Predicting the Severity of Incoming COVID-19 Cases in Filipino Health Networks

DS-GA 1003 Final Project

Lauren D'Arinzo (lhd258), Elizabeth Combs (eac721), Angela Marie Teng (at2507) - *responsible*, Paula Kiatkamolwong (kk4158), Steven Dornberg (dornbs01)

Keywords: coronavirus, public health, supervised learning, Philippine healthcare, binary classification, support vector classifier, gradient boosting model

I. Introduction

In 2020, the COVID-19 pandemic upended billions of lives. Like many countries around the globe, the Philippines and its healthcare system have struggled to contain the virus' spread. We implemented a supervised model which, given a vector of input features from historical COVID-19 cases, could predict the severity of a patient's outcome. If successfully trained and deployed, this type of model could allow health systems and policymakers to better anticipate the individual needs of practices and hospitals and better allocate resources based on data-driven research.

We focused on Philippine data from the national Department of Health (DOH) because of its patient-level reporting. Unlike other available datasets, which are typically aggregated, this dataset allows us to answer the research question: Given a presumed COVID-19 diagnosis and features including age, travel history, epidemiological virus links, can we predict a patient's health status in relation to coronavirus? In particular, will they have a mild or severe case?

II. Related Work

As the country with the third highest number of cases in Southeast Asia, the Philippines has been on an enhanced community quarantine since March 12. Few studies have been published addressing the Philippine context of COVID-19, so data-driven prediction using supervised learning could add value. Edrada et al. gave an overview of the spread of the virus across the archipelago, tracing the roots of the first two cases in the Philippines to "previously healthy Chinese nationals on vacation" in January 2020[1]. Additionally, Rabajante et al. used mathematical models to show that exposure time is a significant factor in the spread of the disease[2].

III. Approach

Dataset Description

The dataset[3] we used is a live source gathered by Filipino data scientists and obtained from DOH records of reported and publicly announced[4] COVID-19 cases. The dataset is currently managed by the Philippine Data Science Group. We chose this dataset because it contains patient-level data[5]; all other publicly available datasets related to clinical COVID-19 observations that we were able to find only contain aggregate counts by country or by city.

Limitations of Dataset

However, because COVID-19 is an ongoing challenge, the DOH dataset is constantly being updated[6]. We were restricted to a total sample size of 157 complete instances [7], as many COVID cases since April are documented but comorbidities and other details of the case remain 'for validation.' Additionally, the DOH is notoriously inaccurately collecting data, and case underreporting and lack of testing is a national issue[8]. Thus, although we have a diverse sample of patients, we are uncertain whether or not it comprehensively represents the population of Philippine patients who have tested positive for COVD-19.

Data Preparation

Target Variable

The original target variable of interest was a multi-class variable 'Health Status', which took on increasing values of severity: recovered, asymptomatic, mild, severe, critical, and died. Given our small sample size[9] and even smaller within-class sample sizes,[10] we grouped the response variable into two groups: (1) recovered, asymptomatic, and mild (2) severe, critical, and died. This reduced the uneven distribution of categories leaving 85 "mild" and 72 "severe" cases. After research, we discovered that predicting the general severity of a patient (mild vs. severe) would be more useful in the Philippine context today, compared to predicting the specific gravity of a case[11].

Feature Engineering

Because the dataset is a live document, it required extensive data cleaning and feature engineering to be compatible with supervised learning approaches[12]. For the numeric variables, we used normalization to scale the data and minimize bias, as we were dealing with feature values that

[1] Edrada, Edna M., et al. "First COVID-19 Infections in the Philippines: A Case Report." Tropical Medicine and Health, vol. 48, no. 1, BioMed Central Ltd., Apr. 2020, doi:10.1186/s41182-020-00203-0.

[2] Rabajante, Jomar F. Insights from Early Mathematical Models of 2019-NCoV Acute Respiratory Disease (COVID-19) Dynamics. Feb. 2020, http://arxiv.org/abs/2002.05296.

[3] Data used for this project comes from the Philippine Department of Health (DOH)

[4] These cases identify patients by PH_id, and are publicized in national newspapers like The Inquirer

[5] Each case is at an individual patient level

[6] Since the prioritization in most health networks right now is efficient care, data documentation is lagged.

[7] Where each complete instance is defined as a case where we know the patient outcome (no pending cases)

[8] Many of the COVID-19 tests have been reserved for national officials, or wealthier and socially well-off members of the country.

[9] We also decided to see if there were any naturally-occurring groupings within the categorical response variable.

[10] Was the problem a true multi-class problem or would it be most important to predict mild or severe cases? Using k-means++ algorithm on standardized training features, we grouped the multi-class health status into a binary one by visualizing clustering distributions by class (Figure 2). Furthermore, individuals and practitioners would be most interested in a prediction of whether a COVID case would become severe or not.

[11] This gave us the opportunity to think about how our project could be useful in a business setting

[12] The dataset had both numeric and categorical variables.

ranged from 10 (age) to 1,000,000 (population). To handle the categorical variables, we used dummy variables to create a sparse dataframe, which could then be used for modeling. Prior to one hot encoding, we used OpenRefine[13] to correct any misspellings and manual-entry errors. We used text clustering methods[14] to combine symptoms, comorbidities, other diseases, and locations to ensure the accuracy of our dataset.

Symptoms and Comorbidities

Due to the nature of free text input, medical terminology and grammar mistakes were initially prevalent in our dataset[15]. To compute accurate feature importances, it was imperative that columns did not overlap and were not duplicated. One example of an overlap that could have lead to inaccurate results and incorrect weighting of feature importance would be two different one-hot encoded columns that represent the same family of comorbidities, such as dis_renal_disease and dis_kidney_disease, or dis_cardiovascular_disease and dis_heart_disease[16]. We examined the symptoms and previous diseases to group and rename diseases that are part of the same primary condition (diabetes, heart disease, renal disease).[17] We also characterized similar ailments under their correct parent hierarchy (for instance, mapping cardiac disease and coronary artery disease both under cardiovascular disease)[18].

