, S.W. Lee, M. Kim, D.H. Kim, Y.-H. Jung, Y. H. Sheen, M.S. Sung, et al., Prevalence and risk factors of urticaria with a focus on chronic urticaria in children, *Allergy Asthma Immunol. Res.* 9 (3) (2017) 212–219.
- [10] R.G. Hamilton, Assessment of human allergic diseases. *Clinical Immunology*, Elsevier, 2019, pp. 1283–1295.
- [11] A. Haccuria, A. Van Muylem, A. Malinovschi, V. Doan, A. Michils, Small airways dysfunction: the link between allergic rhinitis and allergic asthma, *Eur. Respir. J.* 51 (2) (2018).
- [12] L. Souza-Pereira, N. Pombo, S. Ouhbi, V. Felizardo, N. Garcia, Clinical decision support systems for chronic diseases: a systematic literature review, *Comput. Methods Programs Biomed.* (2020) 105565.
- [13] A. Ahmadi, M. Kashefi, H. Shahrokh, M.A. Nazari, Computer aided diagnosis system using deep convolutional neural networks for adhd subtypes, *Biomed. Signal Process. Control* 63 (2021) 102227.
- [14] S. Pathan, P. Kumar, R.M. Pai, S.V. Bhandary, Automated segmentation and classification of retinal features for glaucoma diagnosis, *Biomed. Signal Process. Control* 63 (2021) 102244.
- [15] A. Alahmar, M.E. Crupi, R. Benlamri, Ontological framework for standardizing and digitizing clinical pathways in healthcare information systems, *Comput. Methods Programs Biomed.* (2020) 105559.
- [16] B.H. Tozlu, C. Şimşek, O. Aydemir, Y. Karavelioglu, A high performance electronic nose system for the recognition of myocardial infarction and coronary artery diseases, *Biomed. Signal Process. Control* 64 (2021) 102247.
- [17] R. Xie, I. Khalil, S. Badsha, M. Atiquzzaman, An intelligent healthcare system with data priority based on multi vital biosignals, *Comput. Methods Programs Biomed.* 185 (2020) 105126.
- [18] J.J. Christopher, K.H. Nehemiah, K. Arputharaj, Knowledge-based systems and interestingness measures: analysis with clinical datasets, *J. Comput. Inform. Technol.* 24 (1) (2016) 65–78.
- [19] B. Jana, K. Oswal, S. Mitra, G. Saha, S. Banerjee, Detection of peripheral arterial disease using Doppler spectrogram based expert system for point-of-care applications, *Biomed. Signal Process. Control* 54 (2019) 101599.
- [20] A. Badnjević, L.G. Pokvić, M. Hasićić, L. Bandić, Z. Mašetić, Ž. Kovačević, J. Kevrić, L. Peccia, Evidence-based clinical engineering: machine learning algorithms for prediction of defibrillator performance, *Biomed. Signal Process. Control* 54 (2019) 101629.
- [21] W. Zhang, J. Han, S. Deng, Abnormal heart sound detection using temporal quasi-periodic features and long short-term memory without segmentation, *Biomed. Signal Process. Control* 53 (2019) 101560.
- [22] I. Bonavita, X. Rafael-Palou, M. Ceresa, G. Piella, V. Ribas, M.A.G. Ballester, Integration of convolutional neural networks for pulmonary nodule malignancy assessment in a lung cancer classification pipeline, *Comput. Methods Programs Biomed.* 185 (2020) 105172.
- [23] S.P. Pawar, S.N. Talbar, Lungseg-net: Lung field segmentation using generative adversarial network, *Biomed. Signal Process. Control* 64 (2021) 102296.
- [24] D. Oprić, A.D. Stankovich, A. Nenadović, S. Kovačević, D.D. Obradović, S. de Luka, J. Nesović-Ostojić, J. Milašin, A.Z. Ilić, A.M. Trbović, Fractal analysis tools for early assessment of liver inflammation induced by chronic consumption of linseed, palm and sunflower oils, *Biomed. Signal Process. Control* 61 (2020) 101959.
