RESEARCH

Maintaining proper health records improves machine learning predictions for novel 2019-nCoV

Koffka Khan[1*] and Emilie Ramsahai[2]

*Correspondence:
koffka.khan@sta.uwi.edu
[1]Department of Computing and Information Technology, The University of the West Indies, St. Augustine, Trinidad and Tobago
Full list of author information is available at the end of the article

Abstract

Background:: An ongoing outbreak of a novel coronavirus (2019-nCoV) pneumonia continues to affect the whole world including major cities such as China, USA, Italy, France and the United Kingdom. We present outcome ('recovered', 'isolated' or 'death') risk estimates of the 2019-nCoV over 'early' datasets. A major consideration is how likely are people to die from 2019-nCoV?

Method:: Accounting for the impact of the variations in the reporting rate of 2019-nCoV, we modelled machine learning techniques (AdaBoost, Bagging, Extra-Trees, Decision-Trees and k-Nearest Neighbours Classifiers) on two 2019-nCoV datasets obtained from Kaggle in March 30th 2020. We used 'country', 'age' and 'gender' as features to predict outcome for both datasets. Including the patient's 'disease' history (only present in the second dataset) to predict outcome for the second dataset.

Results:: The use of a patient's 'disease' history improves the prediction of 'death' by more than a 7-fold. Models ignoring a patent's 'disease' history performed poorly in test predictions.

Conclusion:: Our findings indicate the potential of using a patient's 'disease' history as part of the feature set in machine learning techniques to improve 2019-nCoV predictions. This can have a positive effect on predictive patient treatment and result in ease for current overburdened healthcare systems worldwide, especially with an increasing prevalence of second and third wave re-infections in some countries.

Keywords: 2019-nCoV; pneumonia; machine learning; AdaBoost; Bagging; classifiers; disease; death; prediction

1 Introduction

A zoonotic coronavirus has crossed species to infect human populations. This virus, unofficially known as 2019-nCoV, was first detected in people exposed to a seafood or wet market in Wuhan, China. Like other pathogenic human respiratory coronaviruses, 2019-nCoV triggers respiratory disorders that are sometimes severe. More than 1,133,758 confirmed cases were registered as of 5th April 2020, with 62,784 deaths [1].

The disease has now evolved to be spread by human-to-human communication. Typical clinical signs in these patients include fatigue, dry cough, trouble swallowing (dyspnoea), headache, and pneumonia. The development of the disease can result in progressive respiratory failure due to alveolar damage (as seen in computerized transverse chest tomography images) and even death.

Being an ribonucleic acid (RNA) virus [2], 2019-nCoV also has an intrinsic characteristic of a high mutation rate, but, as other coronaviruses, the mutation rate could be significantly lower than other RNA viruses owing to its genome-encoded exonuclease. This feature offers the potential for this recently developed zoonotic viral pathogen to evolve and then being more easily spread from person to person and likely to become more virulent.

Recently, machine learning techniques have been applied successfully to a wide range of problems with health care being no exception [3, 4, 5]. Since, the appearance of 2019-nCoV many researches have employed machine learning techniques to predict patterns related to various genotypic and phenotypic viral traits combined with human social behavior. This has met with varying results. However, with the introduction of new datasets researchers are eager to engage various machine learning techniques to help manage this outbreak.

Initial datasets were very sparse and at first included only a single country. Consequently, as 2019-nCoV spread the increase awareness and record keeping meant datasets grew in number of features and size. Nonetheless, at the beginning of April 2020 the global number of datasets available to researchers are still quite small. Despite this we choose two datasets from Kaggle [6, 7].

We focused our work on predicting impending death from 2019-nCoV based on given data. Our aim was to develop a tool for precise risk prediction to facilitate urgent treatment targeted at high risk individuals. Our analysis focuses on many state-of-the-art algorithmic developments which have demonstrated promise in improving disease prediction. The development of a more in-depth understanding and theoretical study of critical problems related to algorithmic construction and learning theory was crucial in the advancement. Those include trade-offs for optimizing efficiency [8] using physically practical constraints, and integrating prior information and uncertainty. Our contributions are as follows:

1 Create, train and test models based on five machine learning techniques from two Kaggle datasets. Machine learning hyper-parameters were tuned to obtain models with optimal performance.
2 We confirmed that using the patient's 'disease' history resulted in more than a 7-fold increase in the prediction of death.
3 We developed a machine learning tool for death prediction to facilitate urgent treatment targeted at high risk individuals. The tool works for 'early' datasets with few deaths but will improve with the addition of more patient cases. Thus, it can be used for countries now developing cases and those with many cases.
4 In the future improved death predictions can assist worldwide healthcare systems in fighting this outbreak.

The rest of this paper is organised as follows. In Section 2 divulges important background work. We discuss different machine learning techniques and statistical metrics used in this paper. We then outline the method used to setup our experiments including dataset descriptions and parameters utilized for machine learning techniques in Section 3. Our results are given in Section 4 together with a discussion giving the importance of using a patient's 'disease' history as a feature in 2019-nCoV datasets. Finally we present our conclusions in Section 5.

2 Background

In the background we given brief explanations on three Ensemble and two Conventional methods used in machine learning. These are important as they are used to build the models used for predictions in our experiments. Then we discuss the metrics used to evaluate the performance of these models.

2.1 Ensemble Methods

An ensemble is a composite model, combining a set of low-performing classifiers to construct an improved classifier. An individual vote is performed by the classifier and final prediction label returned, which results in majority voting [9]. In essence, ensemble learning methods are meta-algorithms incorporating many methods of machine learning into one predictive model to improve performance. We selected three ensemble methods based on literature performance on assisting with pandemic predictions [10, 11, 12]. These are AdaBoost, Bagging and Extra-Trees classifiers. They are described in the upcoming sub-sections.