Epidemiological Link

The original dataset contained a feature called 'Epi Link,' which was a free text field describing any exposure that patient had to other COVID-19 patients. For example, for a given patient, the field might contain "wife of ph42; contact with other patient ph45." We quantified this exposure by counting the number of exposures using regular expressions (regex[19]). Under this method, the example above would have an Epi_Link_Exposure value of 2.

Date Variables and Data Leakage Concerns

There were multiple date fields in the original dataset, such as Date of Onset of Symptoms, Date of Admission, and Date Final Status Was Documented. We originally derived numeric features representing the various permutations of differences in these variables, for example, 'Days between hospital admission and Date of final status.' However, we were hesitant to include these date features as modeling inputs because of the potential data leakage implications; in a deployed version of this model, a hospital would like to predict the outcome of a patient when they are admitted[20]. Consequently, we decided not to use these features explicitly for modeling, but we did explore their distributions in our exploratory data analysis,[21] considering that the conclusion of Rabajante et al. described how exposure time plays a substantial role in the transmission of COVID-19[22].

Geolocation

In the original dataset, there were two main location columns; the city_name, which contained the city location of the patient, and the long_lat, which contained the longitude and latitude of the patient's address. Upon inspection, we discovered that the city column contained potentially misleading and inaccurate information[23]. Thus, we utilized the longitude and latitude information instead. Using GeoPy[24] and GeoPandas[25], we reverse geo-encoded each entry, with a resulting output of a json-object that contained standardized location data. From there, we extracted each city and region to obtain each location's corresponding population data.

Population-level Census Data

Because the dataset we used contained geo-location and patient-level data, we were able to extract additional features related to this information. First, we derived a new feature called average_income, where the value for each entry corresponds with the average income for a particular region that the patient lives. To obtain each city's average income, we used the Philippine Statistics Authority's Census Data[26]. Second, we extracted another variable called population_density where we divided each patient's location into different geographic levels, like regions and cities, and then mapped each patient to their city's population density accordingly.

Missing Value Treatment

Most of the missing values in our dataset occurred in the target variable--because COVID-19 takes about 14 days to incubate, many of the patients in our dataset of ~4000 cases were still hospitalized. We decided to drop the rows which correspond to patients who are still undergoing treatment, as it was not clear whether they had a mild or severe case of COVID-19.

Train/Validation/Test Split

We randomly split our data into 60% train, 20% validation, and 20% test sets for training and experimentation across tree-based and linear algorithms. Typically, we would set the training set as the earliest cases, the validation set as the cases that occured after training, and the test set as the future cases, since the main goal of our project would be to predict COVID-19 case severity given medical and population data.

[13] https://openrefine.org/

[14] For the full process and the various permutations we used to ensure our data was clean, please see the saved OpenRefine project and documentation on our github repo here: https://github.com/angelaaaateng/ML_COVID_PREDICTION

[15] When we apply one-hot encoding, each symptom and previous medical history ailment will become its own binary column.

[16] In order to remedy this, data cleaning and grouping was heavily applied in respect to normalizing and spelling.

[17] We also corrected misspellings.

[18] Data cleaning in these types of situations will aid in the models accuracy and is a necessary part of working with free-text data.

[19] https://regex101.com/

[20] In this scenario, the hospital would not know the 'Date of final status,' because it has not happened yet.

[21] Please see our master.ipynb on our Github for the full EDA

[22] Unfortunately, we did not have access to features that reflected the time of exposure and onset of symptoms.

[23] Various cities in the same country existed in different provinces, moreover, locations in the city column sometimes contained the city, and sometimes contained the municipality.

[24] https://geopy.readthedocs.io/en/stable/

[25] https://geopandas.org/geocoding.html

[26] The CSV files for the raw income data can be obtained here http://www.psa.gov.ph/

However, given our small total sample size of 157 patients[27], our team decided to do a random split across all our samples. Although our dataset may be influenced by time, we decided to apply a random split because of our dataset constraints, as well as our prior assumption that each case was independent, and that the underlying distribution of this specific population was identically-distributed.

IV. Experiments:

Feature Selection

To determine which features contribute the most to our target variable, we ran a feature importance algorithm on the gradient boosted model. After eight features, the dropoff in variance seems marginal (Figure 3) , which prompted us to retrain a model using only the top eight most important features. The decrease in accuracy between a model that contains all 85 features compared to a model that contained only 8 features was 4%. This accuracy decrease is small enough that we decided to continue hyperparameter-tuning on the 8-feature model to prevent overfitting and preserve interpretability.

Since feature importance is only applicable to linear support vector classifiers, we used the relative weight values of each feature instead. We applied dimensionality reduction using an L1 penalty, and the primal form of the support vector classifier. Using the SelectFromModel() and transform() functions, we created a new dataframe with 30 features instead of our original 88. Because the model accuracy only decreased by 2%, we decided to use this new dimensionality-reduced feature space for hyperparameter tuning.

Results of Baselines and other approaches

Using Sklearn implementations[28], we trained the following classification models on training data and evaluated on the validation set:

Generative Methods	Naive Bayes
Tree Based Methods	Decision Tree; Random Forest; Gradient Boosted Trees
Linear Methods	Support Vector Machines [Linear SVC, Kernelized SVC (Linear, Polynomial, Sigmoid, Radial Basis)]; Logistic Regression

Overall, the best performing classifiers were the gradient boosted model, and linear support vector classifiers which we then performed hyperparameter tuning on and achieved test set accuracies of 0.79 for both SVC and the gradient boosted model (Figures 1, 4, 7).

