Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Results in Engineering 13 (2022) 100363

Contents lists available at ScienceDirect

Results in Engineering

journal homepage: www.sciencedirect.com/journal/results-in-engineering

Modelling Covid-19 infections in Zambia using data mining techniques

Josephat Kalezhi [a,*], Mathews Chibuluma [b], Christopher Chembe [c], Victoria Chama [d], Francis Lungo [e], Douglas Kunda [f]

[a] Department of Computer Engineering, Copperbelt University, Kitwe, Zambia
[b] Department of Information Technology/Systems, Copperbelt University, Kitwe, Zambia
[c] National Institute for Public Administration, Lusaka, Zambia
[d] Department of Computer Science and Information Technology, Mulungushi University, Kabwe, Zambia
[e] School of Social Sciences, Mulungushi University, Kabwe, Zambia
[f] ZCAS University, Lusaka, Zambia

ARTICLE INFO

Keywords:
WEKA
J48 algorithm
Naïve Bayes
Multilayer perceptron
Coronavirus
COVID-19

ABSTRACT

The outbreak of Covid-19 pandemic has been declared a global health crisis by the World Health Organization since its emergence. Several researchers have proposed a number of techniques to understand how the pandemic affects the populations. Reported among these techniques are data mining models which have been successfully applied in a wide range of situations before the advent of Covid-19 pandemic. In this work, the researchers have applied a number of existing data mining methods (classifiers) available in the Waikato Environment for Knowledge Analysis (WEKA) machine learning library. WEKA was used to gain a better understanding on how the epidemic spread within Zambia. The classifiers used are J48 decision tree, Multilayer Perceptron and Naïve Bayes among others. The predictions of these techniques are compared against simpler classifiers and those reported in related works.

1. Introduction

The first case of Coronavirus was reported in 2019 in Wuhan Province of China believed to have cross-transmitted from animal to humans [1]. The virus is in the same family named Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2). The current Coronavirus disease was initially referred to as 2019 novel Coronavirus (2019-nCoV) but now code-named as COVID-19 [2]. The World Health Organization declared COVID-19 a pandemic in March 2020 [3]. Since then, relentless efforts have been put in place by several nations to understand the virus and how to contain the pandemic. The Scientific community has spent many efforts to come up with vaccines that could be used to curb the spread of the virus [4]. Nevertheless, since the admission of the first dose of vaccines, two more variants have emerged. The Delta variant which came with the third wave [5] and the Omicron variant which came with the fourth wave [6]. The efforts of the scientific community in understanding the spread of these different variants can be supplemented through the use of technology. One such field in technology that can be used to understand the spread of the virus is artificial intelligence through machine learning.

Machine learning models have been used in various fields to supplement expert's efforts in those fields to solve problems. For example, in agriculture, the efforts in Ref. [7] used machine learning in supporting soil classification and the works in Ref. [8] used machine learning techniques in classifying crops based on macronutrients and weather data. In health and specifically in the fight against COVID-19, machine learning has been used in various ways. For instance, the works in Refs. [9–11] show various ways in which machine learning can be used in the fight against COVID-19 starting from treatment, medication, screening, prediction, forecasting, drug/vaccine and contact tracing. Despite the privacy concerns associated with contact tracing, the works in Ref. [12] provide a socio-technical framework that can be used by government agencies to implement digital contact tracing.

In order to understand how the pandemic evolves, a number of models have been developed and applied. Among these are data-mining models that have been successfully applied in various problems. Car et al. [13] used Multilayer Perceptron data mining model to classify the spread of COVID-19 Infection. Decision tree data mining algorithm

* Corresponding author.
E-mail addresses: kalezhi@cbu.ac.zm (J. Kalezhi), mathews.chibuluma@cbu.ac.zm (M. Chibuluma), cchembe@edu.nipa.ac.zm (C. Chembe), vchama@mu.ac.zm (V. Chama), flungo@mu.ac.zm (F. Lungo), douglas.kunda@zcas.edu.zm (D. Kunda).

https://doi.org/10.1016/j.rineng.2022.100363
Received 29 September 2021; Received in revised form 8 January 2022; Accepted 1 February 2022
Available online 4 February 2022

showed more efficiency in predicting recovery of patients infected from COVID-19 disease [14]. The works in Ref. [15] compared various deep learning models in predicting COVID-19 infection. The efforts in Ref. [16] proposed the use of hybrid approach by combining adaptive network-based fuzzy inference system (ANFIS) and multi-layered perceptron-imperialist competitive algorithm (MLP-ICA) to predict time series of infected people and mortality rate in Hungary. Furthermore, Sujath et al. [17] used forecasting models such as linear regression, Multilayer perceptron and Vector autoregression to anticipate the epidemiological example of the ailment and pace of COVID-2019 cases in India.