2.1.1 AdaBoost Classifier

Ada-boost or Adaptive Boosting is a classifier combines several classifiers to improve classifier accuracy. AdaBoost is an iterative ensemble method. It creates a strong classifier by combining several poorly performing classifiers and you get an effective classifier with high precision [13]. The basic idea behind Adaboost is to set classifier weights and train the data sample in each iteration in such a way that it ensures precise predictions of unusual observations [14]. Many other machine learning techniques can be used as base classifier if it accepts weights on the training set [15].

The AdaBoost Classifier operations is outlined in the following steps. Initially, Adaboost selects a training subset randomly. It iteratively trains the AdaBoost machine learning model by selecting the training set based on the accurate prediction of the last training. It assigns the higher weight to wrong classified observations so that in the next iteration these observations will get the high probability for classification. Also, it assigns the weight to the trained classifier in each iteration according to the accuracy of the classifier. The more accurate classifier will get high weight. This process iterate until the complete training data fits without any error or until reached to the specified maximum number of estimators. To classify, perform a "vote" across all of the learning algorithms you built.

2.1.2 Bagging Classifier

A Bagging classifier is a meta-estimator ensemble that suits base classifiers on the original dataset's random subsets each, and then aggregates their individual predictions (either by voting or by averaging) to form a final prediction [16]. Usually, such a meta-estimator can be used as a means of reducing the variance of a black-box estimator (e.g., a decision tree), integrating randomization into its construction process and then making an ensemble out of that [17]. A training set of size N is sampled for each trial t = 1,2, $\cdots$ T, with substitution from the original instances. This training set is the same size as the original data but it may not include any instances while others appear more than once. A sample classifier is created by the

learning system, and the final classifier is produced by combining the classifiers from those trials. To classify an instance, every classifier registers a vote for the class. The class with the most votes is then selected (ties are resolved arbitrarily).

2.1.3 Extra-Trees Classifier

This class implements a meta-estimator that fits a number of randomized decision trees (i.e. extra-trees) on different dataset sub-samples and uses average to improve predictive accuracy and balance over-fitting power [18]. Rather than using a bootstrap replica Extra-Trees uses the entire training collection to create the trees. The method randomly selects an attribute to divide the data at each stage of the tree (if it is a continuous attribute then the cut-point is also chosen randomly), i.e. independently of the class labels [19].

2.2 Conventional Methods

Many conventional learning algorithms have gained a huge attraction in many fields of research [20]. We selected two conventional methods based on literature performance on assisting with pandemic predictions [21]. These are Decision-Tree and k-Nearest Neighbour (k-NN) classifiers. They are described in the upcoming subsections.

2.2.1 Decision-Tree Classifier

Decision Trees (DT) is a non-parametric, supervised method of learning used in classification [22, 23, 23, 24, 25]. For each possible outcome, each internal node of the decision tree has a test that includes an attribute and an outgoing branch. Each leaf has a class assigned to it. To classify new records using a decision tree, beginning with the root node, successive internal nodes will be visited before reaching a leaf. The test for the node is applied on the record at each internal node. The test outcome at an internal node specifies the branch being traversed, and the next node being visited. The record class is simply the class of the final leaf node on the respective branch. The conjunction of all the conditions for the branches from the root to a leaf thus constitutes one of the conditions for the leaf-associated class [26].

2.2.2 k-Nearest Neighbors Classifier

The goal is to construct a model that predicts the value of a target variable by learning basic rules of decision inferred from the data characteristics [27]. The nearest neighbor classifier (NNC) assigns a class to the given test pattern that is the class of its closest neighbor in the distance function in the training set. The k-nearest neighbor classifier (k-NN) where k is an integer such that k-nearest is greater than or equal to 1 is a NNC generalization [28]. In the training set, the nearest k neighbors for the given test pattern Y are selected. The class information is maintained for every of the nearest k neighbors. If there are more than two winners in the majority vote there is a tie which is arbitrarily broken to determine the winner.

2.3 Statistical Metrics

Once you have a model based on machine learning techniques, you want to know how well it is performing. The desired model to assist in providing the necessary

treatment for those most at risk, must be good and reliable at predicting deaths. We use the accuracy, precision, recall and F1-Score statistical metrics to measure the performance of the models in our experiments [29, 30]. To calculate these values several other values are needed. True positive (TP), True negative (TN), False negative (FN) and False positive (FP). Both TP and TN indicate a consistent result between the prediction and the actual outcome. Conversely FN and FP indicate the predictions are not the same as the actual condition. In the case of our death prediction, we recognised FP, are not as dangerous as FN. Our aim was to minimize the number of false negatives (FN), as these are the cases where death is not correctly predicted and the patient does not receive adequate medical attention. We briefly describe these metrics and their calculations in the following sub-sections.