Hyperparameter tuning

To tune the gradient boosting model, we used GridSearch and 3-fold cross-validation. We found the best parameters to be: {'learning_rate': 0.1, 'max_depth': 1, 'min_samples_leaf': 1, 'min_samples_split': 30, 'n_estimators': 30'}. To select the optimal number of n_estimators, we plotted the training and validation accuracy over a range of n_estimators, to observe the change in accuracy as we increased the number of boosting stages performed. After 30 boosting stages, we see only incremental increases in accuracy, which prompted our team to select the optimal n_estimators = 30 in response to efficiency constraints. Overall, our best gradient boosting model achieved a test accuracy of 79%.

To tune the support vector classifier, we also used GridSearch and 3-fold cross-validation. We found the best parameters to be: {'C': 1, 'max_iter': 10}. To select the optimal regularization parameter, we created a plot of the training and validation accuracy over different values of C. Overall, our best support vector classifier achieved a test accuracy of 79%.

Error analysis

Evaluation Metrics

The primary metric we used for evaluation was accuracy, which captures the proportion of all correctly identified cases. Additionally, we also calculated F1 score. As the harmonic mean of precision and recall, the F1 score provides additional evaluation of the incorrectly classified cases. This is particularly important in clinical health applications, because the implications of false negatives are often worse than false positives. In the case of our classifier, a false positive is an instance where a COVID-19 case is predicted to be severe, but is actually mild, and a false negative is an instance where the case is predicted to be mild but is actually severe.

Explanation of Results

Linear Support Vector Classifiers

Support vector machines (SVC)[29] are linear classifiers that construct separating hyperplanes in high-dimensional spaces, and can be used for classification or regression tasks[30]. SVC maximizes the margin by minimizing the L2 norm of the weights vector, subject to a penalty given the corresponding constraint. SVC may work particularly well with our dataset because it is a binary classification problem, and linear kernels may be optimized in this scenario, given that the data distribution is independent and identically distributed. We infer that our classes have a linear boundary because the SVC[31] with a linear kernel outperformed those with a radial basis function, polynomial, and sigmoid kernel[32]. At penalty level 1, the "inverse regularization parameter" gives the highest test accuracy.

Gradient Boosting Model (XGBoost)

Gradient boosting is an "additive model in a forward stage-wise fashion"[33], and is useful for modeling non-linear

[27] The response variable distribution varied significantly over time since new, active cases are appended while past cases' health statuses are finalized. The data generation process is another reason we felt it most accurate to group the data into a binary response.

[28] https://scikit-learn.org/

[29] https://scikit-learn.org/stable/modules/svm.html#svm-kernels

[30] For more information on SVC, and their primal/dual forms, please see the appendix.

[31] https://scikit-learn.org/stable/modules/svm.html#svm-kernels

[32] https://scikit-learn.org/stable/modules/svm.html#classification

[33] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html

prediction problems. Because gradient boosting is a tree-based model, it is generally effective when dealing with datasets that have a lot of categorical features. The parameter n_classes_ determines the number of trees that are fit on the "negative gradient of the binomial or multinomial deviance loss function"[34]. In the case of binary classification, n_classses_ would refer to a single regression tree. By taking an ensemble of weak learners, the XGBoost algorithm optimizes a given cost function by iteratively choosing an approximation that minimizes the expected value of the loss function[35]. Like random forests, gradient boosting uses a set of decision trees--although instead of using the majority rules approach, gradient boosting models build decision trees one at a time, and improves on existing weak learners[36]. This additive approach may be one reason why gradient boosting outperformed random forests in our dataset, and achieved the highest test accuracy out of all the tree-based models.

Discussion:

Evaluation of Feature Importance

We hypothesized that we could predict case severity based on age and pre-existing medical conditions because media coverage and CDC[37] guidelines indicate those groups are at higher risk. Our Gradient Boosting Model Feature Importance confirms this as well. The five key predictors for severity of a COVID-19 patient in descending order are: age, hypertension, diabetes, population density, and travel history (Figures 3,6). This finding also portrays a clear picture of economic disparity where the disease disproportionately affects densely-populated communities more than others. Based on this finding, demographic factors should inform government policy and resources allocation related to treatment and testing.

Model Insights

The results and findings of our data model give us important information regarding our dataset and algorithm choice. For a binary classification problem utilizing medical data, many relationships between age, comorbidities and symptoms are well defined and straightforward, such as hypertension and age. Additionally, we would like to select a model that has high interpretability, such as our chosen SVC and XGBoost models.

In contrast to other models that seek to address this difficult question one factor that distinguishes our model from others[38], is how it does not utilize lab test results to predict severity. Other models in the space, such as another model designed at NYU[39], took into account factors such as elevated liver enzymes and hemoglobin levels. Although our model lacks this information, it is not necessarily a disadvantaged, and can also be viewed as a more interpretable and usable model[40].

Possible Next Steps

While our project focused on the completely validated DOH dataset of Philippine COVID-19 cases, there are other areas where we would like to expand in the future. First, as we receive more data from the other ~1500 unvalidated cases, we would be able to build a more accurate model. Second, in our case, we set our target variable to be a binary classification problem--mild or severe. However, there may be benefit in distinguishing more granular levels COVID-19 cases, and treating this question as a multiclass problem rather than a binary one. The decision to set a binary versus multiclass target variable would depend on further research and discussions with representatives at the DOH, as well as doctors and other healthcare policy makers who are more informed on the context of Philippine COVID-19 occurrence. Third, our end goal for this project would be to build it into a usable tool that Filipino citizens can use as a platform to better understand COVID-19 in the context of Philippine healthcare. Eventually, we would like to create a web app where citizens can input their corresponding health and location information, to determine their level of COVID-19 risk. Fourth, while our current model performed well, and at a better level than random guessing, our team hypothesizes that methods like neural networks[41] and other applications of deep learning could help us attain higher performance metrics. By factoring in feature interactions, using a maximum likelihood approach to the problem, and updating the weights[42] of each feature more accurately, we can utilize interconnected nodes to better the complex relationships that exist in this space[43].