In this work authors have used the Waikato Environment for Knowledge Analysis (WEKA) [18] software to analyze the Covid-19 data for Zambia as reported by the Ministry of Health. WEKA is an open-source machine learning library that has been applied to data mining problems successfully. WEKA encompasses a variety of machine learning algorithms and data preprocessing tools [19]. We use WEKA to compare classifiers using the data obtained from Ministry of Health. The classifiers used are J48 decision tree, Multilayer Perceptron and Naïve Bayes among others.

The rest of the paper is structured as follows; Section 2 reviews the literature related to this work. Section 3 presents the data mining models that were adopted. In Section 4, the application of the classifiers to the data is presented. Section 5 evaluates the performance of classifiers. The Conclusion appears in Section 6.

2. Related WORKs

Machine learning and Data mining in particular has been used in predictive models in many fields such as Fraud detection, Text mining, agriculture yield prediction and many more fields [20–22]. In addition, data mining techniques have found their use in medical field for discovery of drugs [23,24] and other health related issues. For example, the Naive Bayes classifier and J48 decision tree algorithm have been used before to build predictive models for MERS-CoV infections [25]. MERS-CoV which broke out in the Middle East is closely related to Covid-19 [26]. Another example is demonstrated in Ref. [27] where data mining and evolutionary algorithms are implemented to develop accurate readmission of patients in the hospital using prediction models. In that study, data mining algorithms were used to explore a classifier to distinguish potential readmitted and non-readmitted patients in the hospital.

Since the beginning of the Covid-19 pandemic, a lot of efforts have been put in place to combat the virus to bring life back to normal. Several researches have been conducted on various aspects of the pandemic from economic, social [28], education and health issues. Several models based on machine learning and data mining techniques have been put forward to predict the spread of Covid-19. Ahmad et al. [29] gives a comprehensive literature of studies that have used machine learning methods to predict the number of confirmed cases of Covid-19. Notably, Chintalapudi et al. [30] used a data driven model called Auto-regressive integrated moving average (ARIMA) found in R statistical package to highlight the importance of Italy's country lockdown and self-isolation in controlling the Covid-19 disease transmissibility in Italian population [30]. The study in Ref. [31] used machine learning techniques to predict a poor prognosis in positive Covid-19 patients and possible outcomes based on officially registered case of Covid-19 in Brazil. The study in Ref. [32] used machine learning to determine the correlation between weather and Covid-19. Results obtained indicates that the weather affected the number of Covid-19 patients.

In addition to machine learning approaches reported above, data mining techniques have been used to model and predict the spread of Covid-19. For instance, Saba et al. [33] used linear regression data mining technique to predict the future incidence of coronavirus cases in Pakistan. The study in Ref. [34] applied Linear regression and long short-term memory (LSTM) models to estimate the number of positive Covid-19 cases in data obtained from the Google Trends website in Iran. A similar study for USA used Pearson and Kendall rank correlations to determine correlations between Google Trends and Covid-19 data [35].

In addition, data mining techniques were used in content analysis of the Chinese Social Media Platform Weibo during the Early Covid-19 Outbreak to determine the correlation between social media post and reported Covid-19 cases [36]. In particular, linear regression model was used to determine if Weibo Covid-19 posts were predictive of the number of cases reported. It was determined that a positive correlation existed between the number of Weibo posts and the number of reported cases from Wuhan.

In [37] predictive data mining models for predicting recovery of Covid-19 cases in South Korea were developed. The attributes considered were gender, age, infection case (how they contracted the virus), number of days (difference between the day a patient was diagnosed Covid-19 positive and the day when they were released from hospital or died), and state of the Patient (whether released or deceased). The models used were Decision tree, Support vector machine, Naïve Bayes, Logistic regression, Random forest and K-nearest neighbor. The performance of these models was evaluated and they reported that the Decision tree classifier yielded the best performance in terms of efficiency and accuracy. The order of performance from best to least performing was as follows: Decision tree, Random forest, Support vector machine, K-nearest neighbor, Naïve Bayes, and lastly Logistic Regression.

It is clear from the works reported above that machine learning and data mining have found their application in Covid-19 pandemic prediction. The related works have further reported a number of aspects such as predicting the spread of the pandemic, future incidences of the pandemic, possible future readmissions of patients, recovery of infected patients among others. One aspect worth investigating is how data mining can be used to understand how the pandemic affect different categories of infected populations.

In this paper, we focus on the performance analysis of a number of classifiers in predicting the spread of Covid-19 in Zambia among different categories of infected populations. The performance of these classifiers are compared with the actual figures obtained from the Ministry of Health and related works.