2.3.1 Accuracy

The most popular classification metric is accuracy, which is the fraction of the samples correctly predicted. It is described by Equation 1.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

2.3.2 Precision

Precision is the proportion of successful predicted occurrences, which are actually positive. It is described by Equation 2.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

2.3.3 Recall

Recall (also called sensitivity) is the proportion of successful events that are correctly predicted. It is described by Equation 3.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

2.3.4 F1-Score

The F1-score is the harmonic mean of recall and precision, with the greater score interpreted as a better model. It is described by Equation 4.

$$F1 - Score = \frac{2 * (Precision * Recall)}{Precision + Recall} \tag{4}$$

3 Literature Review

Randhawa et. al. introduces an intrinsic genomic signature of the COVID-19 virus and uses it for an ultra-fast, scalable, and highly accurate classification of entire 2019-nCoV virus genomes along with a machine learning-based alignment-free approach [31]. Ozturk et. al. proposed a deep model to use X-ray images for early detection of COVID-19 cases [32]. They obtained a 98.08% accuracy and 87.02% accuracy for discrete and multi-classes. Their DarkCovidNet model can help the clinicians make the diagnosis quicker and more accurately. In [33] researchers consider the problem of automatic classification of pulmonary diseases, including the

recently emerged 2019-nCoV, from X-ray images. A state-of-the-art Convolutional Neural Network (CNN) called Mobile Net is trained from scratch to investigate the importance of the extracted features for the classification task. A classification accuracy of 87.66% across is achieved among seven classes. In comparison, this approach achieves 99.18%, 97.36% sensitivity and 99.42% specificity in 2019-nCoV identification. Researchers used artificial intelligence (AI) algorithms to combine chest CT findings with clinical symptoms, exposure history and laboratory testing to quickly diagnose patients who are positive for 2019-nCoV [34]. The system correctly identified 17 of 25 (68%) patients and achieved an area under the curve of 0.92. In [35] researchers propose a clinical text classification paradigm using weak supervision and deep representation to reduce these human efforts. They test Support Vector Machine (SVM), Random Forest (RF), Multilayer Perceptron Neural Networks (MLPNN), and CNN, using a weak supervision paradigm. Precision, recall, and F1 score were used as metrics to evaluate performance. CNN achieved the best performance. Although there are many other publications focusing on supervised machine learning applied to 2019-nCoV in various ways there are none involving death predictions as we start to explore in the next section.

4 Methods

We used two datasets from Kaggle [6, 7]. These datasets were obtained on March 30th 2020. Dataset1 has 1086 cases with nineteen (19) features and dataset2 2756 cases with eighteen (18) features. The feature set and outcome variable were separated and formatted. Initial experiments predicted two outcomes ('Alive' and 'Death') followed by experiments which predicted three outcomes ('recovered', 'isolated' or 'death'). The common outcome is 'death'. We aim to develop a model which accurately predicts 'death'. Hence we construct a model for 'death' independently. We build a model for predicting the probability of death that would not be used to predict 'recovered' or 'isolated'.

The models were evaluated using accuracy, precision, recall and F1-score. For each patient we predict each outcome using a number of features. The initial features were 'country', 'gender' and 'age' from both datasets yielding three sub-samples. We filtered out patient cases that does not include all of the features. This created a total number of 816 cases in dataset1 and 2754 cases in dataset2. We then included the patient's disease history feature, 'disease' from dataset2 which produced our final sub-sample. There were no invalid cases for this feature set. We created an outcome variable for the categorical outcome of 'recovered', 'isolated' and 'death'. The following lists shows our four sub-samples.

- The first sub-sample came from dataset1 and has feature set 'country', 'age' and 'sex' with two outcomes ('alive' and 'death').
- The second sub-sample came from dataset1 and has feature set 'country', 'age' and 'sex' with three outcomes ('recovered', 'isolated' or 'death').
- The third sub-sample came from dataset2 and has feature set 'country', 'age' and 'sex' with three outcomes ('recovered', 'isolated' or 'death').
- The fourth sub-sample came from dataset2 and has feature set 'disease', 'age' and 'sex' with three outcomes ('recovered', 'isolated' or 'death').

The models were trained on these four sub-samples using three Ensemble and two Conventional algorithms. Python version 3.5 and Scikit learn machine learning libraries were used. The five machine learning techniques were used with the following settings in all experiments after tuning for optimal performance:

- AdaBoost Classifier used a Decision Tree Classifier with a maximum depth of 2, learning rate of 2 and number of estimators equal to 100.
- Bagging Classifier used number of estimators equal to 10, a warm start set to false and a random state equal to 3141.
- Extra-Trees Classifier used number of estimators equal to 100 and a random state of 12.
- Decision Tree used default settings.
- k-NN used the number of neighbors set to 1.

Models were trained on 80% of the sub-samples and tested on 20%. The following steps were used for each experiment:

1 The data files were retrieved from the input directory.
2 The data was cleaned.
3 The outcome variable was defined.
4 The data was broken up into training and testing sets.
5 Non-numeric features were mapped to numeric values.
6 The machine learning technique and hyper-parameters was chosen.
7 The model was created using the training data.
8 Predictions were obtained by applying the model on the testing set.
9 Evaluations of model performance using relevant test metrics.

We have done experiments selecting the eventual best model, see Section 5. However, once a healthcare system has obtained the highest performing model they can run the above steps using the model which can give further direction on how to facilitate certain patients based on their health status. For instance, if the health status is death then measures can be put into place to better handle care measures for the patient. This would enable more effective use of healthcare resources in the health center or hospital.

5 Results

Dataset1 [6] provides daily level details (time series data) from 2019-nCoV on the number of infected cases, deaths and recovery. The data was made available from 22 Jan, 2020. The main file that we utilized in this dataset is covid_19_data.csv which is described by the following:

- Sno - Serial number
- Observation Date - Date of the observation in MM/DD/YYYY
- Province/State - Province or state of the observation (Could be empty when missing)
- Country/Region - Country of observation
- Last Update - Time in Coordinated Universal Time (UTC) at which the row is updated for the given province or country. (Not standardised and so please clean before using it)
- Confirmed - Cumulative number of confirmed cases till that date
- Deaths - Cumulative number of of deaths till that date

- Recovered - Cumulative number of recovered cases till that date

Dataset2 [7] is generated by the KCDC (Korea Centers for Disease Control & Prevention) which announces the information of COVID-19 quickly and transparently. The data was made available from 24 Feb, 2020. The main file that we utilized in this dataset is PatientInfo.csv which contains the following fields: patient_id, global_num, sex, birth_year, age, country, province, city, disease, infection_case, infection_order, infected_by, contact_number, symptom_onset_date, confirmed_date, released_date, deceased_date and state.