- [25] B. Navaneeth, M. Suchetha, A dynamic pooling based convolutional neural network approach to detect chronic kidney disease, *Biomed. Signal Process. Control* 62 (2020) 102068.
- [26] M.A. Alzubaidi, M. Otoom, A comprehensive study on feature types for osteoporosis classification in dental panoramic radiographs, *Comput. Methods Programs Biomed.* 188 (2020) 105301.
- [27] D. Wang, N. Pang, Y. Wang, H. Zhao, Unlabeled skin lesion classification by self-supervised topology clustering network, *Biomed. Signal Process. Control* 66 (2021) 102428.

- [28] D.M. Camacho, K.M. Collins, R.K. Powers, J.C. Costello, J.J. Collins, Next-generation machine learning for biological networks, *Cell* 173 (7) (2018) 1581–1592.
- [29] S.A. Dudani, The distance-weighted k-nearest-neighbor rule, *IEEE Trans. Syst. Man Cybern.* (4) (1976) 325–327.
- [30] M. Zareapoor, P. Shamsolmoali, D.K. Jain, H. Wang, J. Yang, Kernelized support vector machine with deep learning: an efficient approach for extreme multiclass dataset, *Pattern Recogn. Lett.* 115 (2018) 4–13.
- [31] S.R. Safavian, D. Landgrebe, A survey of decision tree classifier methodology, *IEEE Trans. Syst. Man Cybern.* 21 (3) (1991) 660–674.
- [32] S. Shamuganathan, Artificial neural network modelling: an introduction. Artificial neural network modelling, Springer, 2016, pp. 1–14.
- [33] G. Biau, E. Scornet, A random forest guided tour, *Test* 25 (2) (2016) 197–227.
- [34] A.B. Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. García, S. Gil-López, D. Molina, R. Benjamins, et al., Explainable artificial intelligence (xai): concepts, taxonomies, opportunities and challenges toward responsible ai, *Inform. Fusion* 58 (2020) 82–115.
- [35] P. Demoly, H. Chabane, J.-F. Fontaine, D. de Boissieu, D. Ryan, E. Angier, J. Just, Development of algorithms for the diagnosis and management of acute allergy in primary practice, *World Allergy Organ. J.* 12 (3) (2019) 100022.
- [36] P.W. Hellings, B. Pugin, G. Mariën, C. Bachert, C. Breynaert, D. Bullens, J. Ceuppens, G. Clement, T. Cox, D. Ebo, et al., Stepwise approach towards adoption of allergen immunotherapy for allergic rhinitis and asthma patients in daily practice in Belgium: a belsaci-abeforcal-euforea statement, *Clin. Transl. Allergy* 9 (1) (2019) 1.
- [37] J.M. Portnoy, A. Pandya, M. Waller, T. Elliott, Telemedicine and emerging technologies for health care in allergy/immunology, *J. Allergy Clin. Immunol.* 145 (2) (2020) 445–454.
- [38] J. Bousquet, H.J. Schünemann, P.W. Hellings, S. Arnavielhe, C. Bachert, A. Bedbrook, K.-C. Bergmann, S. Bosnic-Anticevich, J. Brozek, M. Calderon, et al., Macvia clinical decision algorithm in adolescents and adults with allergic rhinitis, *J. Allergy Clin. Immunol.* 138 (2) (2016) 367–374.
- [39] D. Caimmi, N. Baiz, L. Tanno, P. Demoly, S. Arnavielhe, R. Murray, A. Bedbrook, K. Bergmann, G. De Vries, W. Fokkens, et al., Validation of the mask-rhinitis visual analogue scale on smartphone screens to assess allergic rhinitis control, *Clin. Exp. Allergy* 47 (12) (2017) 1526–1533.
- [40] Y.M. Chae, S.H. Ho, C.S. Hong, C.W. Kim, Comparison of alternative knowledge model for the diagnosis of asthma, *Expert Syst. Appl.* 11 (4) (1996) 423–429.