Conclusion:

The COVID-19 pandemic is more nuanced than the features and predictions of our best-performing models, and our attempt to predict the severity of incoming cases in the Philippines is not meant to be a reductionist and definitive approach to conquering this pandemic. Rather, we hope to use this opportunity to innovate a mechanism to understand more about who is most at risk and why, consequently allowing health systems and policymakers to better anticipate needs and allocate resources.

---

[34] https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html

[35] https://en.wikipedia.org/wiki/Gradient_boosting

[36] https://www.datasciencecentral.com/profiles/blogs/decision-tree-vs-random-forest-vs-boosted-trees-explained

[37] https://www.cdc.gov/coronavirus/2019-ncov/need-extra-precautions/people-at-higher-risk.html

[38] mostly due to missing data

[39] https://theconversation.com/we-designed-an-experimental-ai-tool-to-predict-which-covid-19-patients-are-going-to-get-the-sickest-136125

[40] Another factor that distinguishes our model is the large patient dataset, since many medical studies might utilize a sample size of 100 or less.

[41] https://towardsdatascience.com/understanding-objective-functions-in-neural-networks-d217cb068138

[42] http://news.mit.edu/2017/explained-neural-networks-deep-learning-0414

[43] https://en.wikipedia.org/wiki/Neural_network

Appendix:

All code and data for this project is available at https://github.com/angelaaaateng/ML_COVID_PREDICTION

Figure 1. Accuracy results of baseline methods and extended approaches

Model	Validation Accuracy	Test Accuracy	Additional Symptom Feature Engineering Validation Accuracy	Additional Symptom Feature Engineering Test Accuracy
Baseline Naive Bayes	0.6923076923		0.7884615385	
Baseline Decision Tree	0.8269230769		0.7307692308	
Baseline Random Forest	0.7884615385		0.7692307692	
Baseline XGBoost (Gradient Boosting Model)	0.8461538462		0.7884615385	
Baseline Support Vector Machine	0.8269230769		0.8076923077	
Baseline Logistic Regression	0.8076923077		0.7884615385	
Baseline K Nearest Neighbors	0.7692307692		0.7115384615	
Baseline XGBoost on Top 8 Features	0.5384615385		0.7115384615	
Baseline Random Forest on Top 8 Features	0.5384615385		0.6923076923	
Baseline XGBoost (Gradient Boosting Model) with Top 8 Features, with Normalization	0.8076923077		0.8269230769	
GridSearch XGBoost (Gradient Boosting Model) with Top 8 Features, with Normalization	0.8269230769			
GBM Val Accuracy After Grid Search and N-estimators Search and 3-fold CV	0.8269230769		0.8461538462	
GBM Test Accuracy Baseline After Grid Search Final N-estimators Search and 3-fold CV	0.8269230769	0.7169811321	0.8461538462	0.7924528302
Baseline SVC with Linear Kernel	0.83		0.83	
Baseline SVC with Polynomial Kernel	0.54		0.5	
Baseline SVC with Sigmoid Kernel	0.54		0.81	
Baseline SVC with RBF Kernel	0.62		0.81	
Linear SVC with 30 Features	0.8076923077		0.8461538462	
GridSearch Linear SVC with 30 Features	0.8076923077			
SVC Val Accuracy Baseline After Grid Search and 3fold CV	0.8076923077	0.7735849057	0.8076923077	0.7924528302