Dataset1 has 42.40% female and 57.6% male while dataset2 has 55.95% female and 44.05% male. Neither dataset1 nor dataset2 was skewed based on their age frequency, see age frequency distribution histogram plots on Figure 1 and Figure 2. However, further inspecting both datasets shows that it is particularly unbalanced for the outcome of death. There are only 7.10% of deaths in dataset1 and less than 2.0% in dataset2.

We initially tested fourteen classifiers: gaussian naive bayes, support vector machine, linear discriminant analysis, one versus rest, gradient boosting, random forest, bagging using a decision tree base estimator, bagging using a logistic regression base estimator, neural network multilayer perceptron, adaboost, bagging, extra-trees, decision tree and k-NN. However, for brevity we select the top five of these models, see Section 2.

Training models on unbalanced data produce inaccurate findings for the prediction on death. This is due to the vast number of alive cases. Our initial tests indicate high (0.94-0.97) precision, recall and F1-scores for survival prediction (alive) yet very low (0.31-0.50) for death prediction, see Table 1 trained on sub-sample one. The very low recall values (0.31-0.38) are attributed to the large number of incorrect predictions made for deaths (FN). Improving theses death predictions facilitate targeted treatment to high risk patients. Given that predicting deaths is preferable to having high model accuracy (0.6-0.91), obtaining a high recall is more significant. Thus, the aim of the subsequent experiments is to obtain a high recall value in the prediction of death. Low accuracy can contribute to low precision and recall when estimating positive data points. Low recall is based on a large number of false negatives (FN) and small number of true positives (TP).

On separating the outcome into three categories 'recovered', 'isolated' and 'death' no improvement was obtained in the prediction of death, see Table 2. In this experiment recall remained low (0.31-0.38) for death prediction. Thus, we choose a new dataset on which to build new models. This third experiment was run using sub-sample three, see Table 3. However, again the recall in predicting 'death' was poor (0.10-0.40). In this experiment precision, recall and F1-score remained low (0.02-0.40) for death prediction. We now introduce 'disease' as a feature in the prediction model using sub-sample four, see Table 4. We observed a vast improvement in recall predicting 'death' (0.43-0.86). AdaBoost achieved the highest recall value of 0.86 which was just above Bagging, Extra-Tree and Decision Tree classifiers, all scoring 0.71. Though Bagging did not achieve the highest recall value for deaths its overall death prediction was the best with precision, recall and F1-score at 0.71. Additionally, Bagging successfully predicted 'isolated' cases at a precision (0.72), recall (0.84) and F1-score (0.77). 'Isolated' and 'death' prediction facilitates urgent

treatment targeted at high risk individuals. This model minimizes the number of false negatives (FN) in death predictions so patients needing adequate medical attention are accurately identified. It must be noted that even though k-NN recall value was the lowest (0.43), it improved by more than a threefold (0.43) over its previous performance when 'disease' was not part of the model.

6 Discussion

Machine learning techniques have been applied to the challenging problem of early prediction of mortality of intensive care unit (ICU) patients [36]. A patient's healthcare utilization pattern may provide more precise estimates of risk for adverse events (AE) or death [37]. To do this prediction a machine learning technique is used to predict the risk of AE or death within 90 days of surgery. In another study electronic medical records (EMR) supports the development of machine learning techniques for predicting disease incidence, patient response to treatment, and other healthcare events [38]. The machine learning model is used to optimize performance of predicting mortality and ICU stay time. Experiments in [39] showed that machine-learning approaches applied to raw electronic health records (EHR) data can be used to build models for use in research and medical practice. These approaches can identify novel predictive variables and their effects to inform future research in predicting patient mortality for coronary artery disease. The mortality rate of the novel 2019-nCoV continues to rise and we showed that machine learning techniques have demonstrated their usefulness for predictions in 2019-nCoV.

From our experiment we noted the vast improvement in prediction performance using 'disease' in the model. This increase in the performance of these machine learning techniques is an indication of the high importance of including patient health information in 2019-nCoV cases. This would help clinicians to better predict the worst outcome of a patient with the disease. By using these predictions better health-care measures can be targeted to those in need. This can result in a much higher increase in the number of 'recovered' cases. Additional datasets can strengthen these models in the future as more data becomes available. However, we note that though 1.92% of the cases resulted in death from dataset2, AdaBoost was still able to have a significant recall value of 0.86 and Bagging 0.71.

The AdaBoost ensemble model is used to classify and make accurate and reliable predictions for in-hospital mortality among patients with pancreatic cancer who undergo pancreatic resection [40]. In [41] Bagging is one of the techniques used to predict if a United States heatwave is likely high-mortality or moderate. The Bagging ensemble model performed admirably but there were suggestions for improvement. Another study in [42] observed that in-hospital mortality of elective patients[1] is low, because these admissions do not lead to an emergency or urgent admission. Nonetheless, there are still some cases of death for elective admission in hospitals. The researchers developed a technique by using machine learning-based models to predict death for the case of elective admissions. Bagging with the highest AUC can be considered to correspond to excellent discriminating performance. Adaboost and Bagging models were effective in 'death' prediction for 2019-nCoV.

[1] An elective procedure is one that is chosen (elected) by the patient or physician that is advantageous to the patient but is not urgent. Elective surgery is decided by the patient or their doctor.