- [41] J.J. Christopher, H.K. Nehemiah, A. Kannan, A clinical decision support system for diagnosis of allergic rhinitis based on intradermal skin tests, *Comput. Biol. Med.* 65 (2015) 76–84.
- [42] J.J. Christopher, H.K. Nehemiah, K. Arputharaj, G.L. Moses, Computer-assisted medical decision-making system for diagnosis of urticaria, *MDM Policy Pract.* 1 (1) (2016), p. 2381468316677752.
- [43] C.C. Aggarwal, Data Mining: The Textbook, Springer, 2015.
- [44] S. Kotsiantis, D. Kanellopoulos, P. Pintelas, Data preprocessing for supervised learning, *Int. J. Comput. Sci.* 1 (2) (2006) 111–117.
- [45] K. Bhaskaran, L. Smeeth, What is the difference between missing completely at random and missing at random? *Int. J. Epidemiol.* 43 (4) (2014) 1336–1339.
- [46] E.R. Fernandes, A.C. de Carvalho, X. Yao, Ensemble of classifiers based on multiobjective genetic sampling for imbalanced data, *IEEE Trans. Knowl. Data Eng.* 32 (6) (2019) 1104–1115.
- [47] F. Chartre, A.J. Rivera, M.J. del Jesus, F. Herrera, Remedial-hwr: tackling multilabel imbalance through label decoupling and data resampling hybridization, *Neurocomputing* 326 (2019) 110–122.
- [48] A. Gosain, S. Sardana, Handling class imbalance problem using oversampling techniques: a review. 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), IEEE, 2017, pp. 79–85.
- [49] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, Smote: synthetic minority over-sampling technique, *J. Artif. Intell. Res.* 16 (2002) 321–357.
- [50] S. Beguería, Validation and evaluation of predictive models in hazard assessment and risk management, *Nat. Haz.* 37 (3) (2006) 315–329.
- [51] M. Kubat, An Introduction to Machine Learning, Springer, 2017.
- [52] M. Kuhn, A Short Introduction to the Caret Package, vol. 1, R Found Stat Comput, 2015.
- [53] R. Polikar, Ensemble learning. Ensemble Machine Learning, Springer, 2012, pp. 1–34.
- [54] J. Quinlan, Bagging, Boosting, and c4.5, 1996.
- [55] E. Alfaro, M. Gámez, N. García, et al., adabag: an r package for classification with boosting and bagging, *J. Stat. Softw.* 54 (2) (2013) 1–35.
- [56] R. Kohavi, et al., A study of cross-validation and bootstrap for accuracy estimation and model selection, *IJCAI* 14 (1995) 1137–1145. Montreal, Canada.
- [57] H. Deng, Interpreting tree ensembles with intrees, *Int. J. Data Sci. Anal.* 7 (4) (2019) 277–287.
- [58] M. Sokolova, G. Lapalme, A systematic analysis of performance measures for classification tasks, *Inform. Process. Manag.* 45 (4) (2009) 427–437.
- [59] T. Fawcett, An introduction to roc analysis, *Pattern Recogn. Lett.* 27 (8) (2006) 861–874.
- [60] P. Branco, R.P. Ribeiro, L. Torgo, Ubl: An R Package for Utility-Based Learning, 2016. arXiv:1604.08079.
- [61] H. Wickham, W. Chang, M.H. Wickham, Package ‘ggplot2’. Create Elegant Data Visualisations Using the Grammar of Graphics. Version vol. 1, no. 2, 2016, pp. 1–189.
- [62] H. Wickham, W. Chang, devtools: Tools to Make Developing R Packages Easier, 2016. R Package Version. Vol. 1, 2019, 103, <https://CRAN.R-project.org/package=devtools>.
- [63] D.B. Dahl, D. Scott, C. Roosen, A. Magnusson, J. Swinton, A. Shah, A. Henningsen, B. Puetz, B. Pfaff, C. Agostinelli, et al., Package ‘xtable’: Export Tables to Latex or HTML, 2019. <https://CRAN.R-project.org/package=xtable>.