Figure 2. Results of K-Means Clustering

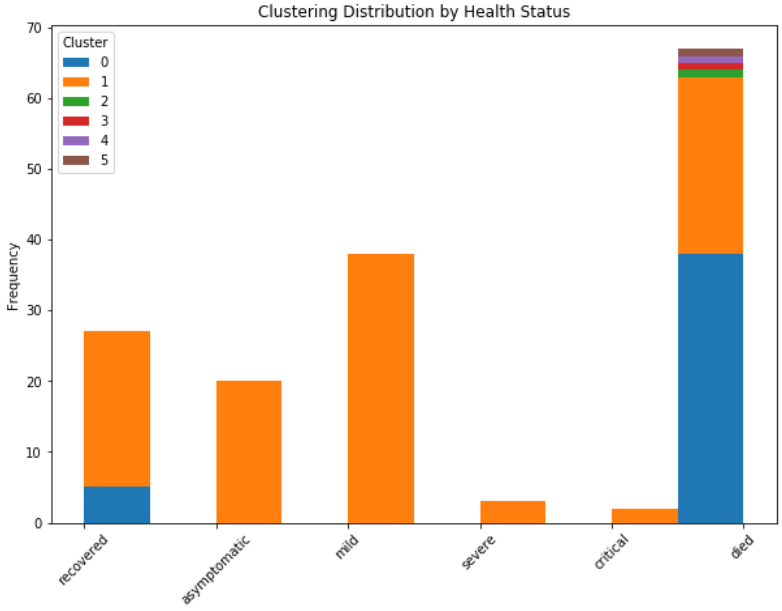


Figure 3. Gradient Boosting to Quantify Feature Importance

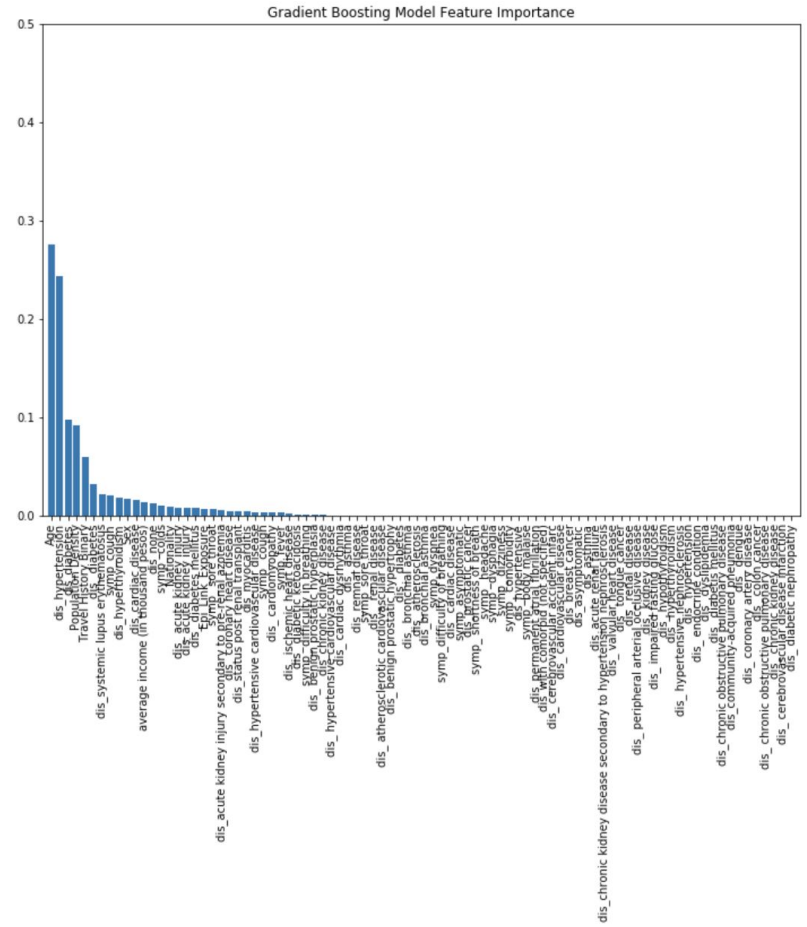


Figure 4. Comprehensive Results of Best Performing Linear Model, Support Vector Machine, on Validation Set

```
Evaluation: Polynomial kernel
              precision    recall  f1-score   support

           0       0.54      1.00      0.70        28
           1       0.00      0.00      0.00        24

    accuracy                           0.54        52
   macro avg       0.27      0.50      0.35        52
weighted avg       0.29      0.54      0.38        52

Evaluation: RBF kernel
              precision    recall  f1-score   support

           0       0.58      1.00      0.74        28
           1       1.00      0.17      0.29        24

    accuracy                           0.62        52
   macro avg       0.79      0.58      0.51        52
weighted avg       0.78      0.62      0.53        52

Evaluation: Sigmoid kernel
              precision    recall  f1-score   support

           0       0.54      1.00      0.70        28
           1       0.00      0.00      0.00        24

    accuracy                           0.54        52
   macro avg       0.27      0.50      0.35        52
weighted avg       0.29      0.54      0.38        52

Evaluation: Linear kernel
              precision    recall  f1-score   support

           0       0.76      1.00      0.86        28
           1       1.00      0.62      0.77        24

    accuracy                           0.83        52
   macro avg       0.88      0.81      0.82        52
weighted avg       0.87      0.83      0.82        52
```