3. Data mining models

A number of data-mining models have been developed over time to solve different problems. The WEKA machine learning library stands out among the tools available for data mining. In this work, a number of classifiers were used, the J48 decision tree, Multilayer Perceptron and Naïve Bayes among others. The J48 decision tree is robust and produces a tree that is easily understandable.

3.1. J48 decision tree classifier

According to Ref. [38], the J48 classifier is a top-down, recursive divide and conquer strategy based on information theory. The classifier is used to generate a C4.5 decision tree. The decision tree can be used for classification. Algorithm 1 shows a simplified high-level J48 algorithm. The details of the J48 classifier can be found in Ref. [38].

ALGORITHM 1
The simplified high-level J48 Algorithm [38].

Step 1: The classifier first selects an attribute to split on to serve as a root node. A branch is then created for every possible attribute value.
Step 2: The instances are then split into subsets, one for every branch extending from the root node.
Step 3: Repeat the procedure recursively from step 1 for each branch, that is, selecting an attribute for each node, and using only attributes that reach the branch.
Step 4: Stop if all attributes have the same class.

According to Weka there are a number of steps to be followed in applying the J48 classifier. Firstly, the dataset has to be prepared in a suitable format. Following this the dataset should be loaded in WEKA. After loading the dataset, the classifier in this case the J48 classifier is selected. After running the classifier on the dataset, several output is presented.

The output contains several information. The decision tree in a graphical format is produced. In addition the summary of the dataset is presented. Cross-validation details are also presented if used. A pruned decision tree is presented in textual form. The number of leaves are also presented and the total number of nodes. Following this is the predictive performance of the classifier. The confusion matrix is also presented. In addition to the classification error, the Kappa statistic, mean absolute error, root mean-square error information is presented.

3.2. Naïve Bayes

The Naïve Bayes classifier comprises classification algorithms that are based on Bayes' theorem [39]. The theorem aims to find the probability of event occurring when given the probability of another event that has already taken place. Equation (1) states the Bayes theorem.

$$P(A|B) = P(B|A)\ P(A)/P(B) \quad (1)$$

According to (1), P(A) is the apriori probability of event A whereas P (B) is the prior probability of event B. P (A|B) is the posterior probability which is the probability of event A happening given that event B has occurred. P(B|A) is the probability of event B happening given that event A has occurred. Instead of referring to events, the Bayes' formula can be recast to

$$P(y|X) = P(X|y)P(y)/P(X) \quad (2)$$

where y represents a class and X represents a feature vector. X comprises features which are assumed to be independent of each other.

In the Naïve Bayes method, all the attributes in the dataset are used. It is assumed that all attributes are equally important and statistically independent. The naïve assumption in Naïve Bayes is that the X in equation (2) can be split into features that are independent.

In order to apply Naïve Bayes classifier in WEKA, the dataset should be prepared first with appropriate features. This is followed by selecting the classifier, in this case the Naïve Bayes classifier. The output of the Naïve Bayes model is in this case a table. The output first shows the attributes and classes values for the features. Also displayed is a summary that comprises correctly/incorrectly classified instances and classification errors.

The detailed accuracy by class information such as true positive rate, false positive rate, precision, recall among others is also output. Finally, the confusion matrix is also displayed.

3.3. Multilayer perceptron

The multilayer perceptron is a neural network that employs the back propagation algorithm for training [38]. A multilayer perceptron has three layers which are input, hidden and output layers. On the input layer, the data to be analyzed is input into the neural network. In the hidden layers, inputs are taken and outputs through application of activation functions. The output layer produces the final result.

3.4. Other classifiers

There are several other classifiers that can be employed in data-mining [38]. For example, Random Forest is an algorithm that constructs a number of decision trees during training process. When used for classification the algorithm produces a class that is selected by majority of the decision trees. K Nearest Neighbor is a supervised machine learning algorithm that determines a number of K data points in feature space closest to the features (attributes) in question. Logistic Regression is a statistical model that determines the probability of existence of a particular class. Support Vector Machine algorithms work to determine a hyperplane that separate data belonging to different classes. Further details about these classifiers can be found in Ref. [38].

4. Application of classifiers to predictive modelling of COVID-19 cases

The Covid-19 cases in Zambia are reported by the Ministry of Health through the Zambia National Public Institute [40]. The dataset used had several attributes and comprised 10,664 instances after cleaning. The instances were for a period from March 18, 2020 to 11 September 2020. The attributes were Epid Number (epidemic number), Sex, Age, Date Specimen Collected, Date Specimen Received, Specimen Condition, District, Province, Final Result, Date of Reporting, Address and lastly, Category. The Final Result attribute recorded a positive Covid-19 test result. The Category attribute recorded various categories of those tested. These included, among others, Contact to Known case, Hospital Screening, Brought In Dead and Routine Screening. In this work, five attributes were used which include Sex, Age, Province, Date of Reporting, and Category. Other attributes were not relevant for this study. In general, the selection of suitable classifier attributes requires careful analysis.

