iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 11 **Issue:** VI **Month of publication:** June 2023

DOI: https://doi.org/10.22214/ijraset.2023.53544

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VI Jun 2023- Available at www.ijraset.com

Animal Disease Prediction using Machine Learning Techniques

Sana Rehman[1], Bhanushikha Rathore[2], Mr.Roshan Lal[3]
[1, 2, 3]*Computer Science & Engineering Amity University Uttar Pradesh, India*

Abstract: ***A great uprising has been witnessed in numerous Animal Diseases in the past many years. Many of these diseases have the tendency to transform into zoonotic diseases which can turn out to be very infectious and impact both animals as well as humans. Machine Learning is the field of study that deals with making machines/computers learn on their own so that further predictions can be made for varied Applications. Human disease detection using Machine Learning Techniques has been there from quite a while but very few advancements have been made for Animal Diseases. Through this research paper we make a new contribution in the aforementioned field by deploying ML techniques to classify certain Animal Diseases along with predicting the spread of the disease. Animal disease when turn into Zoonosis can have a huge scale impact on both Human and Animal species. So, through this project we also used certain techniques to predict if the disease is zoonotic or not.***
Keywords: Zoonosis, Machine Learning, Regression, Classification, Research.

I. INTRODUCTION

Within the past so many years we have witnessed a great uprising in the spread of numerous infectious diseases both in animals and humans. In addition to that, spread of zoonotic diseases also serves as a great threat to both the species. The outbreak of COVID-19 is a perfect example of how can zoonotic diseases can leave such a large-scale devastating impact on the world. Apart from COVID, diseases like bird flu and swine fever have caused so many deaths in past years. With all the aforementioned reasons, it has become imperial for Healthcare Industry to deploy systems that can predict the outbreak of such deadly diseases so that we can be prepared for any pandemic like situation or at least have strong combat mechanism to fight against them.

Machine learning algorithms have become effective tools for understanding and forecasting zoonotic diseases, which are illnesses that can spread from animals to people. These methods make use of the enormous amounts of data that are accessible from numerous sources, including records of animal and human health, environmental conditions, and genetic information. Machine learning algorithms can find patterns and relationships in these complicated datasets that would not be seen using conventional statistical techniques. Through this paper we attempt to use various Machine Learning techniques in order to predict the outbreaks of such diseases as well as use traditional ML algorithms to classify animal diseases as per their symptoms. There are several models that help in predicting chronic diseases in human like Heart Diseases, Diabetes etcetera, however very less development has been made for animal diseases in this area. But, especially after COVID has struck the world its high time that we start monitoring Animal Diseases that have the tendency to transform into zoonosis.

II. LITERATURE REVIEW

In paper, 'Ensemble Approach for Zoonotic Disease Prediction Using Machine Learning Techniques' Authors: Rama Krishna Singh and Vikash Chandra Sharma compares the efficiency of traditional techniques with that of Machine Learning Techniques in predicting the impact of Zoonotic Diseases. [1] In another paper, authors compare two important Cluster Analysis techniques namely ANN and K-means clustering for Dairy Cattle breed. [2] Authors talks about how Machine Learning can help in animal epidemic situation, followed by application of machine learning in animal disease analysis and prediction. [3] Paper points out how farm animal movement can be an influential factor for the spread of disease in animals at a faster rate also since the same animals are used as means of food for humans it is a possibility the infections can transform into zoonosis. The paper uses techniques like random forest to compute the probability of swine movements in two regions. [4]

A. Research Methodology

The aim through this work of research is to first carry out an in-depth analysis of the gathered data so as to draw meaningful insights from it and further use them in next phase of the research. Secondly, we will deploy machine learning techniques to perform the following tasks:

1) Firstly, in order to justify our project, we wanted to investigate on the fact that how many Animals all overthe world are dying by a particular disease. If the relation between number of cases and number of deaths is linear, this means that we need to have morerobust practices regarding sustaining Animal's health. So, we used Linear regression to predict the number ofanimal deaths caused by the specified disease
2) The second most important aim of our project is to monitor the diseases which has the tendency to transform into zoonotic. As a part of this, we first will drive a comparative analysis between two important techniques to predict human deaths as per the affectedcases to measure the severity of zoonosis in those diseases.
3) After obtaining knowledge on what all diseases have tendency to be zoonotic, we will check that which species of animal is most likely to spread diseases to humans. For this purpose, we have introducedXGBosst algorithm which is discussed in detail in thefurther sections of this paper.
4) To expand our project scope, we further have taken a new dataset which was extracted from WOAH (World organisation of animal Health). Further we used the dataset to classify the Animal Disease on the basis of the symptoms observed in them. Logistic Regression,Naïve Byes, Decision tree, Random Forest, SVM,ANN were the few techniques used for this purpose.
5) Lastly, we will use CNN to predict if the disease is zoonotic or not on the basis of symptoms of the disease.

III. ABOUT THE DATASET

The first dataset is gathered from EMPRES Global Animal Disease Information System which is run by food and agricultural organization of united nations. It maintains the record of various animal diseases and its distribution across the world. The dataset contains 24 features in total where disease, number of cases and deaths are few of them.

The second dataset was extracted form WOAH (world organisation of animal health) which has over 17000 entries and contains the records of over 63 diseases observed acrossdifferent species of animals. This dataset contains 38 columnsin total which corresponds to the varied features of dataset. 35of these features represents different symptoms observed fordifferent diseases, they are categorical in nature i.e., if 'x' disease has 'y' symptom, so the value will be '1' in that particular cell for the corresponding disease and value will be'0', if 'y' not a symptom of 'x'. The rest three columns containdetails regarding disease name, species in which disease is found and if disease is zoonotic or not respectively.

A. Data Exploration

Before implementing ML Models, we explored the datasetto understand the parameters and figure out some meaningfultrends in the dataset.