Figure 5. Results of Hyperparameter Tuning, N-Estimators Parameter

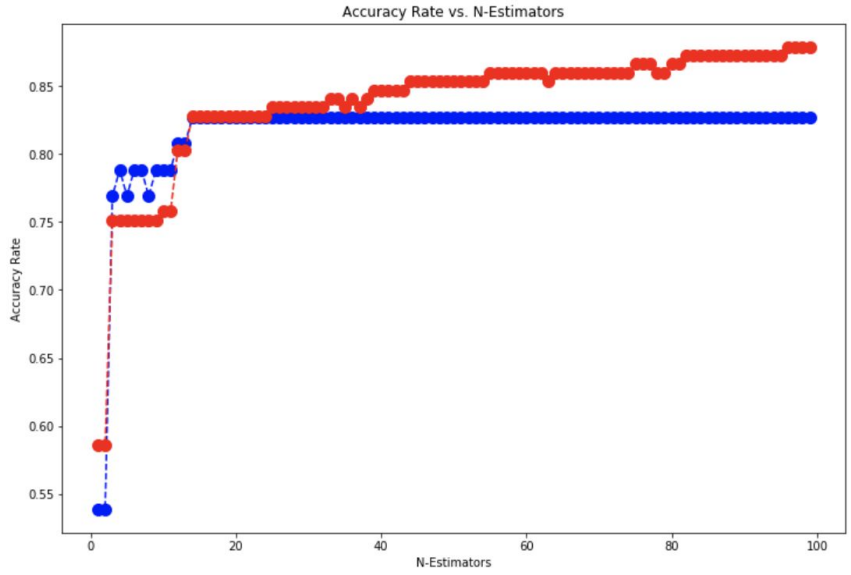
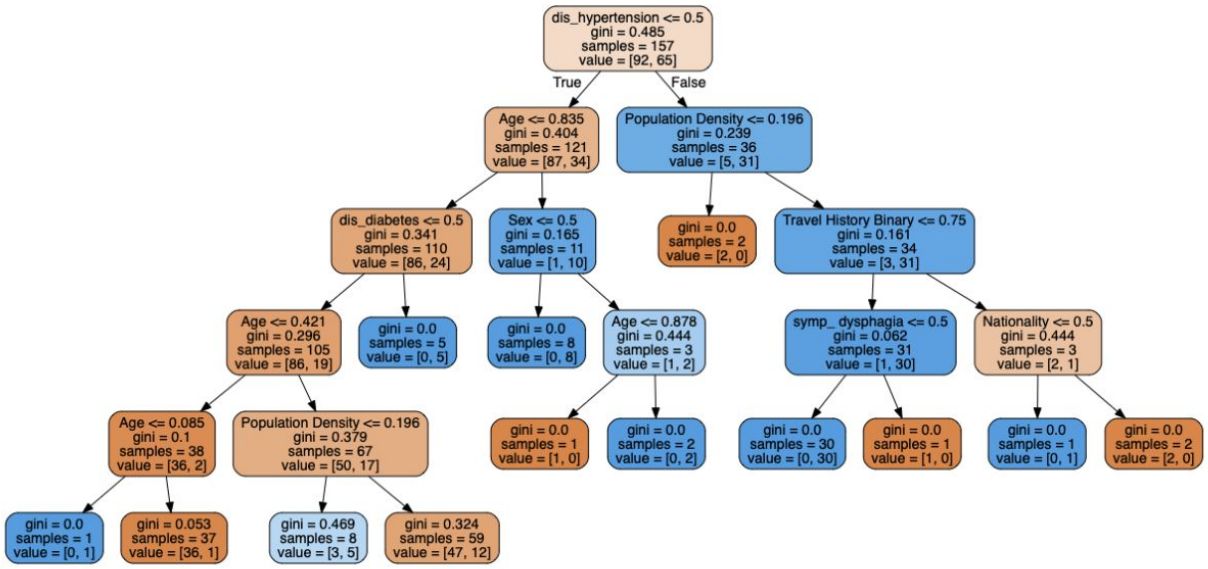


Figure 6. Feature Importance Results of Decision Tree at Max Depth 5 (A) and Max Depth 3 (B)



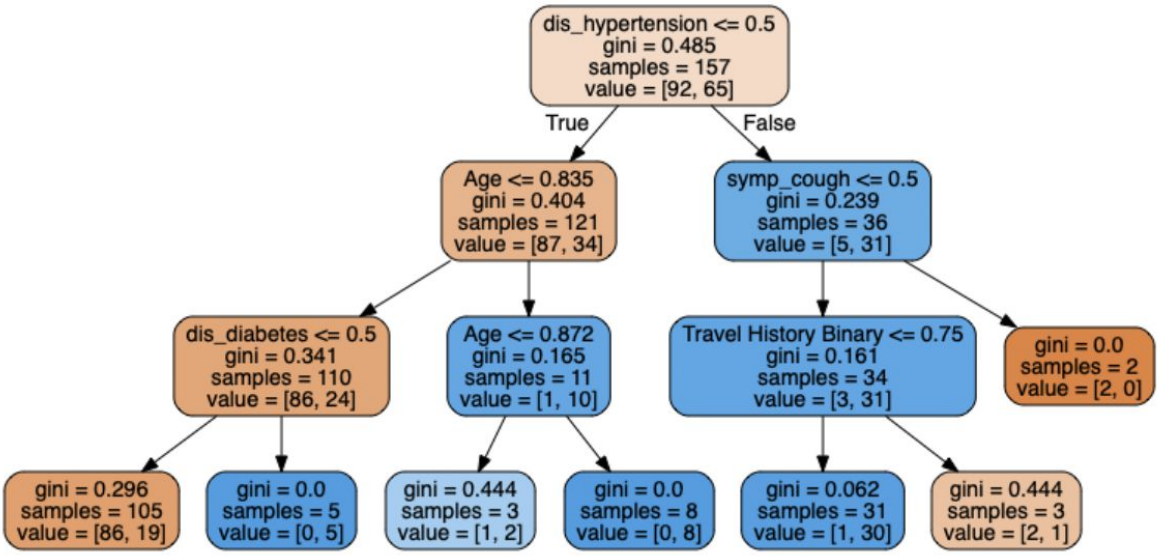


Figure 7. Comprehensive Results of Best Performing Linear Model, Support Vector Machine, on Test Set

Linear SVC Classifiaction Report on Test Set

	precision	recall	f1-score	support
0	0.69	0.96	0.80	25
1	0.94	0.61	0.74	28
accuracy			0.77	53
macro avg	0.82	0.78	0.77	53
weighted avg	0.82	0.77	0.77	53

Gradient Boosting Model Classifiaction Report on Test Set

	precision	recall	f1-score	support
0	0.64	0.92	0.75	25
1	0.88	0.54	0.67	28
accuracy			0.72	53
macro avg	0.76	0.73	0.71	53
weighted avg	0.77	0.72	0.71	53

Figure 8. Pairwise correlations of numeric features

Figure 9. Objective Functions and Primal/Dual Forms:

SVC

The primal problem that SVC[44] solves is the following:

$$\begin{aligned} &\min_{w,b,\zeta} \frac{1}{2} w^T w + C \sum_{i=1}^{n} \zeta_i \\ &\text{subject to } y_i(w^T \phi(x_i) + b) \geq 1 - \zeta_i, \\ &\zeta_i \geq 0, i = 1, \ldots, n \end{aligned}$$

And the dual problem it solves is[45]:

$$\begin{aligned} &\min_{\alpha} \frac{1}{2} \alpha^T Q \alpha - e^T \alpha \\ &\text{subject to } y^T \alpha = 0 \\ &0 \leq \alpha_i \leq C, i = 1, \ldots, n \end{aligned}$$

Bibliography:

Chen, Nanshan, et al. "Epidemiological and Clinical Characteristics of 99 Cases of 2019 Novel Coronavirus Pneumonia in Wuhan, China: A Descriptive Study." *The Lancet*, vol. 395, no. 10223, Lancet Publi

Edrada, Edna M., et al. "First COVID-19 Infections in the Philippines: A Case Report." *Tropical Medicine and Health*, vol. 48, no. 1, BioMed Central Ltd., Apr. 2020, doi:10.1186/s41182-020-00203-0.