This result is spectacular and prompts immediate interest in the fruitfulness of using the Bagging[2] model built on sub-sample four in other 2019-nCoV datasets. At the time of writing this paper many more deaths have been reported than are used in these experiments. This is not publicly available but these experiments showed including 'disease' in datasets improves the performance of models using machine learning techniques in 'death' prediction. This can be very valuable to clinicians in allocating treatment to 2019-nCoV patients. By utilizing future datasets or current with additional data the result can reduce the burden on health care systems worldwide.

6.1 Generalizations of the AUC for the multi-class setting

A receiver operating characteristic (ROC), or simply ROC curve, is a graphical plot that illustrates the performance of a binary classification system as its threshold for discrimination varies. It is created by plotting the fraction of true positive from the positive (TPR = true positive rate) versus the fraction of false positives from the negative (FPR = false positive rate) at different thresholds [43]. TPR is also known as sensitivity, and FPR is one minus the specificity or true negative rate [44]. The computed area under the curve (AUC) the ROC is simply the mean AUC from all pairwise class comparisons [45]. We now discuss this generalization and how it is used in our analysis. We assume classes are labelled as 0, 1, 2, 3, $\cdots s-1$ with $s > 2$. In order to generalize the AUC we find class pairs (a, b). A good classifier should assign a high probability to the correct class, while the other classes are assigned low probabilities. Let $\hat{C}(a|b)$ provide the chance of a randomly drawn member of class b having a greater probability for class a than a randomly drawn member of class a. We use the following definitions to compute $\hat{C}(a|b)$:

- $\hat{p}(x_q)$ is the estimated probability that observation (x_q originates from class a.
- For all class a observations (x_q, let $d_m = \hat{p}(x_q)$ be the estimated probability of belonging to class a.
- For all class b observations (x_q, let $e_m = \hat{p}(x_q)$ be the estimated probability of belonging to class a.

Next, the combined set of values e_1,$\cdots$,e_{n_b}, d_1,$\cdots$,d_{n_a} is ranked in increasing order. Let k_m be the rank of the $q-th$ observation from class a. Then, the cumulative number of cases at which the b point class has a smaller approximate chance of belonging to the a class than the a point class follows:

$$\sum_{m=1}^{n_a} k_m - k = \sum_{m=1}^{n_a} k_m - \sum_{m=1}^{n_a} k = U_a - n_a(n_a+1)/2 \tag{5}$$

where U_a is the sum of the ranks from the class a samples. Because there are $n_0 n_1$ pairs of points from two classes, the probability that a randomly chosen class b point has a lower estimated probability of belonging to class a than a randomly chosen class a point is:

$$\hat{C}(a|b) = \frac{U_a - n_a(n_a+1)/2}{n_a n_b} \tag{6}$$

[2]Bagging has better general performance.

Since we cannot distinguish $\hat{C}(a|b)$ from $\hat{C}(b|a)$, we define:

$$\hat{C}(a|b) = \frac{1}{2}(\hat{C}(a|b) + \hat{C}(b|a)) \tag{7}$$

as the measure for the separability for classes a and b. The overall AUC of a multi-class classifier (MCC) is then given by the average value for $\hat{C}(a|b)$:

$$MCC = \frac{2}{s(s-1)} \sum_{a<b} \hat{C}(a|b) \tag{8}$$

Here, the multiplier is $\frac{2}{s(s-1)}$ because there are $s(s-1)$ ways in which distinct pairs can be constructed taking different orderings into account. Since only half of these pairs are calculated the value of the enumerator is 2.

We determined multi-class AUC-ROC scores for each model in our experiments. As AdaBoost and Bagging were the best models we only present their multi-class AUC-ROC scores. The area under the ROC curve (AUC-ROC) results were considered excellent for AUC values between 0.9-1, good for AUC values between 0.8-0.9, fair for AUC values between 0.7-0.8, poor for AUC values between 0.6-0.7 and failed for AUC values between 0.5-0.6. In our first experiment, "Metrics of machine learning models for two most common outcomes on dataset1," AdaBoost obtained 0.53 and Bagging 0.80. In our second experiment, "Metrics of machine learning models for three most common outcomes on dataset1," AdaBoost obtained 0.68 while Bagging got 0.66. In our third experiment, "Metrics of machine learning models for three most common outcomes on dataset2," AdaBoost obtained 0.79 while Bagging got 0.80. Finally, in our final experiment, "Metrics of machine learning models for two most common and 'disease' outcomes on dataset2," AdaBoost acheived 0.60 while Bagging attained 0.74. For purposes of this study these multi-class AUC-ROC scores supports our choice of Bagging being the best classifier for death prediction.

6.2 Hyper-parameter settings

Each experiment was run 1000 times with varying hyper-parameter value(s). The hyper-parameters were randomly chosen for each run. The best performance for each run based on our specified criteria was kept. Though we experimented on many different hyper-parameter settings for each model to attain an 'optimal' value our attempts were not exhaustive. Thus, other researchers may be able to use hyper-parameter settings which may obtain better results than we did. However, because of the high computational overhead and time limits of achieving this possible outcome these efforts are left as future work. For instance, with AdaBoost the number of weak learners or estimators of 100 was experimentally found to be 'optimal' for this work but using other values with a tweaked learning rate may lead to more encouraging results. This can also apply to the Bagging model by varying the number of estimators and/or random state to values not generated in our experiments.

7 Conclusion

This paper presents the results of using machine learning techniques for building models to predict 2019-nCoV deaths based on patients demographics and health

conditions. AdaBoost and Bagging machine learning models produced the best results in predictions 'death'. These model demonstrates high predictive ability when trained with the disease feature. As further data is accessible these models can be retrained in the future to evaluate if the model accuracy improves. In addition, other features could be used to build new models with these machine learning techniques. This work should provide researchers with possible directions for developing further machine learning predictive models to help fight the 2019-nCoV outbreak. This can have a positive effect on predictive patient treatment and produce a resulting ease to current overburdened healthcare systems worldwide especially with increasing prevalence of second and third wave re-infections in some countries.