1) Firstly, we studied the corelation between different variables using corelation matrix and scatter plots forboth datasets.
2) Next, we observed the top 4 most common diseases interms of the number of cases observed:

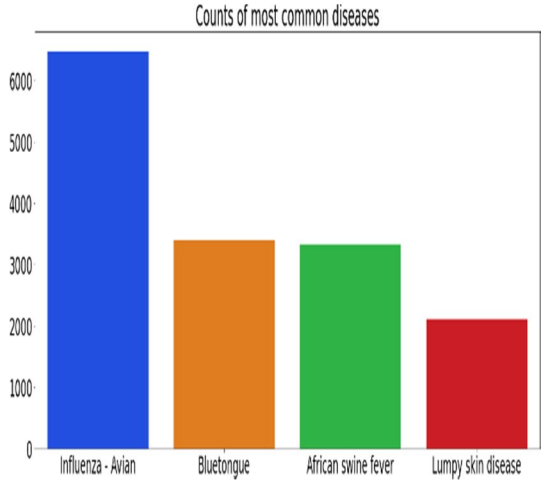


Fig. 2

3) We then observed the top 4 species that are most affected by the diseases worldwide:

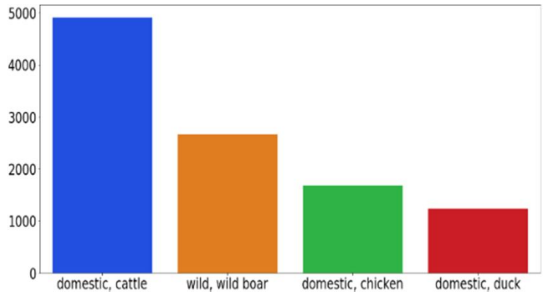


Fig. 3

4) Age distribution of humans affected by zoonoticdiseases:

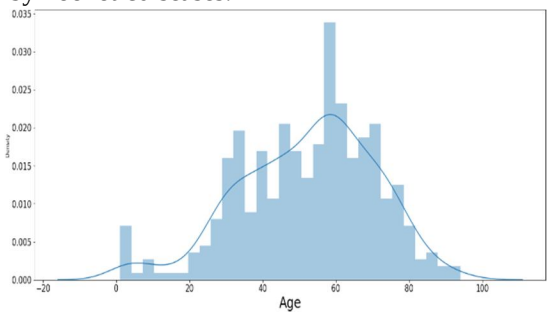


Fig. 4

IV. ML MODELS

Machine Learning is the field of study which deals with making computers learn and perform tasks without explicitly programming them. When it comes to building a ML model, it basically consists of two phases training and testing. In training phase, we make the machine learn using training data, once trained, the model is tested against the new data so as to measure its performance. Machine learning is further broadly categorised into supervised and unsupervised. In supervised machine learning the model is trained using labelled dataset where as in unsupervised the model itself learns by recognising patterns in the data.

Below is the representation of how the models will be built and flow chart of entire process.

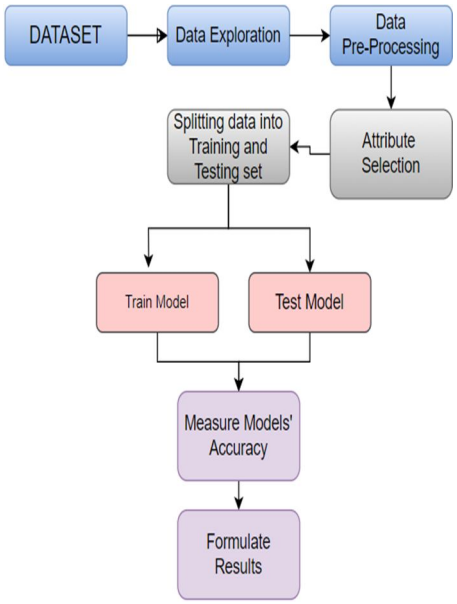


Fig. 5

In the upcoming section we will discuss all the ML models that have been used as a part of our Research.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 11 Issue VI Jun 2023- Available at www.ijraset.com

A. Linear Regression

It is one of the most important machine learning algorithms which is used to perform prediction functions. Linear regression deals with continuous form of data. Linear regression basically studies and observes the relationship between two different variables namely dependent or target variable i.e., the parameter that we predict using the independent variable. This relationship is linear in nature i.e., the graph has a straight line. The variables can either be positively linearly related or negatively linearly related. The equation of this straight line is given as follows:

$$Y=mx+c \quad (1)$$

Where the target variable is 'Y' and the independent variable is 'x' , 'c' is the intercept of line and 'm' is the slope of line.
The aim of linear regression is to find a regression line which best fits the data such that the difference between actual values and predicted values is as minimum as possible. This aforementioned task is performed by using a cost function which measures the accuracy that how well the input variables are mapped to the output variable.

B. Logistic Regression

It is another most important supervised machine learning algorithm. Unlike Linear Regression, logistic regression deals with categorical kind of data. The target variable has to be categorical in nature i.e., it should attain a discrete value like'Yes' and 'No' or '0' and '1'. Logistic regression makes useof Sigmoid function to predict the values of output variable and map those values between '0' and '1' so as to classify which class the output belongs to. In multinomial and ordinallogistic regression, we can have more than 2 classes as well.

C. Support Vector Machine(SVM)

Support Vector Machine (SVMs) is a machine learning algorithm which is commonly used in problems of classification and regression. The aim of SVM is to find a hyperplane that best divides the data into number of classes. For binary classification, SVM tries to find a general plane th at aligns the edges of the two classes. The margin is defined as the distance between the hyperplane and the nearest pointin each class.. The SVM equation is:
Given a set of training data: {(x1, y1), (x2, y2), ... , (xn, yn)}where xi is the input data and yi is the class label (+1 or -1):

1) First, transform the input data to a high dimensional space using the non-linear mapping function φ(x) so that the data is hyperplane separable.
2) Next, search a hyperplane that maximizes the distance in between the two classes. This can be intimated as the following optimization problem:
   Maximize ρ considering
   yi (w^T φ(xi) + b) ≥ ρ
   for all i=1,2,3...,n ||w||. = 1 where w is the weight vector, b is the bias, ρ is the margin and ||w|| is the norm of w.
3) Once we have found the optimal hyperplane, we canuse it to estimate the class of new data points by computing the sign of the function

$$f(x) = w^T \varphi(x) + b. \quad (2)$$

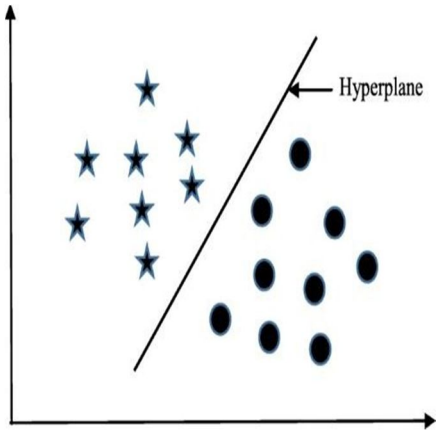


Fig. 6

D. Decission Tree

Decision trees are a well-known machine learning algorithm utilized for classification and regression purposes. A decision tree embodies a hierarchical structure wherein every internal node signifies a property test, each branch symbolizes a test result, and every leaf node signifies a class label or numerical value.