Kay, Cherish, and L. Pastor. "Sentiment Analysis on Synchronous Online Delivery of Instruction Due to Extreme Community Quarantine in the Philippines Caused by COVID-19 Pandemic." *Asian Journal of Multidisciplinary Studies*, vol. 3, no. 1, 2020, https://asianjournal.org/online/index.php/ajms/article/view/207.

Kong, Weifang, and Prachi P. Agarwal. "Chest Imaging Appearance of COVID-19 Infection." *Radiology: Cardiothoracic Imaging*, vol. 2, no. 1, Radiological Society of North America (RSNA), Jan. 2020, p. e200028, doi:10.1148/ryct.2020200028.

Nicomedes, CJ, and R. Avila. *An Analysis on the Panic of Filipinos During COVID-19 Pandemic in the Philippines*. 2020, doi:10.13140/RG.2.2.17355.54565.

Rabajante, Jomar F. *Insights from Early Mathematical Models of 2019-NCoV Acute Respiratory Disease (COVID-19) Dynamics*. Feb. 2020, http://arxiv.org/abs/2002.05296.

Schultz, Marcus J., et al. "Current Challenges in the Management of Sepsis in ICUs in Resource-Poor Settings and Suggestions for the Future." *Intensive Care Medicine*, vol. 43, no. 5, Springer Verlag,

Sohrabi, C., et al. "World Health Organization Declares Global Emergency: A Review of the 2019 Novel Coronavirus (COVID-19)." *Elsevier*, https://www.sciencedirect.com/science/article/pii/S1743919120301977. Accessed 15 May 2020.

Sohrabi, Catrin, et al. "World Health Organization Declares Global Emergency: A Review of the 2019 Novel Coronavirus (COVID-19)." *International Journal of Surgery*, vol. 76, Elsevier Ltd, 1 Apr. 2020, pp. 71–76, doi:10.1016/j.ijsu.2020.02.034.

Watanabe, Shumpei, et al. "Bat Coronaviruses and Experimental Infection of Bats, the Philippines." *Emerging Infectious Diseases*, vol. 16, no. 8, 2010, pp. 1217–23, doi:10.3201/eid1608.100208.

[44] https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html
[45] https://scikit-learn.org/stable/modules/svm.html#svm-kernels

Wenham, C., et al. “COVID-19: The Gendered Impacts of the Outbreak.” *Thelancet.Com*, https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(20)30526-2/fulltext?te=1&nl=in-her words&emc=edit_gn_20200317. Accessed 15 May 2020.

Wu, YC, et al. “The Outbreak of COVID-19: An Overview.” *Ncbi.Nlm.Nih.Gov*, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7153464/. Accessed 15 May 2020.

Wu, Yi Chi, et al. “The Outbreak of COVID-19: An Overview.” *Journal of the Chinese Medical Association*, vol. 83, no. 3, Wolters Kluwer Health, 2020, pp. 217–20, doi:10.1097/JCMA.0000000000000270.

Complete Machine Learning Guide to Parameter Tuning in Gradient Boosting (GBM) in Python. https://www.analyticsvidhya.com/blog/2016/02/complete-guide-parameter-tuning-gradient-boosting-gbm-python/.

1.13. Feature Selection. https://scikit-learn.org/stable/modules/feature_selection.html.

Sklearn.Svm.SVC¶. https://scikit-learn.org/stable/modules/generated/sklearn.svm.SVC.html.

1.4. Support Vector Machines. https://scikit-learn.org/stable/modules/svm.html#classification.

SVM Hyperparameter Tuning Using GridSearchCV. https://towardsdatascience.com/svm-hyper-parameter-tuning-using-gridsearchcv-49c0bc55ce29.

1.4.6. Kernel Functions.

Sklearn.Svm.LinearSVC — Scikit-Learn 0.23.0 Documentation. https://scikit-learn.org/stable/modules/generated/sklearn.svm.LinearSVC.html. Accessed 16 May 2020.

1.13. Feature Selection — Scikit-Learn 0.23.0 Documentation. https://scikit-learn.org/stable/modules/feature_selection.html. Accessed 16 May 2020.

B’Comparison of Different Linear SVM Class | Diksha_Gabha | Plotly’. https://chart-studio.plotly.com/~Diksha_Gabha/3579.embed. Accessed 16 May 2020.

SVM: Feature Selection and Kernels - Towards Data Science. https://towardsdatascience.com/svm-feature-selection-and-kernels-840781cc1a6c. Accessed 16 May 2020.

Feature Selection Techniques in Machine Learning with Python. https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-python-f24e7da3f36e. Accessed 16 May 2020.

1.11. Ensemble Methods — Scikit-Learn 0.23.0 Documentation. https://scikit-learn.org/stable/modules/ensemble.html#gradient-boosting. Accessed 16 May 2020.

3.2.4.3.5. Sklearn.Ensemble.GradientBoostingClassifier — Scikit-Learn 0.23.0 Documentation. https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html. Accessed 16 May 2020.

Feature Importance and Feature Selection With XGBoost in Python. https://machinelearningmastery.com/feature-importance-and-feature-selection-with-xgboost-in-python/. Accessed 16 May 2020.