Before applying complex classifiers, a baseline accuracy was initially undertaken. In this case a simple classifier was applied to the dataset and its accuracy noted. Following this, a more complex classifier was then applied to the dataset. The idea is that if the more complex classifier gives better accuracy than the simpler one, then the more complex classifier was preferable. In this work, a baseline rules classifier called ZeroR was applied to the dataset. Another simpler classifier called OneR was also applied to the dataset. These were considered simple compared to J48, Naïve Bayes and Multilayer Perceptron and others.

The Covid-19 statistics used in this work were stored in a relational database on a remote computer serving as a server. The Covid-19 data was made available to requesting clients through representational state transfer (REST) application programming interfaces (API) where it was sent in JavaScript object notation (JSON) format.

The python programming language was used to write the client. The weka code used for this work is available as a python package at [41] and further details on how to use the code can be found in Ref. [42]. In order to further process the JSON formatted data, it was converted into a dataframe using readily available modules.

The models were applied in two cases. At national level and provincial level. The provincial level was chosen instead of district level in order to simplify the results.

4.1. Applying the models at national level

In order to apply the models at national level, the entire dataset for the whole period from March to September 2020 was first considered. The decision tree produced when applying the J48 algorithm on all five attributes was huge and complex. Furthermore, the performance was not impressive. Fig. 1 shows a simpler decision tree produced by the J48 algorithm for the whole country with only four attributes where the Date of Reporting attribute was not considered. This covered the period 1st August 2020 to 11 September 2020.

The algorithm first branches according to provinces. Then further branches vary from one province to another.

4.2. Applying the models at provincial level

To apply the models at provincial level, datasets corresponding to a particular province was extracted. Fig. 2 shows the decision tree produced by the J48 algorithm where the Date of Reporting attribute was not considered. This tree was produced for one of the ten provinces,

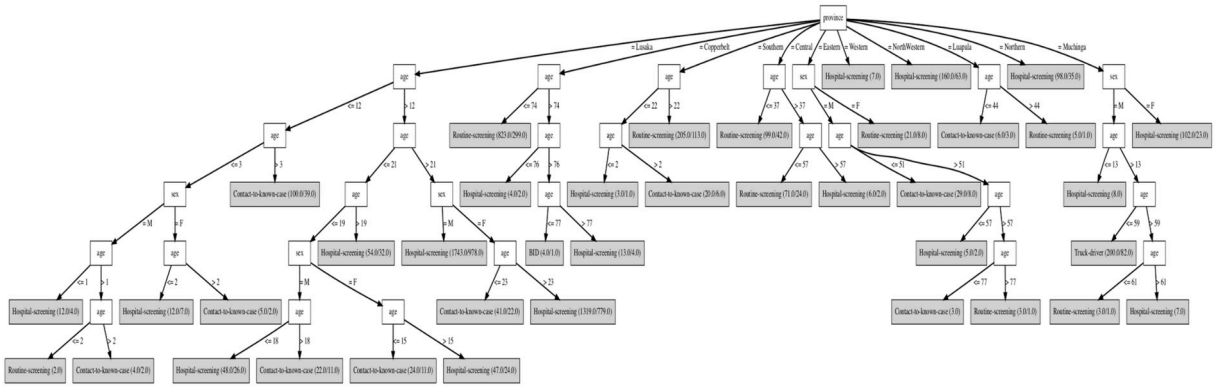

Fig. 1. Decision tree produced by the J48 algorithm for the whole country from 1st August 2020 to 11 September 2020.