The basic idea behind decision trees is to split the data recursively depending on the values of the input features until all data points belong to the same class or yield pure nodes with the same numeric value. is. The tree is built top-down, starting at the root node and partitioning the data at each internal node based on features that maximize class separation.

Here are the steps involved in constructing a decision tree:

1) *Choose the function that works best for you:* The first step is to choose the best feature that divides the data into the purest subsets. This is done by computing the information gain or Gini index for each feature, which measures the amount of entropy or contamination in the data before and after splitting.
2) *Split the data: Once the best features are selected, the data are divided into sub-sets based on the values of the selected features. Each subset is then used to create a new branch of the tree.*
3) *Appeal:* The mentioned steps are repeated recursively for each subset until the stopping criteria are met. This criterion can be the maximum tree depth, the minimum number of data points per leaf node, or the minimum information gained.
4) *Forecast:* To predict the class or numeric value of a new data point, start at the root node and traverse the tree based on the values of the input features until you reach the leaf nodes. Classes or values associated with leaf nodes are then used as predictions.

Decision trees can steer both categorical and numeric data and can be utilized for both classification and regression tasks. They are easy to interpret and can capture complex decision boundaries. However, decision trees are prone to overfitting if not properly pruned, where small changes in data can lead to large changes in tree structure.

E. Random Forest

Random forest is an ensemble architecture model that enhances prediction accuracy and resilience by amalgamating multiple decision trees. In a random forest, numerous decision trees are trained and deployed and the ultimate prediction is derived by averaging the predictions of all the trees. Here are the steps involved in training a RandomForest:

1) Bootstrap example: Firstly, bootstrap samples are extracted from the original training data using a surrogate. This means that some data points in the bootstrap sample may appear multiple times and some may be omitted.
2) Random feature selection: For each tree, a random subset of features is chosen. This is done to ensure that each tree contains a diverse set of features and to reduce correlations between trees.
3) Train the decision tree. A decision tree is trained on the bootstrap samples using the selected traits. The tree grows until stopping criteria such as maximum tree depth or minimum number of data points per leaf node are met.
4) Repeat above 3 steps several times to form a forest of decision trees.
5) Forecast: To make assumptions for new data points, all trees in the forest are used to make predictions, and the last prediction is obtained by averaging the predictions of all trees. The Random Forest algorithm is a modification of the standard Decision Tree algorithm. The splitting criterion for each node in the tree is estimated based on the Gini impurity or entropy of the data.

The splitting criterion for each node in the tree is intended on the Gini impurity or entropy of the data.

The Gini entropy is calculated using the following equations: Gini impurity:

$$Gini(p) = 1 - \Sigma (pi\hat{}2) \quad (3)$$

where p is the vector of class probabilities, and pi is the probability of class i.

Entropy:

$$H(p) = - \Sigma (pi \log2(pi)) \quad (4)$$

where p is the vector of class probabilities, and pi is the probability of class i.

Random forests typically perform better than single decision trees because they are less prone to overfitting and can capture more complex decision boundaries.

F. Naive Bayes

Naive Bayes is a probabilistic machine learning algorithm that is efficiently used for purpose of classification . It utilizes on Bayes' theorem, which establishes that the probability of a hypothesis (class) given evidence (input characteristics) is proportionate to the probability of the evidence given the hypothesis, multiplied by the prior probability of the hypothesis.

The "naive" term means that the input features are conditionally independent when given the class labels.
Here are the steps involved in training a Naive Bayes classifier:

1) Compute prior probabilities. The prior probability for each class is computed as the percentage of training examples belonging to that class.
2) Calculate conditional probabilities. The conditional probability for each feature given a class is computed as the proportion of training examples that have that feature and belong to that class divided by the proportion of training examples.
3) Forecast: To generate a prediction for a new data point, the algorithm calculates the probability of each class given the evidence (input features) by utilizing Bayes' theorem. Subsequently, the prediction is made by selecting the class with the highest probability among all the classes.

The Naive Bayes algorithm is commonly used with text data. The input features are typically word frequencies in the document. In this case the algorithm is called Multinomial Naive Bayes and the conditional probabilities are computed using the multinomial distribution

G. XG Boost

XG Boost is an implemented of the Gradient Boosted Decision Trees algorithm.
The algorithm works by going over cycles that builds a new model after end of every cycle and then all these models are combined together to form an ensemble model. Before the cycle we take a base model 'x' which is naïve in nature and use it to make predictions. The predictions can be incorrect in nature since model is not that robust yet. Then this base model is fetched into the boost cycle where it further calculates errors made in prediction for each observation. Then a new model 'y' also called 'Error predicting model' is built which is used to predict the errors that was calculated by model 'x'. Further the predictions made by model 'y' are then added to the ensemble of models. This process keeps repeating by making use of previous predictions in calculating new errors, building a new error-predicting model, and then, adding it to the ensemble model. A more detail version on implementation of this algorithm is explained in further sections of this paper.

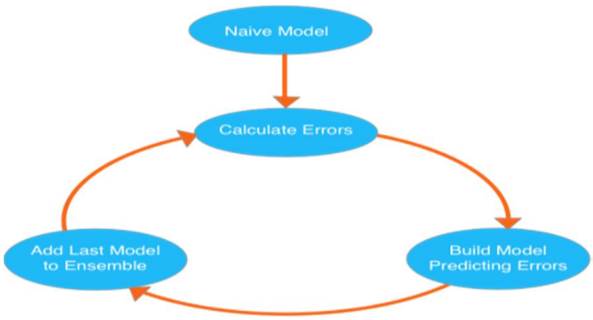


Fig. 7

H. ANN (Artificial Neural Network)

Artificial neural network (ANN) is an approach through which we try to simulate the working of a human brain. Just like biological cells that exists in human brain, ANN contains a network of artificial neurons that gives machines the capability to think like human and make decisions like humans. Just like humans who make some decision on the basis of studying the inputs properly and then giving some output, ANN also studies the variable relationship between input and outputs and then makes some decision. ANN can be taken as weighted directed graph where the artificial neurons correspond to the nodes of graph and the weighted edges represents the relation between the input and the output.
Further, ANN has a varied architecture where the execution is spread among different layers of ANN:

1) Input Layer: Layer is responsible for fetching the input in different formats from some external source and then these inputs are mathematically given a notation like x(n) for every n number of input.
2) Hidden Layer: The hidden layer is actually responsible for the computational part of ANN. As a part of it, first the inputs are taken from previous layers and they are multiplied with the corresponding weights of the edges. This computation is done for every 'n' number of input and then all the results are added to give the weighted sum of all inputs plus some bias.

$$\sum_{i=1}^{n} W(i) * X(i) + b \qquad (5)$$

This sum if turns out to be '0' then a certain bias is introducedinto the total weighted inputs to make it non-zero. The total weighted inputs value can range over 0 to +ve infinity but some threshold value is set so that the output comes as desired. For this purpose, we pass our total weighted inputs through some activation function whose job is to transfer theinput to desired output by choosing the nodes which makes to the output layer by firing them.