*List of Abbreviations
nCoV: novel coronavirus
RNA: ribonucleic acid
k-NN: k-Nearest Neighbour
DT: Decision Tree
NNC: nearest neighbor classifier
TP: True positive
TN: True negative
FN: False negative
FP: False positive
CNN: Convolutional Neural Network
AI: artificial intelligence
SVM: Support Vector Machine
RF: Random Forest
MLPNN: Multilayer Perceptron Neural Networks
UTC: Coordinated Universal Time
KCDC: Korea Centers for Disease Control & Prevention
ICU: intensive care unit
AE: adverse events
EMR: electronic medical records
EHR: electronic health records
ROC: receiver operating characteristic
TPR: true positive rate
FPR: false positive rate
AUC: area under the curve
AUC-ROC: area under the ROC curve

Declarations

7.1 Ethics approval and consent to participate
'Not applicable'

7.2 Consent for publication
'Not applicable'

7.3 Availability of data and materials
Datasets obtained from Kaggle and listed in References section [no. 6 and 7]. Place here for reader convenience.
SudalaiRajkumar: Novel Corona Virus 2019 Dataset. data retrieved March 30, 2020 from Kaggle, https://www.kaggle.com/sudalairajkumar/novel-corona-virus-2019-dataset (2020)
KimHoo: Data Science for COVID-19 in South Korea. data retrieved March 30, 2020 from Kaggle, https://www.kaggle.com/kimjihoo/coronavirusdataset (2020)

7.4 Competing interests
'Not applicable'

7.5 Funding
'Not applicable'

7.6 Authors' contributions
KK: conception and design, methodology, software; searches and selection of literature; analysis and synthesis of data from the included literature; drafting the manuscript. ER: conception and design; analysis and synthesis of data from the included literature; revising the first draft of the manuscript.

7.7 Acknowledgments
'Not applicable'

7.8 Authors' information
Dr. Koffka Khan received the B.Sc., M.Sc., M.Phil., and D.Phil degrees from the University of the West Indies (UWI). He is currently an Assistant Lecturer at UWI and has up-to-date, published numerous papers in journals & proceedings of international repute. His research areas are computational intelligence, routing protocols, wireless communications, information security, adaptive video streaming and machine learning.
Dr. Emilie Ramsahai is a consulting Data Scientist, with more than 20 years industry experience. She is currently working with UWI-Roytec in programme development and course writing. She completed her PhD in Statistics and a Masters in Computer Science, both at the University of the West Indies, where she has also lectured the Big Data and Visualisation course from the Masters in Data Science, offered by the Department of Computing and Information Technology, St Augustine Campus. She also completed her fellowship at the International Centre for Genetic Engineering and Biotechnology (ICGEB) in New Delhi, India and continues to publish and collaborate with a number of researchers in this area.

Author details
[1]Department of Computing and Information Technology, The University of the West Indies, St. Augustine, Trinidad and Tobago. [2]UWI School of Business & Applied Studies Ltd (UWI-ROYTEC), 136-138 Henry Street, 24105 Port of Spain, Trinidad and Tobago.

References

1. World Health OrganizationThe World Health Organization: Coronavirus disease 2019 (COVID-19)Situation Report –76. data retrieved from World Development Indicators, https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200405-sitrep-76-covid-19.pdf?sfvrsn=6ecf0977$_2$(2020)
2. Lu, R., Zhao, X., Li, J., Niu, P., Yang, B., Wu, H., Wang, W., Song, H., Huang, B., Zhu, N., *et al.*: Genomic characterisation and epidemiology of 2019 novel coronavirus: implications for virus origins and receptor binding. The Lancet **395**(10224), 565–574 (2020)
3. Chen, M., Hao, Y., Hwang, K., Wang, L., Wang, L.: Disease prediction by machine learning over big data from healthcare communities. Ieee Access **5**, 8869–8879 (2017)
4. Rodrigues, L.L., Shetty, D.K., Naik, N., Maddodi, C.B., Rao, A., Shetty, A.K., Bhat, R., Hameed, Z.: Machine learning in coronary heart disease prediction: Structural equation modelling approach. Cogent Engineering **7**(1), 1723198 (2020)
5. LaPierre, N., Ju, C.J.-T., Zhou, G., Wang, W.: Metapheno: A critical evaluation of deep learning and machine learning in metagenome-based disease prediction. Methods **166**, 74–82 (2019)
6. SudalaiRajkumar: Novel Corona Virus 2019 Dataset. data retrieved March 30, 2020 from Kaggle, https://www.kaggle.com/sudalairajkumar/novel-corona-virus-2019-dataset (2020)
7. KimHoo: Data Science for COVID-19 in South Korea. data retrieved March 30, 2020 from Kaggle, https://www.kaggle.com/kimjihoo/coronavirusdataset (2020)
8. Khan, K., Sahai, A.: A glowworm optimization method for the design of web services. International Journal of Intelligent Systems and Applications **4**(10), 89 (2012)
9. Hosni, M., Abnane, I., Idri, A., de Gea, J.M.C., Alemán, J.L.F.: Reviewing ensemble classification methods in breast cancer. Computer methods and programs in biomedicine (2019)
10. Wang, F., Li, Z., He, F., Wang, R., Yu, W., Nie, F.: Feature learning viewpoint of adaboost and a new algorithm. IEEE Access **7**, 149890–149899 (2019)
11. Alsouda, Y., Pllana, S., Kurti, A.: Iot-based urban noise identification using machine learning: Performance of svm, knn, bagging, and random forest. In: Proceedings of the International Conference on Omni-Layer Intelligent Systems, pp. 62–67 (2019)
12. Verma, A.K., Pal, S., Kumar, S.: Prediction of skin disease using ensemble data mining techniques and feature selection method—a comparative study. Applied biochemistry and biotechnology **190**(2), 341–359 (2020)
13. Lu, Y., Wang, S., Wang, J., Zhou, G., Zhang, Q., Zhou, X., Niu, B., Chen, Q., Chou, K.-C.: An epidemic avian influenza prediction model based on google trends. Letters in Organic Chemistry **16**(4), 303–310 (2019)
14. Li, X., Wang, L., Sung, E.: Adaboost with svm-based component classifiers. Engineering Applications of Artificial Intelligence **21**(5), 785–795 (2008)
15. Potes, C., Parvaneh, S., Rahman, A., Conroy, B.: Ensemble of feature-based and deep learning-based classifiers for detection of abnormal heart sounds. In: 2016 Computing in Cardiology Conference (CinC), pp. 621–624 (2016). IEEE
16. Hu, W., *et al.*: Novel host markers in the 2009 pandemic h1n1 influenza a virus. Journal of Biomedical Science and Engineering **3**(06), 584 (2010)