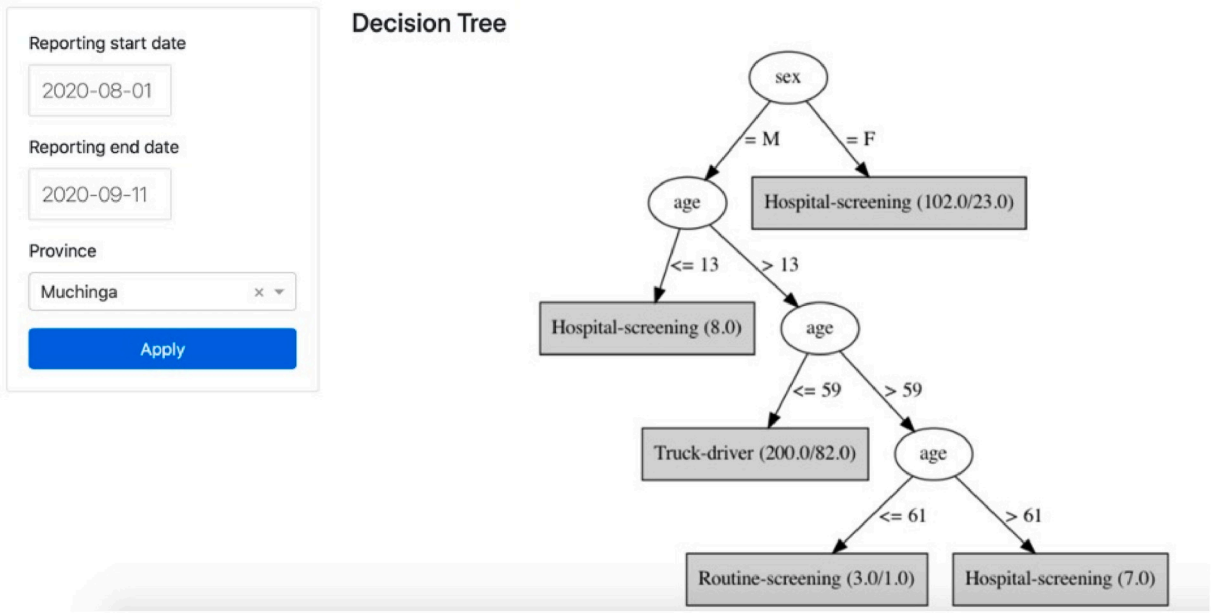

Fig. 2. Decision tree produced by the J48 algorithm for Muchinga province over the stated period.

namely Muchinga. As can be seen in the Figure, the algorithm first branched on gender, followed by age in the case of males. Then cases were further categorized as Hospital-screening, Truck-driver and Routine-screening. In this case, the algorithm indicated that all females who were found with Covid-19 were diagnosed through Hospital screening, implying that they were unwell. Furthermore, according to Fig. 2, there are a few young ones (males) who are found with Covid-19 and these are diagnosed through hospital screening. The majority of the males who were found to be Covid-19 positive were 59 years and above.

5. Evaluating the performance of classifiers

There are several ways to evaluate the performance of a classifier. These include evaluating on an independent test set, evaluating using a holdout method (where data is partitioned into a training set and test set) and cross-validation. The holdout method can be repeated on different training and test datasets, referred to as repeated holdout. Cross-validation improves on repeated holdout where the variance of the estimate can be reduced systematically. With cross-validation, a training set is first created. Following this, a classifier is then created followed by evaluating the full performance of that classifier. With stratified cross-validation, the variance of the estimated performance can be reduced even further.

In order to assess the performance of the classifiers used in this work, several methods were used. Evaluating on an independent test set, evaluating using a hold-out method (where data is partitioned into a training set and test set), cross-validation was undertaken.

5.1. Baseline accuracy

Before evaluating the performance of the classifiers, a baseline accuracy evaluation was undertaken. In this case, a ZeroR classifier was used. ZeroR is the simplest among classifiers in that it only considers the most frequent values among the target attributes [38]. It completely ignores the rest of the attributes. Table 1 shows the results of applying WEKA to the Covid-19 dataset for Muchinga province for the period 1st August to 11 September 2020. Here the attributes considered were Sex,

Table 1
Results of running ZeroR with 10-fold cross-validation for Muchinga province from 1st August to 11 September 2020.

Classifier	Correctly classified instances (%)	Incorrectly classified instances (%)	Total number of instances	Root mean squared error
ZeroR	49.375	50.625	320	0.3881

Age, Date of Reporting, and Category.

5.2. OneR accuracy

The OneR (One Rule) classifier generates one rule for every predictor attributes [38]. Following this, the classifier then selects a rule with the smallest total error. When the OneR classifier was used, the performance was better compared to ZeroR. Table 2 shows the obtained results after running OneR with 10-fold cross-validation for Muchinga province from 1st August to 11 September 2020 with the date attribute considered. The attributes considered were Sex, Age, Date of Reporting, and Category.

5.3. J48 accuracy

Table 3 shows the obtained results after running J48 classifier with 10-fold cross-validation for Muchinga province from 1st August to 11 September 2020. The attributes considered were Sex, Age, Date of Reporting, and Category since Province was fixed. The inclusion of Date of Reporting attribute improved the performance of the classifier.

With different random number seeds a repeated holdout was undertaken in WEKA. Table 4 illustrates the correctly classified instances for different random seeds and 10-fold cross-validation. The sample mean (μ) and the standard deviation (σ) were obtained from the following formulas

$$\mu = \sum x_i \Big/ \mathrm{N} \tag{3}$$

$$\sigma = \sqrt{\frac{1}{\mathrm{N}-1}\sum (x_i - \mu)^2} \tag{4}$$

where x_i is the percentage of correctly classified instances for an experiment and N is the number of experiments. As indicated in Table 4, the mean is not significantly different from each experiment.