3) Output Layer: The input after going through a series of computations in thehidden layer finally results into an output, the nodes which gets fired by the activation layer reaches the output layer.

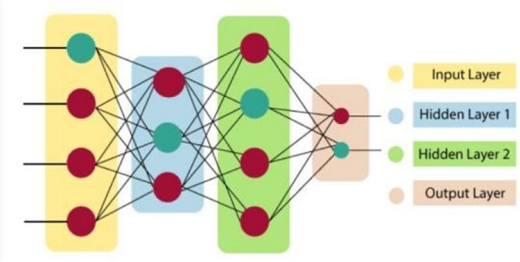


Fig. 8

I. Convolutional Neural Networks

Convolutional Neural Network is again a type of feed- forward neural network where it contains some additional layers as compared to the traditional neural network. For instance, these layers are:

1) Input layer

2) Convolution Layers: This is the major part of CNN where the input is scanned and then passed through some non-linear activation function for generating output. This output is fed to the next convolution layer until all are completed.

3) Pooling Layer: This layer is basically used to reducethe dimensionality of the input images.

4) Dense layer: Neurons in this layer takes input from neurons from all previous layers.

5) Output Layer

MLP uses backpropagation to train the input nodes, this is done in order to reduce the value of cost function by iteratively adjusting weights, which ultimately helps in minimizing cost function.

Backpropagation starts by forwarding the weighted sums through the layers and then gradient is calculated of cost function which is nothing but mean squared error for corresponding input-output pairs. After that, while propagating backwards the weights of first hidden layer are updated by the gradient value computed earlier, and this how this process keeps on propagating backwards until the starting point of network. This process stops when the gradient value has not changed much from the previous one.

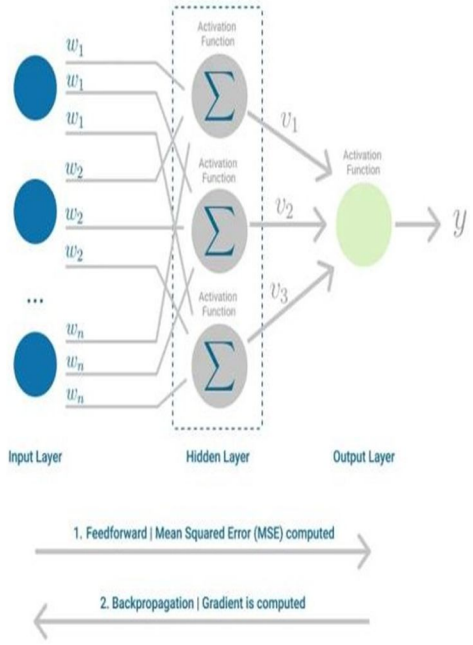


Fig. 9

V. IMPLEMENTATION AND RESULTS

In earlier section of this paper, we laid down five different goals that we proposed to achieve through this research work. In the following sections the executions and results of these goals are described:

1) As discussed in the earlier section of this paper we used linear regression to predict the outbreak of animal deaths as per the cases observed. The independent variable is 'sumCases' i.e., the number of affected cases and the target variable is 'sumDeaths' i.e., the number of deaths due to the specified disease. The variables shared a positive relationship which is demonstrated by the following figure*:*

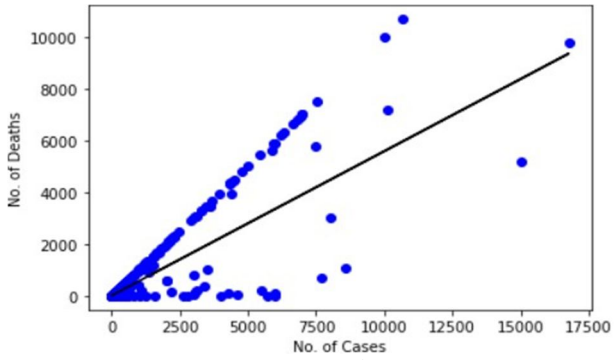


Fig. 10

The results of outbreak as well as the positive relationship between the animal and the cases observed stands for the factthat we need more robust practices when it comes to Animal health conditions all across the world because a huge numberof animal deaths are observed yearly.

2) Next We used Logistic regression and Support vector machine to predict human deaths as per the affected casesto measure the severity of zoonosis in those diseases. Theindependent variables were 'Humans Age' and 'HumansAffected' were extracted by performing PCA (Principal Component Analysis) on entire set of features and then finding the top two features that are most related with output or target variable 'Human Deaths', which further had integer values so we transformed these values and divided into two classes '0' (If no human died of the diseases) and '1' (if human deaths observed due to the diseases).

TABLE I. PERFORMANCE MEASURES: The below table conatainesthe performance summary of the above-mentioned algorithms and their specific purpose

Aim	Technique	Accuracy	Precision	Recall	F1-Score
Animal Deaths outbreak Prediction	Linear Regression	74%	-	-	-
Predict human deaths to measure severity of zoonosis	Logistic Regression	83%	82%	99%	90%
	Support Vector Machine (SVM)	96%	96%	100%	98%

In addition to this we drew a comparative analysis betweenLogistic Regression and SVM, the results are depicted as follows, Clearly SVM performed better than Logisticregression:

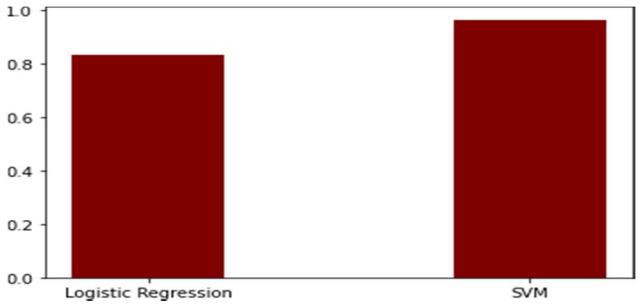


Fig. 11

3) Implementation of XGBoost to predict which outbreaks ofanimal diseases are more likely to get humans sick.

XGBoost algorithm can't just take a data-frame. It requires data to be in the form of matrix. So before executingXGBoost algorithm we first prepared data-frames in the formof DMatrix which were then fetched to the boosting algorithm.