17. Lee, S.-J., Xu, Z., Li, T., Yang, Y.: A novel bagging c4. 5 algorithm based on wrapper feature selection for supporting wise clinical decision making. Journal of biomedical informatics **78**, 144–155 (2018)
18. Leo, J., Luhanga, E., Michael, K.: Machine learning model for imbalanced cholera dataset in tanzania. The Scientific World Journal **2019** (2019)
19. Do, T.-N., Lenca, P., Lallich, S., Pham, N.-K.: Classifying very-high-dimensional data with random forests of oblique decision trees. vol. 292
20. Yuan, C., Moayedi, H.: Evaluation and comparison of the advanced metaheuristic and conventional machine learning methods for the prediction of landslide occurrence. Engineering with Computers, 1–11 (2019)
21. Qiang, X., Kou, Z.: Scoring amino acid mutation to predict pandemic risk of avian influenza virus. BMC bioinformatics **20**(8), 288 (2019)
22. Balasundaram, A., Bhuvaneswari, P.: Comparative study on decision tree based data mining algorithm to assess risk of epidemic (2013)
23. Sandhu, R., Gill, H.K., Sood, S.K.: Smart monitoring and controlling of pandemic influenza a (h1n1) using social network analysis and cloud computing. Journal of Computational Science **12**, 11–22 (2016)
24. Nsoesie, E.O., Beckman, R., Marathe, M., Lewis, B.: Prediction of an epidemic curve: A supervised classification approach. Statistical communications in infectious diseases **3**(1) (2011)
25. Bouadma, L., Barbier, F., Biard, L., Esposito-Farese, M., Le Corre, B., Macrez, A., Salomon, L., Bonnal, C., Zanker, C., Najem, C., et al.: Personal decision-making criteria related to seasonal and pandemic a (h1n1) influenza-vaccination acceptance among french healthcare workers. PLoS One **7**(7) (2012)
26. Stein, G., Chen, B., Wu, A.S., Hua, K.A.: Decision tree classifier for network intrusion detection with ga-based feature selection. In: Proceedings of the 43rd Annual Southeast Regional conference-Volume 2, pp. 136–141 (2005)
27. Özkasap, Ö., Genç, Z., Atsan, E.: Epidemic-based approaches for reliable multicast in mobile ad hoc networks. ACM SIGOPS Operating Systems Review **40**(3), 73–79 (2006)
28. Viswanath, P., Sarma, T.H.: An improvement to k-nearest neighbor classifier. In: 2011 IEEE Recent Advances in Intelligent Computational Systems, pp. 227–231 (2011). IEEE
29. Ramsahai, E., Walkins, K., Tripathi, V., John, M.: The use of gene interaction networks to improve the identification of cancer driver genes. PeerJ **5**, 2568 (2017)
30. Chen, A.W.: Predicting adverse drug reaction outcomes with machine learning. International Journal Of Community Medicine And Public Health **5**(901-904), 678 (2018)
31. Randhawa, G.S., Soltysiak, M.P., El Roz, H., de Souza, C.P., Hill, K.A., Kari, L.: Machine learning using intrinsic genomic signatures for rapid classification of novel pathogens: Covid-19 case study. Plos one **15**(4), 0232391 (2020)
32. Ozturk, T., Talo, M., Yildirim, E.A., Baloglu, U.B., Yildirim, O., Acharya, U.R.: Automated detection of covid-19 cases using deep neural networks with x-ray images. Computers in Biology and Medicine, 103792 (2020)
33. Apostolopoulos, I.D., Aznaouridis, S.I., Tzani, M.A.: Extracting possibly representative covid-19 biomarkers from x-ray images with deep learning approach and image data related to pulmonary diseases. Journal of Medical and Biological Engineering, 1 (2020)
34. Mei, X., Lee, H.-C., Diao, K.-y., Huang, M., Lin, B., Liu, C., Xie, Z., Ma, Y., Robson, P.M., Chung, M., et al.: Artificial intelligence–enabled rapid diagnosis of patients with covid-19. Nature Medicine, 1–5 (2020)
35. Wang, Y., Sohn, S., Liu, S., Shen, F., Wang, L., Atkinson, E.J., Amin, S., Liu, H.: A clinical text classification paradigm using weak supervision and deep representation. BMC medical informatics and decision making **19**(1), 1 (2019)
36. Veith, N., Steele, R.: Machine learning-based prediction of icu patient mortality at time of admission. In: Proceedings of the 2nd International Conference on Information System and Data Mining, pp. 34–38 (2018)
37. Ehlers, A.P., Roy, S.B., Khor, S., Mandagani, P., Maria, M., Alfonso-Cristancho, R., Flum, D.R.: Improved risk prediction following surgery using machine learning algorithms. eGEMs **5**(2) (2017)
38. Huang, L., Shea, A.L., Qian, H., Masurkar, A., Deng, H., Liu, D.: Patient clustering improves efficiency of federated machine learning to predict mortality and hospital stay time using distributed electronic medical records. Journal of Biomedical Informatics **99**, 103291 (2019)
39. Steele, A.J., Denaxas, S.C., Shah, A.D., Hemingway, H., Luscombe, N.M.: Machine learning models in electronic health records can outperform conventional survival models for predicting patient mortality in coronary artery disease. PloS one **13**(8) (2018)
40. Velez-Serrano, J.F., Velez-Serrano, D., Hernandez-Barrera, V., Jimenez-Garcia, R., de Andres, A.L., Garrido, P.C., Alvaro-Meca, A.: Prediction of in-hospital mortality after pancreatic resection in pancreatic cancer patients: A boosting approach via a population-based study using health administrative data. PloS one **12**(6) (2017)
41. Anderson, G.B., Oleson, K.W., Jones, B., Peng, R.D.: Classifying heatwaves: developing health-based models to predict high-mortality versus moderate united states heatwaves. Climatic change **146**(3-4), 439–453 (2018)
42. Steele, R., Hillsgrove, T.: Predicting all-condition, in-hospital mortality of elective patients at time of scheduling. In: 2019 SoutheastCon, pp. 1–5 (2019). IEEE
43. Drummond, C., Holte, R.C.: Cost curves: An improved method for visualizing classifier performance. Machine learning **65**(1), 95–130 (2006)
44. Kumar, R., Indrayan, A.: Receiver operating characteristic (roc) curve for medical researchers. Indian pediatrics **48**(4), 277–287 (2011)
45. Hand, D.J., Till, R.J.: A simple generalisation of the area under the roc curve for multiple class classification problems. Machine learning **45**(2), 171–186 (2001)