4.2. Permutation Feature Importance — Scikit-Learn 0.23.0 Documentation. https://scikit-learn.org/stable/modules/permutation_importance.html. Accessed 16 May 2020.

Breiman, Leo. “Random Forests.” *Machine Learning*, vol. 45, no. 1, Oct. 2001, pp. 5–32, doi:10.1023/A:1010933404324.

Sklearn.Preprocessing.StandardScaler — Scikit-Learn 0.23.0 Documentation. https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.StandardScaler.html. Accessed 16 May 2020.

Normalization vs Standardization — Quantitative Analysis. https://towardsdatascience.com/normalization-vs-standardization-quantitative-analysis-a91e8a79cebf. Accessed 16 May 2020.

Preprocessing with Sklearn: A Complete and Comprehensive Guide. https://towardsdatascience.com/preprocessing-with-sklearn-a-complete-and-comprehensive-guide-670cb98fcfb9. Accessed 16 May 2020.

Machine Learning - Should You Ever Standardise Binary Variables? - Cross Validated. https://stats.stackexchange.com/questions/59392/should-you-ever-standardise-binary-variables. Accessed 16 May 2020.

Sklearn.Impute.SimpleImputer — Scikit-Learn 0.23.0 Documentation. https://scikit-learn.org/stable/modules/generated/sklearn.impute.SimpleImputer.html. Accessed 16 May 2020.

How, When and Why Should You Normalize / Standardize / Rescale Your Data? https://medium.com/@swethalakshmanan14/how-when-and-why-should-you-normalize-standardize-rescale-your-data-3f083def38ff. Accessed 16 May 2020.

Liu, Qinghe, et al. *Assessing the Global Tendency of COVID-19 Outbreak*. doi:10.1101/2020.03.18.20038224. Accessed 15 May 2020.

Wenham, C., et al. “COVID-19: The Gendered Impacts of the Outbreak.” *Thelancet.Com*, https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(20)30526-2/fulltext?te=1&nl=in-her words&emc=edit_gn_20200317. Accessed 15 May 2020.

Kay, Cherish, and L. Pastor. “Sentiment Analysis on Synchronous Online Delivery of Instruction Due to Extreme Community Quarantine in the Philippines Caused by COVID-19 Pandemic.” *Asian Journal of Multidisciplinary Studies*, vol. 3, no. 1, 2020, https://asianjournal.org/online/index.php/ajms/article/view/207.

Wu, YC, et al. “The Outbreak of COVID-19: An Overview.” *Ncbi.Nlm.Nih.Gov*, https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7153464/. Accessed 15 May 2020.

Sohrabi, C., et al. “World Health Organization Declares Global Emergency: A Review of the 2019 Novel Coronavirus (COVID-19).” *Elsevier*, https://www.sciencedirect.com/science/article/pii/S1743919120301977. Accessed 15 May 2020.

Kong, Weifang, and Prachi P. Agarwal. “Chest Imaging Appearance of COVID-19 Infection.” *Radiology: Cardiothoracic Imaging*, vol. 2, no. 1, Radiological Society of North America (RSNA), Jan. 2020, p. e200028, doi:10.1148/ryct.2020200028.

Nicomedes, CJ, and R. Avila. *An Analysis on the Panic of Filipinos During COVID-19 Pandemic in the Philippines*. 2020, doi:10.13140/RG.2.2.17355.54565.

Roman, Adriel. “Minimizing Plagiarism Incidence in Research Writing in One State University in the Philippines.” *Asian Journal of Multidisciplinary Studies*, vol. 1, no. 3, Mar. 2018, pp. 1–7, https://www.asianjournal.org/index.php/ajms/article/view/141.

Wu, Yi Chi, et al. “The Outbreak of COVID-19: An Overview.” *Journal of the Chinese Medical Association*, vol. 83, no. 3, Wolters Kluwer Health, 2020, pp. 217–20, doi:10.1097/JCMA.0000000000000270.

Rabajante, Jomar F. *Insights from Early Mathematical Models of 2019-NCoV Acute Respiratory Disease (COVID-19) Dynamics*. Feb. 2020, http://arxiv.org/abs/2002.05296.

Schultz, Marcus J., et al. “Current Challenges in the Management of Sepsis in ICUs in Resource-Poor Settings and Suggestions for the Future.” *Intensive Care Medicine*, vol. 43, no. 5, Springer Verlag, 1 May 2017, pp. 612–24, doi:10.1007/s00134-017-4750-z.

Chen, Nanshan, et al. “Epidemiological and Clinical Characteristics of 99 Cases of 2019 Novel Coronavirus Pneumonia in Wuhan, China: A Descriptive Study.” *The Lancet*, vol. 395, no. 10223, Lancet Publishing Group, Feb. 2020, pp. 507–13, doi:10.1016/S0140-6736(20)30211-7.

Sohrabi, Catrin, et al. “World Health Organization Declares Global Emergency: A Review of the 2019 Novel Coronavirus (COVID-19).” *International Journal of Surgery*, vol. 76, Elsevier Ltd, 1 Apr. 2020, pp. 71–76, doi:10.1016/j.ijsu.2020.02.034.

Edrada, Edna M., et al. “First COVID-19 Infections in the Philippines: A Case Report.” *Tropical Medicine and Health*, vol. 48, no. 1, BioMed Central Ltd., Apr. 2020, doi:10.1186/s41182-020-00203-0.

Watanabe, Shumpei, et al. “Bat Coronaviruses and Experimental Infection of Bats, the Philippines.” *Emerging Infectious Diseases*, vol. 16, no. 8, 2010, pp. 1217–23, doi:10.3201/eid1608.100208.