Before preparing the data, we did some basic data cleaning:

a) Step-1: Data cleaning

First, we removed all the features from the dataset that contained information regarding the target variable. So, features like "humansGenderDesc", "humansAge", "humansAffected" and"humansDeaths" were dropped from the dataset.

Next, we created a column 'diseaseInfo_numeric', which contains Boolean values: '0' if humans were affected by the disease and '1' if humans not affected. Furthermore, all the unnecessary information like 'latitude', 'longitude', 'id' etcetera from the dataset was removed, as it won't help in accomplishing the goal.

humansAffected
TRUE
FALSE
FALSE
FALSE
FALSE
FALSE

Fig. 12

b) Step-2: Data-frames Preparation

We started by converting the features that were categorical in nature into numerical format. To serve this purpose, we used One-hot encoding method which creates separate column for each unique category, then each observation is tested against each column and that cell value is marked with '1' if that observation belongs to that column and '0' if it does not. The feature 'country' was converted into a hot-matrix 'region' using above method.

countryChina	countryFrance	countryIndonesia	countryMontenegro	countryRepublic of Korea	countryRussian Federation
0	1	0	0	0	0
0	0	1	0	0	0
0	0	0	0	1	0
0	0	0	1	0	0
0	0	0	0	0	1
1	0	0	0	0	0

Fig. 13

Similarly for feature 'speciesDescription' we created separate columns for each animal species and created a one-hot matrix named 'species', of different species that was listed under the 'speciesDescription' feature, this was done to discover that if some specific specie is more likely to spread disease to human. For instance, it can be a possibility that domestic species are more likely to transfer disease to human so we created a Boolean column 'isdomestic' and for each observation that containedthe keyword domestic under 'speciesDescription' init was marked '1' in 'is_domestic' column and '0' ifotherwise.

c) Step-3: Combining all Data-Frames

From the previous steps we prepared three separate data-frames namely: 'diseaseInfo_numeric', 'region' and 'species'. We will bundle all these data-frames and convert them into a matrix.

d) Step-4

Dividing dataset into two subsets which is training and testing.
We split the dataset into the ratio of 70:30 where 70is for training and 30 is for testing.

e) Step-5:

Converting the clean data -frame to Dmatrix because our data-frame contains binary set of values that are '0' and '1' and once converted to a matrix it will beknown as a sparse matrix, and Dmatrix helps to storeand access sparse matrix values more easily which will in turn improve model training process.

f) Step-6: Training the model

Three inputs that were provided for training themodel:

1) Training data: passing the data in the form of Dmatrix generated in previous step.
2) Number of training rounds: Number of training rounds means number of times the boosting cycle will go on and will keep adding models to the ensemble of modelsand hence improving the naïve model which was used as a base model for starting point.
3) Objective Function: This help in determining the results. In our casesince we are predicting something which is binary in nature (Humans get affected or not) so we have used 'binary logistic' function which is actually logistic regression for binary classification.

Now let's examine the results of our naïve or base model:

Table II. Performance Results for Naïve Model Used in XGBOOST

Error on training data	Number of training rounds	1	0.014698
		2	0.014698
Error on testing data	----	---	0.0121520972167777

As we can see no improvement was observed after second round of training. Now that we have build our base model andtested its performance, we can go ahead and tune or improvethe model by making some changes as discussed in further section.

g) Step-7: Tuning the Model

After we having our base model, we made an attempt to improve the performance of model by:

1) Avoiding overfitting: More the number of layers in decision trees more likely is the chance for the model to capture randomness in the data rather than important variation, which can land the model into overfitting problem, hence decreasing the depth of trees by keeping a smaller number of layersin gradient decision tress.
2) Avoid imbalance classes
3) Training for more rounds
4) Early stopping while training: this means that if no improvement is observed in model's performance after a certain number of rounds so, we will stop the training, again this is done to prevent overfitting our model.

XGBOOST MODEL 'M' WITH TRAINING ROUNDS 'N'=10
False_cases=sum(label =FALSE) #when no humans affected True_cases=sum(label=TRUE)#when humans are affected
While n≤10:
M.XGBoost(data->training data
Max.depth=3 #maximum depth of each decision treees_rounds=3 #if improvement is not observed in the
model's performance for these manyrounds then we'll stop training
objective_func= 'binary_logistic' scaling=False_cases/True_cases
#avoiding imbalance classes*)*
predicted ->pred(M)Err ->mean(test)

- Results

TABLE III. PERFORMANCE RESULTS FOR XGBOOST MODEL AFTER TWEEKING SOME PARAMETRES.

Error on training data	Number of training Rounds	1	0.016126
		2	0.014866
		3	0.014866
		4	0.014866
		5	0.014614
		6	0.014530
		7	0.014530
		8	0.014530
		9	0.014614
Error on testing data	----	--	0.012152097 2

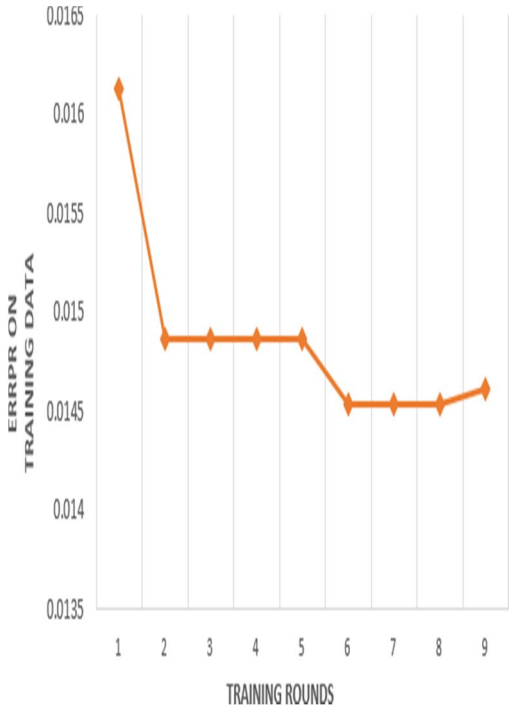


Fig. 14

- Observations:

In the first training round the error comes out to be higher as compared to the error in the first round of training for naïve model, this is because we decreased the depth of decision tress for the boostingrounds to avoid overfitting. From second training rounds onwards, we observe that as the number of training rounds increases the error decreases this is because more the number of training or boosting round more accurately the model captures the variation in the dataset. In the 9th round of training suddenly error increases because this might be the point that our model has started overfitting the data,so as discussed in the previous section we stopped the training rounds as soon as no improvement seenin the performance.