5.4. Comparisons

Table 5 compares results for different classifiers. Among these, the Decision tree (here referred to as J48), Naïve Bayes, Random Forest, Support Vector Machine, K Nearest Neighbor and Logistic Regression were also studied in Ref. [37] where they obtained good results. The J48 and Naïve Bayes were also studied in Ref. [25] and showed good results. As shown in Table 5, classifiers such as J48, Multilayer Perceptron, Naïve Bayes, Support Vector Machine and Logistic Regression performed better than ZeroR and OneR.

Table 6 compares results for various classifiers across the nation. However, despite performing better than ZeroR and OneR, the accuracy was not impressive and therefore performance of several individual provinces was studied.

Tables 7 and 8 compare results across the other provinces of the country. The results show that in a number of provinces, the performances of complex classifiers were better than baseline accuracy. However, there are cases where simpler classifiers such as OneR performed well.

6. Conclusion

A number of data-mining models have been applied in order to uncover hidden patterns among Covid-19 captured patient features. These include ZeroR, OneR, J48 decision tree, Multilayer Perceptron, Naïve Bayes, Random Forest, Support Vector Machine, K Nearest Neighbor and Logistic Regression. The individual performances of these

Table 2
Results of running OneR with 10-fold cross-validation for Muchinga province from 1st August to 11 September 2020.

Classifier	Correctly classified instances (%)	Incorrectly classified instances (%)	Total number of instances	Root mean squared error
OneR	86.875	13.125	320	0.2562

Table 3
Results of running J48 with 10-fold cross-validation for Muchinga province from 1st August to 11 September 2020.

Classifier	Correctly classified instances (%)	Incorrectly classified instances (%)	Total number of instances	Root mean squared error
J48	87.1875	12.8125	320	0.2215

Table 4
Results of running J48 with different random seeds and repeated holdout for Muchinga province from 1st August to 11 September 2020.

Random seed	Correctly classified instances (%)
1	87.1875
2	87.5
3	87.5
4	86.5625
5	86.5625
6	86.875
7	86.25
8	86.25
9	86.875
10	86.5625
Sample mean	86.8125
Standard deviation	0.4612

Table 5
Comparison of results for Muchinga province with 10-fold cross validation from 1st August to 11 September 2020. Total number of instances = 320.

Classifier	Correctly classified instances (%)	Incorrectly classified instances (%)	Root mean squared error
ZeroR	49.375	50.625	0.3881
OneR	86.875	13.125	0.2562
J48	87.1875	12.8125	0.2215
Multilayer Perceptron	87.5	12.5	0.2128
Naïve Bayes	87.5	12.5	0.2239
Random Forest	85.9375	14.0625	0.2394
Support Vector Machine	88.4375	11.5625	0.3288
K Nearest Neighbor	82.5	17.5	0.2663
Logistic Regression	87.5	12.5	0.2177

Table 6
Comparison of results for the nation (Zambia) with 10-fold cross validation from 1st August to 11 September 2020. Total number of instances = 5338.

Classifier	Correctly classified instances (%)	Incorrectly classified instances (%)	Root mean squared error
ZeroR	37.692	62.308	0.2978
OneR	47.0776	52.9224	0.3637
J48	57.8494	42.1506	0.2645
Multilayer perceptron	48.5013	51.4987	0.2837
Naïve Bayes	48.1266	51.8734	0.2825
Random Forest	53.1847	46.8153	0.2899
Support Vector Machine	55.2454	44.7546	0.3014
K Nearest Neighbor	51.8359	48.1641	0.3268
Logistic Regression	55.976	44.024	0.2619

Table 7
Comparison of Correctly Classified instances for some provinces of Zambia with 10-fold cross validation from 1st August to 11 September 2020.

Province	Number of instances	ZeroR (%)	OneR (%)	J48 (%)	Multilayer perceptron (%)	Naïve Bayes (%)
Lusaka	3433	41.5963	49.2281	49.9854	48.9659	50.9758
Copperbelt	844	62.3223	68.128	67.1801	66.2322	67.7725
Central	176	59.0909	83.5227	81.25	81.8182	81.25
Southern	228	42.5439	54.8246	47.807	47.807	57.8947
Eastern	61	54.0984	67.2131	77.0492	80.3279	72.1311
Northwestern	160	60.625	82.5	78.75	80	78.75
Luapula	11	0	100	81.8182	100	54.5455
Northern	98	64.2857	86.7347	84.6939	84.6939	82.6531
Muchinga	320	49.375	86.875	87.1875	87.5	87.5

Table 8
Comparison of Correctly Classified instances for additional classifiers reported in Ref. [37] with 10-fold cross validation from 1st August to 11 September 2020.

Province	Number of instances	Random Forest (%)	Support Vector Machine (%)	K Nearest Neighbor (%)	Logistic Regression (%)
Lusaka	3433	43.7518	49.1407	43.1984	51.2962
Copperbelt	844	62.6777	68.128	61.4929	69.4313
Central	176	73.8636	84.0909	69.8864	82.3864
Southern	228	45.1754	55.2632	40.3509	54.8246
Eastern	61	75.4098	67.2131	78.6885	77.0492
Northwestern	160	75	81.25	73.125	80
Luapula	11	90.9091	100	90.9091	100
Northern	98	85.7143	86.7347	81.6327	84.6939
Muchinga	320	85.9375	88.4375	82.5	87.5

classifiers have been presented and results indicate good performances in a number of cases. However, there were cases where simple classifiers such as OneR also performed well. The purpose of studying different data mining models was to gain an understanding on how the pandemic affects different population categories. Since each model has its own principles and assumptions, this work compared several models to determine those that predicts the occurrence of the pandemic with minimal errors.

The models will provide insights on population groups most affected by the pandemic. Furthermore, the study can be beneficial not only to Covid-19 prediction but also to other pandemics experienced by the nation. The models also provide insights on the most effective way to contain the pandemic across population groups.

Credit author statement

Josephat Kalezhi: Funding acquisition, Writing - Original Draft, Software, Conceptualization, Investigation, Visualization, Writing - Review & Editing. Mathews Chibuluma: Data Curation, Software, Methodology, Visualization, Investigation, Writing - Review & Editing. Christopher Chembe: Software, Writing - Original Draft, Conceptualization, Writing - Review & Editing. Victoria Chama: Software, Formal analysis. Francis Lungo: Validation, Formal analysis, Resources. Douglas Kunda: Funding acquisition, Resources, Conceptualization, Methodology, Supervision, Project administration, Writing - Review & Editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This project was funded by the National Science and Technology Council (NSTC).

References

[1] Center for Disease Control and Prevention, Coronavirus disease 2019 (COVID-19) - how it spreads. https://www.cdc.gov/coronavirus/2019-ncov/prepare/transmission.html, 4th March 2020.
[2] A. Al-Gheethi, E. Noman, Q.A.A. Al-Maqtari Q, Novel Coronavirus (2019-nCoV) Outbreak; a Systematic Review for Published Papers, SSRN Electronic Journal, 2020.
[3] D. Cucinotta, M. Vanelli, WHO declares COVID-19 a pandemic, Acta Biomed.: Atenei Parmensis 91 (1) (2020) 157.
[4] J. Zhao, et al., COVID-19: vaccine development updates, Front. Immunol. 11 (2020) 3435.
[5] S. Alexandar, et al., A comprehensive review on Covid-19 Delta variant, Int. J. Pharmaceut. Chem. Res. 5 (2021) 83–85.
[6] S.S.A. Karim, Q.A. Karim, Omicron SARS-CoV-2 variant: a new chapter in the COVID-19 pandemic, Lancet 398 (2021) 2126–2128, 10317.
[7] I. Shinya, et al., Artificial intelligence system for supporting soil classification, Result. Eng. 8 (2020) 100188.
[8] D. Ritesh, D.K. Dash, G.C. Biswal, Classification of crop based on macronutrients and weather data using machine learning techniques, Result. Eng. 9 (2021) 100203.
[9] S. Lalmuanawma, J. Hussain, L. Chhakchhuak, Applications of machine learning and artificial intelligence for Covid-19 (SARS-CoV-2) pandemic: a review, Chaos, Solit. Fractals 139 (2020) 110059.
[10] O. Shahid, et al., Machine learning research towards combating COVID-19: virus detection, spread prevention, and medical assistance, J. Biomed. Inf. 117 (2021) 103751.
[11] H. Lv, et al., Application of artificial intelligence and machine learning for COVID-19 drug discovery and vaccine design, Briefings Bioinf. 226 (2021).
[12] R. Vinuesa, et al., A socio-technical framework for digital contact tracing, Result. Eng. 8 (2020) 100163.
[13] Z. Car, et al., Modeling the spread of COVID-19 infection using a multilayer perceptron, Comput. Math. Method. Med. (2020) 2020.
[14] L.J. Muhammad, et al., Predictive data mining models for novel coronavirus (COVID-19) infected patients' recovery, SN Comput. Sci. 14 (2020) 1–7.
[15] T.B. Alakus, T. Ibrahim, Comparison of deep learning approaches to predict COVID-19 infection, Chaos, Solit. Fractals 140 (2020) 110120.
[16] G. Pinter, et al., COVID-19 pandemic prediction for Hungary; a hybrid machine learning approach, Mathematics 86 (2020) 890.
[17] R. Sujath, M.C. Jyotir, E.H. Aboul, A machine learning forecasting model for COVID-19 pandemic in India, Stoch. Environ. Res. Risk Assess. 34 (2020) 959–972.
[18] S.R. Weka Garner, The waikato environment for knowledge analysis, in: Proceedings of the New Zealand Computer Science Research Students Conference, 1995.
[19] E. Frank, et al., Weka-a machine learning workbench for data mining, in: Data Mining and Knowledge Discovery Handbook, Springer, 2009, pp. 1269–1277.
[20] S.J. Hong, S.M. Weiss, Advances in predictive models for data mining, Pattern Recogn. Lett. 22 (1) (2001) 55–61.
[21] G. Ruß, Data mining of agricultural yield data: a comparison of regression models, in: Industrial Conference on Data Mining, Springer, 2009.
[22] D.L. Olson, D. Wu, Predictive Data Mining Models, Springer, 2017.
[23] J. Bajorath, Compound data mining for drug discovery, in: Bioinformatics, Springer, 2017, pp. 247–256.
[24] A. Lavecchia, Machine-learning approaches in drug discovery: methods and applications, Drug Discov. Today 20 (3) (2015) 318–331.
[25] I. Al-Turaiki, M. Alshahrani, T. Almutairi, Building predictive models for MERS-CoV infections using data mining techniques, J. Infect. Publ. Health 9 (6) (2016) 744–748.
[26] D. Giannis, I.A. Ziogas, P. Gianni, Coagulation disorders in coronavirus infected patients: COVID-19, SARS-CoV-1, MERS-CoV and lessons from the past, J. Clin. Virol. (2020) 104362.
[27] B. Zheng, et al., Predictive modeling of hospital readmissions using metaheuristics and data mining, Expert Syst. Appl. 42 (20) (2015) 7110–7120.
[28] C. Chembe, A. Banda, J. Kalezhi, D. Kunda, 5G awareness and its link to COVID-19: case of Zambians with access to internet, Zambia ICT J. 5 (1) (2021) 30–40.
[29] A. Ahmad, et al., The number of confirmed cases of covid-19 by using machine learning: methods and challenges, Arch. Comput. Methods Eng. (2020) 1–9.

[30] N. Chintalapudi, G. Battineni, F. Amenta, COVID-19 disease outbreak forecasting of registered and recovered cases after sixty day lockdown in Italy: a data driven model approach, J. Microbiol. Immunol. Infect. 53 (3) (2020) 396–403.
[31] F.S.H. Souza, et al., Predicting the Disease Outcome in Covid-19 Positive Patients through Machine Learning: a Retrospective Cohort Study with Brazilian Data, medRxiv, 2020.
[32] A. Fadli, et al., Simple correlation between weather and COVID-19 pandemic using data mining algorithms, in: IOP Conference Series: Materials Science and Engineering, IOP Publishing, 2020.
[33] N. Saba, W. Akram, T. Ahmed, Predicting COVID-19 incidence using data mining techniques: a case study of Pakistan, BRAIN Broad Res. Artif. Intell. Neurosci. 11 (4) (2020) 168–184.
[34] S.M. Ayyoubzadeh, et al., Predicting COVID-19 incidence through analysis of google trends data in Iran: data mining and deep learning pilot study, JMIR Publ. Health Surveil. 6 (2) (2020) e18828.
[35] A. Mavragani, K. Gkillas, COVID-19 predictability in the United States using Google Trends time series, Sci. Rep. 10 (1) (2020) 1–12.
[36] J. Li, et al., Data mining and content analysis of the Chinese social media platform Weibo during the early COVID-19 outbreak: retrospective observational infoveillance study, JMIR Publ. Health Surveil. 6 (2) (2020) e18700.
[37] L.J. Muhammad, M.M. Islam, S.S. Usman, S.I. Ayon, Predictive data mining models for novel coronavirus (COVID-19) infected patients' recovery, SN Comput. Sci. 1 (4) (2020) 206, https://doi.org/10.1007/s42979-020-00216-w. Epub 2020 Jun 21. PMID: 33063049; PMCID: PMC7306186.
[38] I.H. Witten, E. Frank, M.A. Hall, Data Mining: Practical Machine Learning Tools and Techniques, fourth ed., Morgan Kaufmann, 2016.
[39] F. Alam, S. Pachauri, Detection using weka, Adv. Comput. Sci. Technol. 10 (6) (2017) 1731–1743.
[40] Zambia national public health Institute, Available: https://znphi.co.zm. (Accessed 19 February 2021).
[41] Available online: python-weka-wrapper3 0.2.5 https://pypi.org/project/python-weka-wrapper3/. (Accessed 1 January 2022).
[42] Introduction — python-weka-wrapper3 0.2.5 documentation, Available online: https://fracpete.github.io/python-weka-wrapper3/#. (Accessed 1 January 2022).