Figures
Tables

Figure 1 Distribution of patient age for dataset1. Age frequency distribution histogram plot for dataset1. See dataset1Age.jpg

Figure 2 Distribution of patient age for dataset2. Age frequency distribution histogram plot for dataset2. See dataset2AgeDisease.jpg

Table 1 Metrics of machine learning models for two most common outcomes on dataset1.

Outcome	Metric	AdaBoost	Bagging	Extra-Trees	Decision Tree	k-NN
Alive	Precision	0.95	0.95	0.94	0.95	0.95
	Recall	0.96	0.97	0.97	0.97	0.95
	F1-Score	0.95	0.96	0.95	0.96	0.95
Death	Precision	0.45	0.50	0.44	0.50	0.42
	Recall	0.38	0.38	0.31	0.38	0.38
	F1-Score	0.42	0.43	0.36	0.43	0.40
	Accuracy	0.60	0.92	0.91	0.91	0.91

Table 2 Metrics of machine learning models for three most common outcomes on dataset1.

Outcome	Metric	AdaBoost	Bagging	Extra-Trees	Decision Tree	k-NN
Recovered	Precision	0.29	0.44	0.47	0.38	0.34
	Recall	0.81	0.59	0.56	0.56	0.41
	F1-Score	0.23	0.51	0.51	0.45	0.37
Isolated	Precision	0.82	0.85	0.84	0.84	0.81
	Recall	0.30	0.81	0.83	0.78	0.78
	F1-Score	0.44	0.83	0.83	0.81	0.80
Death	Precision	0.09	0.50	0.44	0.50	0.42
	Recall	0.31	0.38	0.31	0.38	0.38
	F1-Score	0.14	0.43	0.36	0.43	0.40
	Accuracy	0.38	0.74	0.74	0.71	0.69

Table 3 Metrics of machine learning models for three most common outcomes on dataset2.

Outcome	Metric	AdaBoost	Bagging	Extra-Trees	Decision Tree	k-NN
Recovered	Precision	0.34	0.40	0.39	0.39	0.29
	Recall	0.12	0.18	0.12	0.12	0.31
	F1-Score	0.18	0.25	0.19	0.19	0.30
Isolated	Precision	0.62	0.69	0.69	0.69	0.66
	Recall	0.50	0.88	0.91	0.91	0.64
	F1-Score	0.55	0.77	0.78	0.78	0.65
Death	Precision	0.02	0.33	0.33	0.33	0.11
	Recall	0.40	0.10	0.20	0.20	0.10
	F1-Score	0.04	0.15	0.25	0.25	0.11
	Accuracy	0.38	0.65	0.65	0.65	0.53

Table 4 Metrics of machine learning models for two most common and 'disease' outcomes on dataset2.

Outcome	Metric	AdaBoost	Bagging	Extra-Trees	Decision Tree	k-NN
Recovered	Precision	0.22	0.36	0.30	0.30	0.31
	Recall	0.20	0.22	0.11	0.11	0.39
	F1-Score	0.21	0.27	0.16	0.16	0.34
Isolated	Precision	0.66	0.72	0.71	0.71	0.71
	Recall	0.57	0.84	0.88	0.88	0.62
	F1-Score	0.61	0.77	0.78	0.78	0.66
Death	Precision	0.08	0.71	0.56	0.56	0.30
	Recall	0.86	0.71	0.71	0.71	0.43
	F1-Score	0.15	0.71	0.63	0.63	0.35
	Accuracy	0.47	0.66	0.66	0.66	0.55