Final error on train data: 0.014530

Final error on test data : 0.0121520972

h) Step-8: Examining the Model

We can examine our model by stacking all thegradient Decision tress used on top of each other andpick up the feature that shows up the most in each node for every separate tree.

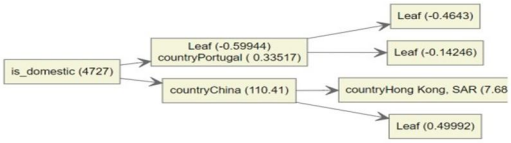


Fig. 15

The leftmost side of the picture is the top of the treeand the right-most is the bottom of tree. So, we canconclude that 'is_domestic' is the most important feature across all the trees in our ensemble model as it holds the highest position in the tree and also because the numeric value mentioned against thefeature which is the quality factor is also the highestwhen compared to the quality factor of other features. A visual representation of how important and informative the features were to determine if humans get affected by disease or not.

Fig. 16

The above plot shows that whether the affected animal was domestic or not was the most important factor to determine if human get affected by the same disease as well. Actually, it makes sense, because humans are more likely to get affected by certain zoonotic disease by coming in contact to a domestic animal rather than a wild or equine animal.

4) Animal Disease classification

With the help of second dataset that was collected from WOAH (World organisation of animal health), we used various models Logistic Regression, Naïve Byes, Decision tree, Random Forest, SVM, ANN to classify the diseases as per their symptoms.

In ANN (Artificial Neural network) one input layer , two hidden layers and one output layer is added. The activation function for the hidden layers used is 'ReLu', rectifying linear unit and the activation function used at output layer is 'softmax'.

TABLE IV.PERFORMANCE MEASURES: The below table contains the performance summary of the abovementioned algorithms inclassifying animal disease asper the observed symptoms

Technique	Accuracy	Precision	Recall	F1-Score
Logistic Regression	90.8%	91%	91%	91%
Decision Tree	91.1%	91%	91%	91%
Support Vector Machine (SVM)	91.13%	91%	91%	91%
Random Forest	90.9%	91%	91%	91%
Naïve Bayes	78.8%	76%	79%	75%
ANN	91.2%	91%	91%	91%

Visual representation of the above performance measures:

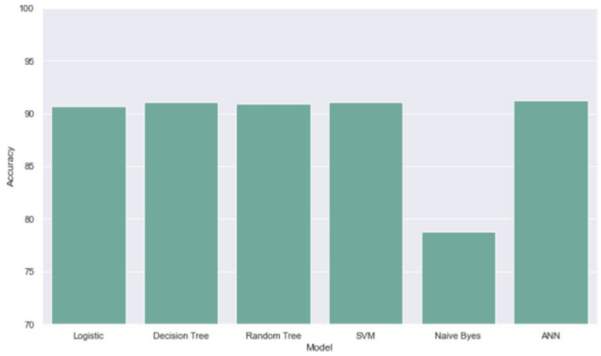


Fig. 17

As depicted in the above graph, the worst performance was shown by Naïve Bayes, this sis because naïve bayes is more suitable for classifying data into a smaller number of categories but here we are trying to classify 63 different diseases. ANN gave the best performance in classifying the animal disease followed by SVM, Decision Tree, Random Forest and logistic regression.

5) CNN to predict if Disease Zoonotic or not:

In the previous section of this paper, we used Logistic regression and SVM to predict the number of human deaths as per the affected cases to measure the severity of zoonosis in those particular diseases.

We propose a new model where we will predict if the diseaseis zoonotic or not on the basis of symptoms observed for thatdisease.

a) Proposed model

- Data pre-processing and cleaning: Before jumping to build a CNN model we first performed a LASSO (Least Absolute Shrinkage) regression technique on our dataset. LASSO help in selecting important features from the dataset and eliminating the parameters that are not important, this is done to increase the accuracy of the model and is achieved by shrinking the data towards a central point. The idea is to sub-sample our dataset and run LASSO on it multiple times. So, for 'n' number of times that LASSO is performed over 'm' number of sub-samples of data and 37 variables we will get a set of values like $\boldsymbol{yi}$=[$\boldsymbol{yi}$,**1** $\boldsymbol{yi}$,**2** $\boldsymbol{yi}$,**37**] where 'i' is the observation. So, for nay variable 'a' we count the number of observations for which the variable was non-zero and if this count is greater or equal to the threshold value manually set then that variable 'a' is selected for further analysis.
- CNN Model: With the help of CNN model, we will be predictingthe status of disease in regard to zoonosis. So, the prediction will be done for two classes that arezoonosis class: '1' and no-zoonosis class: '0'.

Below is the visual representation of the modelarchitecture:

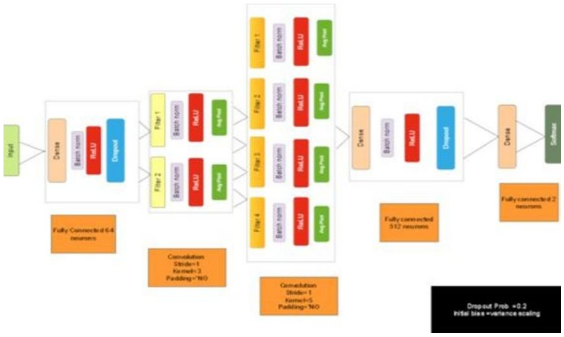


Fig. 17

The input layer accepts a '1D-Numerical' array which contains all the 37 features that were selectedthrough LASSO and Majority Voting process. So, the dimension of input layer will be Nx37 where 'N' is the number of training rounds or examples. The dense layer just after the input layer contains 64 neurons and it linearly combines the 37 variables with some bias factor. Further the activation functionrectified linear unit (ReLu) is used to perform the non-linear transformation of the input.

Also, a dropout factor is introduced in the model where 20%of data is dropped just to avoid overfitting scenario.After the first dense layer or the fully connected layerwe have two consecutive Convolution layers. Inconvolution layer1 we have applied two filters where kernel width=3 and stride=1. So, this first Convolution layer converts the output block of denselayer i.e. Nx64 to a tensor having dimensions Nx64x1. This tensor further undergoes batch-normalization, non-linear transformation by ReLu and average pooling to give an output tensor of dimension Nx31x2.

The filters in second convolution layers are applied with kernel width =5 and stride=1 and the output given by second convolution layers is tensor of dimension Nx13x4, this tensor will serve as an input to the next dense layer. The final output of the disease belonging to class 'zoonosis' or class 'no-zoonosis', is given at the end of SoftMax layer where categorical cross -entropy is used as the loss function. In the next section we can see the performance results shown by our model.

- Results of our model: Confusion matrix:

TABLE V. CONFUSIION MATRIX FOR CNN CLASSIFIER

Out of 6065 true non-zoonotic diseases the classifier was ableto predict all 6065 correctly. Out of 9732 true zoonotic diseases 9716 diseases were correctly predicted as zoonotic

Total Cohort		True Condition	
		Non-zoonotic	Zoonotic
Disease predicted	Non-zoonotic	True positive (TP) 6065	False Negative (FN) 0
	Zoonotic	False Positive (FP) 16	True Negative (TN) 9716

Observation: when it comes to diagnosing a medicalcondition, the false negatives should be less becauseif our model will predict that human is not affected when actually the human is affected by the disease, it will lead us into thinking that we are not affected hence no prevention would be taken which in turn can turn out to be very dangerous situations.

As we can see in the above results false negatives are zero which stands for the fact that our model hasdone a good job in predicting the disease as zoonoticif it actually is, due to this people can start taking precautions from early stage.

Some additional performance measures of the modelare listed below:

TABLE VI.ADDITIONAL PERFORMANCE MEASURE

TPR (total positive rate) = TP/TP+FN, Total negative Rate (TNR)=TN/TN+FP,

Accuracy= TP+TN/TP+TN+FP+FN, Precision =TP/TP+FP

Measure	Class Weight	Score
TPR%	2.9 : 1	100
TNR%	2.9 : 1	100
Accuracy%	2.9 : 1	99.8
Precision%	2.9 : 1	99.7
Train Accuracy %	2.9 : 1	94
Test Accuracy%	2.9 : 1	99.9
Training Loss	2.9 : 1	0.1515

VI. LIMITATIONS

There are a number of constraints that need to be takeninto account, despite the fact that machine learning approacheshave showed considerable promise in the prediction of zoonotic illnesses. First off, the reliability of predictions is largely dependent on the accessibility of data. The reliabilityof machine learning models can sometimes be impacted by theinadequate, inconsistent, or biased nature of zoonotic diseasedata. Additionally, due to differences in data formats and standards, it might be difficult to integrate data from many sources.

The complexity of zoonotic illnesses is another drawback. Thecomplex connections between animals, people, and the environment that these diseases entail make it challenging to include all important variables in a predictive model. These complexities are frequently simplified by machine learning models, which may leave out important factor.

Moreover, machine learning models are typically trained to make predictions based on trends seen in the past. But zoonotic illnesses might display unusual or uncommon patterns that have never been seen before, making it difficultfor models to precisely forecast such occurrences. This constraint becomes especially clear when addressing newly orrecently remerging zoonotic illnesses for which there is insufficient historical data.

VII. FUTUREWORK

Machine learning shows enormous promise for influencing the direction of preventive healthcare in the field of zoonotic disease prediction. The importance of early detection and preventative measures in lessening the effects of zoonotic illnesses is becoming increasingly clear as technology develops. Machine learning algorithms have the potential to completely change how we forecast and prevent certain diseases because of their capacity to analyse enormous volumes of data and spot trends. The integration of several datasets is a critical component of upcoming work in machine learning-based zoonotic disease prediction. Machine learning models can develop a thorough grasp of the intricate interactions between animal hosts, environmental conditions, human health data, and genetic information by incorporatingdata from a variety of sources.

The creation of sophisticated machine learning algorithms that can manage high-dimensional and heterogeneous data will also be crucial. For example, deep learning algorithms may extract subtle features and relationships from complicated datasets, allowing for more precise forecasts and a greater comprehension of the dynamics of zoonotic diseases

The development of machine learning-powered real-time surveillance systems will be another area of emphasis. Thesesystems have the capacity to continuously analyse data coming from sources including environmental sensors, animal monitoring devices, and health records for people and animals. These models can alert medical practitioners and policymakers to impending zoonotic disease outbreaks by spotting early warning indicators and aberrant patterns, enabling quick response and containment.

VIII. CONCLUSION

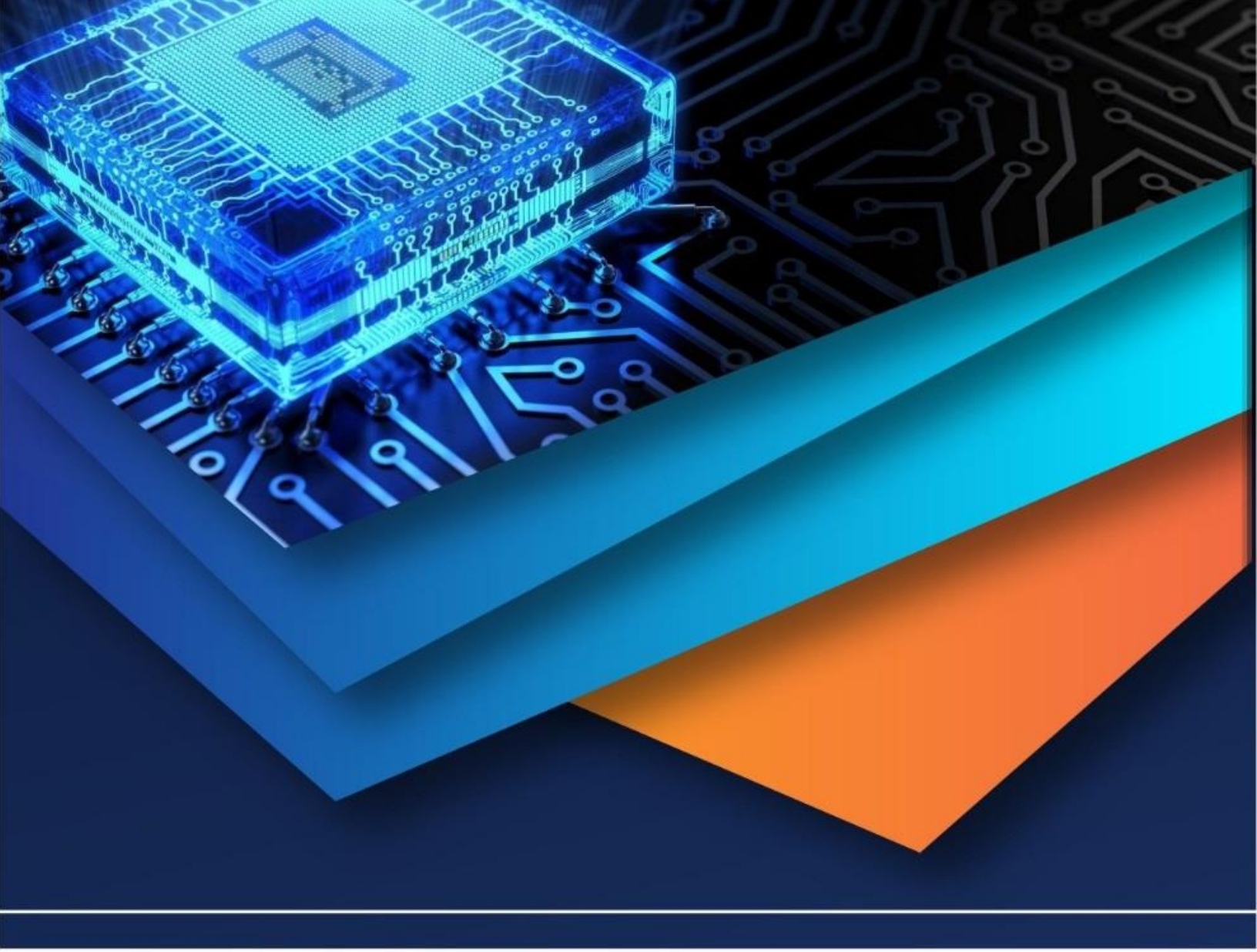
Machine Learning Techniques have proved to be useful in many areas. After COVID it was very imperial to have built certain systems which can predict the outbreak of such infectious diseases which not only has the tendency to threaten the livelihood of animal species but humans as well. With thehelp of the predictions made using ML techniques we can beprepared to combat any pandemic like situation that is likelyto become a great threat to living beings.

The observation of the overall speediness of outbreak detection indicates that the surveillance and detection systems, despite their distinct and separate nature, can potentially be more efficient than previously anticipated.However, in situations involving a highly contagious zoonotic disease, the importance of prompt detection, reporting, and response cannot be emphasized enough. Extra analysis, perhaps the usage of private information, may also be of fee to provide a higher understanding the reporting completeness of zoonotic illnesses in each human and animalpopulations.

REFERENCES

[1] Singh, R. K., & Sharma, V. C. (2015). Ensemble Approach forZoonotic Disease Prediction Using Machine Learning Techniques.
[2] Atıl, H., & Akıllı, A. (2016). Comparison of artificial neural network and K-means for clustering dairy cattle. International Journal of Sustainable Agricultural Management and Informatics, 2(1), 40-52.
[3] Zhang, S., Su, Q., & Chen, Q. (2021). Application of machine learning in animal disease analysis and prediction. Current Bioinformatics, 16(7), 972-982.
[4] Valdes-Donoso P, VanderWaal K, Jarvis LS, Wayne SR, Perez AM. Using Machine Learning to Predict Swine Movements within a Regional Program to Improve Control of Infectious Diseases in the US. Front Vet Sci. 2017 Jan 19;4:2. doi: 10.3389/fvets.2017.00002. PMID:28154817; PMCID: PMC5243845.
[5] Morota, G., Ventura, R. V., Silva, F. F., Koyama, M., & Fernando, S.
[6] C. (2018). Big data analytics and precision animal agriculture symposium: Machine learning and data mining advance predictive big data analysis in precision animal agriculture. Journal of animal science, 96(4), 1540-1550.
[7] Peters, D. P., McVey, D. S., Elias, E. H., Pelzel□McCluskey, A. M., Derner, J. D., Burruss, N. D., ... & Rodriguez, L. L. (2020). Big data– model integration and AI for vector□borne disease prediction. Ecosphere, 11(6), e03157.
[8] Corley, C. D., Pullum, L. L., Hartley, D. M., Benedum, C., Noonan, C., Rabinowitz, P. M., & Lancaster, M. J. (2014). Disease prediction models and operational readiness. PloS one, 9(3), e91989.
[9] Buza, T., Arick, M., Wang, H., & Peterson, D. G. (2014). Computational prediction of disease microRNAs in domestic animals. BMC Research Notes, 7(1), 1-13.
[10] Kasbohm, E., Fischer, S., Küntzel, A., Oertel, P., Bergmann, A., Trefz, P., ... & Köhler, H. (2017). Strategies for the identification of disease- related patterns of volatile organic compounds: prediction of paratuberculosis in an animal model using random forests. Journal of Breath Research, 11(4), 047105.

[11] Ortiz-Pelaez, Á., & Pfeiffer, D. U. (2008). Use of data mining techniques to investigate disease risk classification as a proxy for compromised biosecurity of cattle herds in Wales. BMC Veterinary Research, 4(1), 1-16.
[12] Li, X., Zhang, Z., Liang, B., Ye, F., & Gong, W. (2021). A review: Antimicrobial resistance data mining models and prediction methods study for pathogenic bacteria. The Journal of Antibiotics, 74(12), 838- 849.
[13] Huang, S., Cai, N., Pacheco, P. P., Narrandes, S., Wang, Y., & Xu, W. (2018). Applications of support vector machine (SVM) learning in cancer genomics. Cancer genomics & proteomics, 15(1), 41-51.
[14] Schuldt, C., Laptev, I., & Caputo, B. (2004, August). Recognizing human actions: a local SVM approach. In Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004. (Vol. 3, pp. 32-36). IEEE.
[15] Tsang, I. W., Kwok, J. T., Cheung, P. M., & Cristianini, N. (2005). Core vector machines: Fast SVM training on very large data sets. Journal of Machine Learning Research, 6(4).
[16] Salazar, D. A., Vélez, J. I., & Salazar, J. C. (2012). Comparison between SVM and logistic regression: Which one is better to discriminate?. Revista Colombiana de Estadística, 35(SPE2), 223-237.
[17] Musa, A. B. (2013). Comparative study on classification performance between support vector machine and logistic regression. International Journal of Machine Learning and Cybernetics, 4(1), 13-24.
[18] Sanson, R. L., Pfeiffer, D. U., & Morris, R. S. (1991). Geographic information systems: their application in animal disease control. Rev sci tech, 10(1), 179-195.
[19] De La Rocque, S., Rioux, J. A., & Slingenbergh, J. (2008). Climate change: effects on animal disease systems and implications for surveillance and control. Rev Sci Tech, 27(2), 339-54.
[20] Hästein, T., Hill, B. J., & Winton, J. R. (1999). Successful aquatic animal disease emergency. Rev. sci. tech. Off. int. Epiz, 18(1), 214- 227.
[21] Morgan, N., & Prakash, A. (2006). International livestock markets and the impact of animal disease. Rev Sci Tech, 25(2), 517-528.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY