Predicting Patient Health: A Machine Learning Approach

Elena Robles, Mateo Elosua, Margherita Tonon

November 2024

IE University

1 Introduction

Predictive analysis plays a critical role in early detection and prevention of diseases. Medical history and lifestyle habits are driving factors in health outcomes; however, the extent to which each risk factor weighs in on illnesses is harder to determine. Predicting patient risk can help ensure long-term patient health, reduce costs by avoiding expensive medical tests, and allow more effective time allocation of medical professionals.

Machine learning (ML) methods are increasingly used for analyzing healthcare datasets. Applications include diagnosing diseases, predicting treatment responses, and assessing risk factors [1]. Nonetheless, models can lack accuracy due to biased datasets and inadequate algorithm choices.

The objective of this study is to predict whether an individual is likely to be healthy or unhealthy based on various lifestyle factors. The outcomes of this study could advance predictive models in healthcare, aiding in preventive medical care. Identifying which risk factors are the most likely to reduce health could be used in educational campaigns to promote healthy lifestyles, shifting the focus to disease prevention rather than disease treatment.

2 Methodology

2.1 Data Processing

The selected dataset for the investigation is centered around patient health [2]. Among the features in the dataset are binary valued features indicating whether a person had a certain disease (1) or not (0). Diseases included heart attack, stroke, angina, asthma, skin cancer, chronic obstructive pulmonary disease, diabetes, depressive disorder, kidney disease, and arthritis. Adding up these binary-valued features for every individual, we created a health severity score ranging from 0 to 10. We created two classes from the health severity score, letting scores 0 to 4 be class 0, representing healthy people, and scores 5 to 10 be class 1, representing unhealthy people. The result of this separation led the data to become imbalanced, having 231023 samples for class 0 and 6607 samples for class 1.

To classify individuals into class 0 or class 1, we utilized both categorical and continuous features, namely: Sex, Age Category, Smoker Status, Alcohol Drinker, General Health, Diabetes, and BMI.

2.2 Algorithm Selection

We applied decision trees, logistic regression, and neural networks with the sigmoid activation function to the transformed dataset.

Decision trees divide the predictor space into simple regions using the dataset features, taking a top-down, greedy approach. A random forest is an ensemble of decision trees trained on bootstrapped datasets of the training data. At each possible tree split, a random subset of the k total features is considered, with the default size being approximately $\sqrt{k}$. The change in the features considered at each split introduces variability to the decision trees included in the ensemble. A majority vote of all tree classifications provides the forest's final classification of the given sample. We applied random forests to the dataset because of their high interpretability, given that seeing which features are selected most often at each split emphasizes which features play the most important role in classification. Moreover, they are robust to overfitting due to their aggregation of results, reducing the variance between training and test sets.

Logistic regression estimates the probability of a binary outcome. To classify an instance as class 1, the probability of the instance belonging to class 1 must exceed a predefined threshold, by default set to 0.5. The binary focus of this algorithm makes it suitable for the healthy and unhealthy classes in our model. Moreover, logistic regression is also highly interpretable, as we can observe the coefficients pertinent to each feature to see the power each feature has in classification [3].

A neural network is a model in which layers of artificial neurons are connected via activation functions, where a weight and bias is assigned to each of the connections between neurons of the current layer and neurons of the next layer. The network is able to automatically tune these weights and biases to achieve a desired outcome. Neural networks are advantageous due to their ability to capture more intricate patterns via hidden layers. Because the sigmoid function is used in logistic regression, we selected the sigmoid as the activation function for the neural networks to compare whether a highly complex neural network would perform better than a single-layered logistic regression model [4].

2.3 Algorithm Application and Evaluation Metrics

First, we applied all algorithms to the imbalanced data as a baseline test for model performance [5]. Then, random under-sampling was used to resample the majority class (class 0) to have the same number of samples as the minority class (class 1) [6].

We evaluated model performance on the recall of class 1 rather than training and test errors. Recall of class 1 is calculated with the confusion matrix. In binary classification, confusion matrices are 2 × 2 matrices, where the first row represents the samples that belong to class 0 (healthy), and row 2 represents the samples that belong to class 1 (unhealthy). Column 1 represents the samples that are classified as class 0, and column 2 represents the samples that are classified as class 1. Therefore, element (1,1) of the matrix represents the true negatives, element (1,2) represents false negatives, element (2,1) represents false positives, and element (2,2) represents true positives. In the scenario of predicting patient health, a true positive would represent telling a patient they are sick when they are in reality sick, and a true negative would be telling a patient they are not sick when they are in reality not sick. On the other hand, a false negative would be predicting a patient is healthy when they are in reality unhealthy, and a false positive would be predicting a patient is unhealthy when they are in reality healthy.

Recall is calculated by dividing the true positives by the sum of the true positives and false negatives, and ranges between 0 and 1 [7]. We want to minimize the number of false negatives, and thus maximize the recall, as we do not want our model to predict a patient is healthy when they are not, potentially leading them to neglect their deteriorating health.

Precision goes hand-in-hand with recall, calculated by dividing the true positives by the sum of true positives and false positives. Like recall, precision also ranges between 0 and 1. Precision represents how many of the predicted class 1 instances truly belong to class 1: of the patients classified as unhealthy, what proportion of them were truly unhealthy? Although we do not directly aim to maximize precision, it indicates how well the model balances the predictions of the two classes [7].

Another useful metric is the Area Under the Precision and Recall Curve (AUPRC). Decision trees, logistic regression, and neural networks all utilize a threshold which they base their classifications on; when a value greater than the imposed threshold is predicted by the model, the sample is classified as belonging to class 1, and when a value less than the threshold is predicted, the sample is classified as belonging to class 0. The thresholds, which range between 0 and 1 for the selected algorithms, are conceptually similar to decision boundaries. The PRC plots precision versus recall for different thresholds, showing how changing the threshold affects the model's precision and recall. The ideal PRC is a straight line from (0, 1) to (1, 1), indicating that perfect precision and perfect recall can be obtained simultaneously. The AUPRC measures how well the model balances precision and recall: 0 represents the model performs no better than random chance, and 1 represents perfect recall and perfect precision can be achieved simultaneously [8].

3 Results and Discussion

To compare the performance of the three chosen algorithms, we observed the recall of class 1, the PRC, and the AUPRC. Because the PRC plots the values of precision and recall based on different thresholds used for classification, we used the PRC to extract thresholds given desired recall values. Specifically, we looked at thresholds for recall 1 and recall 0.7. Any recalls, precisions, and AUPRCs were obtained by taking the mean of results from 5-fold cross validation (CV).

3.1 Decision Trees

We utilized random forests with 50 trees, selecting this number based on the plot of class 1 recall versus number of trees and identifying where the recall started to stabilize. We focused on controlling the length of the trees by varying the `min_samples_leaf` hyperparameter in the `RandomForestClassifier` class in Scikit-Learn [9]. This hyperparameter dictates the minimum number of samples that must be present in each tree leaf. We compared trees with 1, 10, 100, and 1000 minimum samples.

First, we applied random forests to the imbalanced data. As the tree complexity increased, the number of samples predicted to be in class 1 also increased. With 1000 and 100 minimum number of samples per leaf, there were no true positives and no false positives present in any confusion matrices - the forests were only predicting class 0, leading to a recall of 0. With 10 minimum samples, some forests did not predict class 1 at all while others predicted class 1 at a lower rate than class 0. These forests had a recall of 0.0369 ± 0.00483. The forests with 1 minimum sample per leaf had a mean recall of 0.152 ± 0.0076, and they were able to predict class 1 more often, although class 0 was still predicted the most (Appendix 6.1). Although having 1 sample per leaf performed better than other hyperparameter values, this random forest only identified around 15% of class 1 correctly. This is low, especially as we want to ensure all unhealthy patients are told they are unhealthy - not just 15% of them. This model obtained a mean precision of 0.2094 ± 0.0144, indicating that only 20% of the patients predicted as being unhealthy are truly unhealthy, consequently indicating the model suffers from a high rate of false positives when the threshold is set to 0.5.

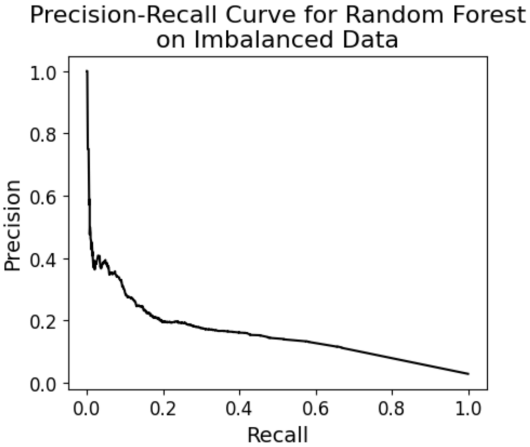

Figure 1: PRC for a random forest with 50 trees and minimum 1 sample per leaf, fitted on the imbalanced data.

Figure 1 depicts the PRC for a random forest with `min_samples_leaf = 1`, the random forest which performed the best on the imbalanced data. The steep decline when recall is close to 0 shows that the random forest obtains high precision at low recall, and high recall at low precision. When the recall is 1, the precision is almost 0; at this threshold, the model mainly only predicts class 1, thus producing a high number of false positives, decreasing the precision. The mean AUPRC was 0.137 ± 0.00611. A model purely based on random chance would have an AUPRC of 0, indicating this random forest performs better than random chance, yet is still far from optimal given that the maximum AUPRC value is 1.

Extracting relevant thresholds, the threshold for a recall of 1 was 0, meaning all examples must be classified as class 1 regardless of their true class to achieve a perfect recall. To achieve a recall of 0.7, a threshold of 0.0025 is needed. This signifies the model does not need to be highly confident that a sample is class 1 to classify it as class 1, as it only needs to see that the probability of a sample being in class 1 exceeds 0.0025. For both recall 1 and 0.7, the model classifies many healthy patients as sick, and while many unhealthy patients would be correctly classified, it could lead healthy patients to worry unnecessarily.

Random forests trained on imbalanced data were therefore not successful in terms of recall, AUPRC, and possible alternative decision boundaries.

Somewhat different relationships emerged when training random forests on the balanced data. Trees with 1 minimum sample per leaf performed the worst, with a recall of 0.801 ± 0.0126. There was almost no significant difference in recall between decision trees with 10, 100, and 1000 minimum samples per leaf, with mean recalls of 0.866 ± 0.00935, 0.863 ± 0.0143, and 0.843 ± 0.00835, respectively. Trees with 10 minimum samples per leaf performed slightly better than the trees with 1000 minimum samples per leaf. All trees with

10, 100, and 1000 minimum samples performed significantly better than random forests with 1 minimum sample per leaf. However, confusion matrices continued to not have elements centered on the leading diagonal - the ideal behaviour to achieve high precision and recall (Appendix 6.2).

The AUPRC followed a similar pattern, where forests with 1 minimum sample performed worse than forests with 10, 100, and 1000 samples. The mean AUPRC's for trees with minimum leaf samples 1, 10, 100, and 1000 were, respectively, 0.166 ± 0.0148, 0.234 ± 0.00838, 0.248 ± 0.0149, and 0.225 ± 0.0127.

Random forests performed significantly better in terms of recall of class 1 when trained on the balanced data than on the imbalanced data. When the data is balanced, both classes contribute equally to the splitting criteria at each node of the tree, ensuring the majority class 0 does not have a more predominant influence than class 1. However, the AUPRC's did not drastically differ because the overall shape of the curve remained the same (Figure 2), indicating that the precision-recall balance did not change. Even though balancing the data led to models achieving higher recall, false positives increased at the same rate as true positives, not causing an improvement in the AUPRC.

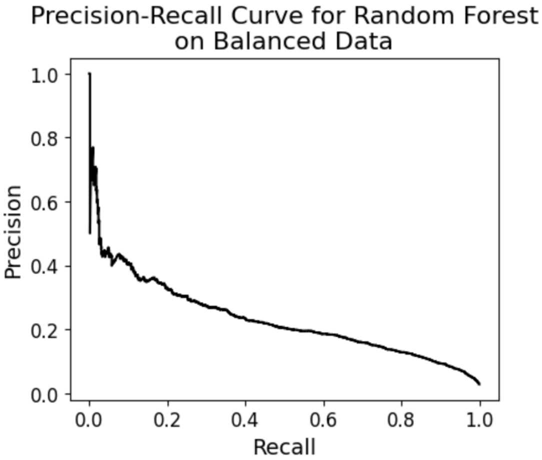

Figure 2: PRC for a random forest with 50 trees and minimum 10 samples per leaf, fitted on the balanced data.

We extracted thresholds based on the PRC for the balanced random forest. For recall to be 1, we obtained a threshold of 0.005, which is higher than the threshold we had obtained for the imbalanced data for a recall of 0.7. However, this value is still low, as it indicates that a sample only needs a 0.5% probability of belonging to class 1 to be classified as class 1. For the recall to be 0.7, we obtained a threshold of 0.700; this indicates the model is less biased towards predicting class 0, and can afford to classify samples as class 1 when it is 70% certain that the samples belong to class 1. Actually using the 0.05 threshold to classify, and looking at precision and recall using 5-fold CV, we obtained a recall between 1 and 0.99, as expected. However, the precision was 0.0304 ± 0.000382: only 3% of the samples classified as class 1 are actually true class 1 samples, indicating a high false positive rate. When using the 0.65 threshold, we obtained a recall between 0.69 and 0.7 with a larger precision 0.163 ± 0.00249. The inverse relationship between precision and recall is evident here, emphasizing how we are maximizing recall at the expense of precision.

Random forests can be used to extract the importance of different features. Figure 3 depicts the feature importance for the random forest with 10 minimum leaf samples trained on the balanced data. The feature importances were found using the `model.feature_importances_` attribute in Scikit-learn, which orders features based on how significantly they decrease impurity across splits in a tree [10].

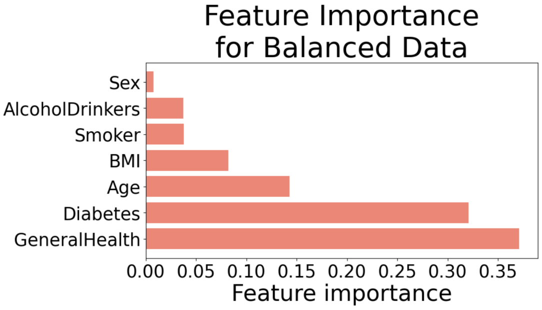

Figure 3: Feature importance for optimal random forest trained on balanced data.

General Health was the feature most indicative of whether someone was classified in class 1 or 0. This attribute ranking first is intuitive, providing a check that the random forest is classifying in a rational way that aligns with what would be expected by a doctor in the real world. Diabetes and Age closely follow General Health; doctors can identify these risk factors in patients and easily distinguish whether someone is likely to be unhealthy. Logically, these features should not be used as definite testing or immediate discrimination but could still provide useful insights to medical professionals.

3.2 Logistic Regression

When applying logistic regression to the imbalanced data using the `LogisticRegression` class on Scikit-Learn [11], the mean recall was 0.0572 ± 0.00855, which is lower than the recall obtained for decision trees trained on the balanced data with the standard threshold of 0.5. Confusion matrices still did not have elements centered on the leading diagonal, and tended to predict many true and false negatives (Appendix 6.3). The AUPRC was 0.258 ± 0.0157; we do not have a substantial improvement from random forests. Figure 4 shows the PRC for logistic regression trained on the imbalanced data. Again, we observe the steep decline for recall close to 0, showing that for the recall to be greater than 0, the precision tends to be less than 0.5.

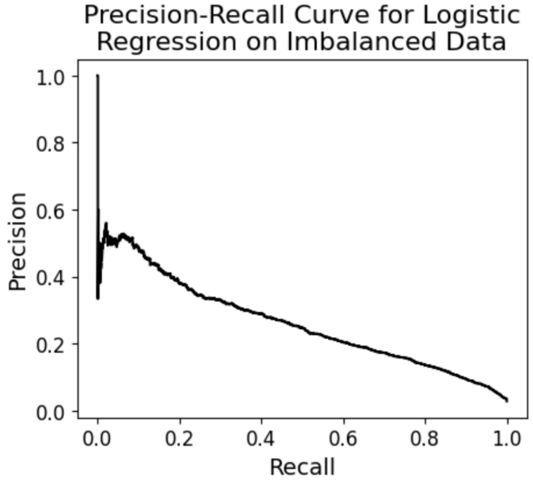

Figure 4: PRC for logistic regression trained on the imbalanced data.

Extracting the thresholds, for the recall to be 1 the threshold must be between 0 and 0.001, similar to the threshold obtained for the random forests trained on the imbalanced data. To obtain a recall of 0.7, the highest threshold is 0.01. The low thresholds imply that in both cases most samples would have to be classified as class 1, leading to many false positives and consequently low precision. Therefore, training a logistic regression model on the imbalanced data has not been successful with respect to the recall, PRC, AUPRC, and the extracted classification thresholds.

When logistic regression was trained on the balanced data, class 1 was predicted more often than when trained on the imbalanced data (Appendix 6.4). Samples had a mean recall of 0.849 ± 0.00880, yet a mean AUPRC of 0.251 ± 0.0192. The lack of change in AUPRC is due to the PRC curve shape (Figure 5), resembling what was previously observed with the imbalanced data, implying the false positives have increased along with the recall.

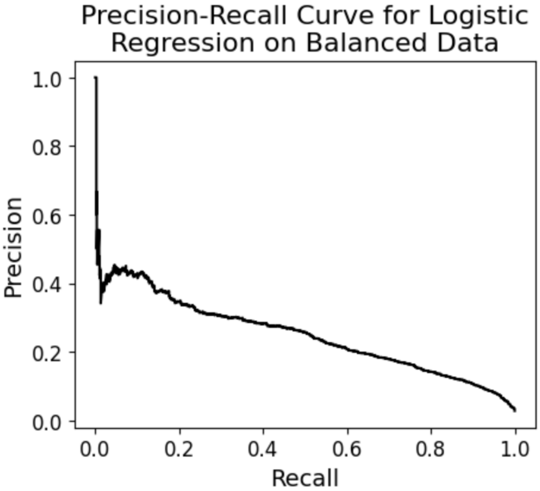

Figure 5: PRC for logistic regression trained on the balanced data.

Extracting the threshold for a recall of 1, the threshold ranges between 0 and 0.02, higher than the threshold obtained for logistic regression on the imbalanced data, but lower than the threshold for random forests trained on the balanced data. For a recall of 0.7, we obtained a threshold of 0.705, higher than any threshold we have yet seen, implying the model is able to be quite confident before classifying a sample as class 1. Using the 0.02 threshold to classify, we obtain a recall between 0.99 and 1, and an average precision of 0.0315 ± 0.000475. Using the 0.705 threshold to classify, we obtain a recall between 0.69 and 0.7, and a mean precision 0.170 ± 0.00646 - the highest precision between all models.

Logistic regression is advantageous in that we can observe the coefficients corresponding to each feature to understand the influence features have in model predictions. We analyzed the coefficients for logistic regression models trained on balanced data, as training on balanced data performed better in terms of recall than training on imbalanced data. We used 5-fold CV to obtain a mean measure of each feature's coefficients.

The coefficients corresponding to Age, Sex, General Health, Alcohol Drinkers, Smoker, Diabetes, and BMI were 0.489 ± 0.000952, 0.141 ± 0.0414, -1.11 ± 0.0113, -0.374 ± 0.0396, 0.841 ± 0.0264, 1.89 ± 0.0177, and 0.0212 ± 0.00223, respectively. As the features with positive coefficients increase, namely Age, Sex, Smoker, Diabetes, and BMI, the probability of samples being classified as class 1 increases. As patients get older, smoke more, have diabetes, and increase in BMI, they are more likely to be unhealthy. Sex was encoded as 0 (female) and 1 (male), indicating that males tend to be more likely to be classified as unhealthy. Nonetheless, the magnitude of the Sex coefficient is only 0.141, indicating that Sex does not have as strong of an impact on determining a patient's health as other features. As the features with negative coefficients increase, namely General Health and Alcohol Drinkers, the probability of samples being in class 1 decrease. As General Health increases, it makes sense that patients are less likely to be assigned to the unhealthy class. It is interesting that this relationship holds for Alcohol Consumption as well. The feature with the largest coefficient, and therefore the most relevant influence on classifying patients as unhealthy, was Diabetes, followed by Smoker. BMI was the feature with the lowest coefficient, indicating this feature has the lowest predictive power for this model. This is interesting, as in random forests we saw that Sex, Alcohol Drinkers, and Smoker were the features with the lowest feature importance.

Having identified these relationships can help doctors in medical diagnosis: seeing whether patients possess any of these attributes can help them quickly detect whether they are likely to be healthy or unhealthy. However, doctors should be cautious about using only these measures, as the logistic regression model that these coefficients were extracted from has low precision and tends to predict false positives.

3.3 Neural Networks

We tested neural networks with varying hidden layer sizes, using the `MLPClassifier` class on Scikit-Learn [12]. We tested neural networks with: one layer of 1 neuron, one layer of 8 neurons, two layers of 10 neurons each, one layer of 50 neurons, one layer of 100 neurons, and two layers, one of 100 neurons and one of 10 neurons - leading to layer sizes 1, 8, (10, 10), 50, 100, and (100, 10).

Applying neural networks to the imbalanced data performed the poorest compared to all other algorithms. Excluding the network with 100 hidden layers, all hidden layer sizes had a recall of 0, as class 1 was never predicted (Appendix 6.5). When the number of hidden layers was 100, the network had a mean recall of 0.00174 with a standard deviation of 0.00143, significantly lower than recall from past models. Nonetheless, the AUPRC did not differ dramatically, having a mean of 0.257 ± 0.0162, as the PRC curve shape remained similar due to the initial steep decline and low precision at high recall (Figure 6).

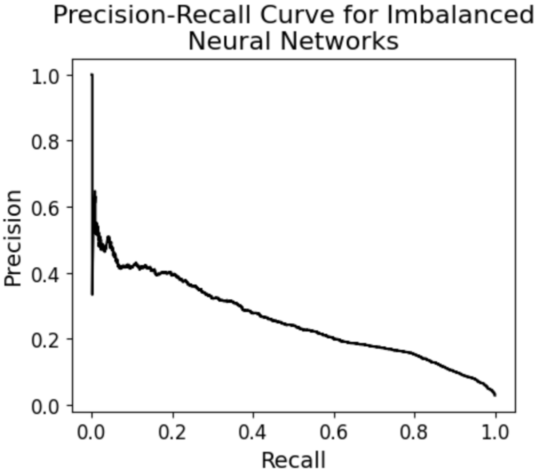

Figure 6: PRC for 100 hidden layer neural network trained on the imbalanced data.

Extracting the thresholds from the PRC, for a recall of 1 a threshold between 0 and 0.0003 was obtained, and for a recall of 0.7 a threshold between 0.0.0005 and 0.07 was obtained. Both thresholds are low, implying to achieve a good recall the model needs to classify most samples as positive, leading to an increase in the false positive rate and consequently a decrease in precision.

For the balanced data, there were also variations based on the number of hidden layers. The neural network with 1 hidden layer performed with the most variation, as in some instances it had a recall of 0 as class 1 was never predicted, but other instances it did predict class 1 (Appendix 6.6). The mean recall was 0.174 with a large standard deviation 0.347, and the mean AUPRC was 0.0833 ± 0.072. The network with hidden layer size (10, 10) was also varied, as it had a mean recall of 0.751 but a standard deviation of 0.0947, and a mean AUPRC 0.0595 ± 0.0493. There was no significant difference between any of the remaining different layer sizes, as most had a recall of 0.86. The AUPRC for these neural networks did not increase compared to the previous models, as reflected by the PRC having the familiar shape of a steep initial decline and low precision at high recall (Figure 7). The plotted PRC is of a network with 50 hidden layers, but all networks had a similar shape.

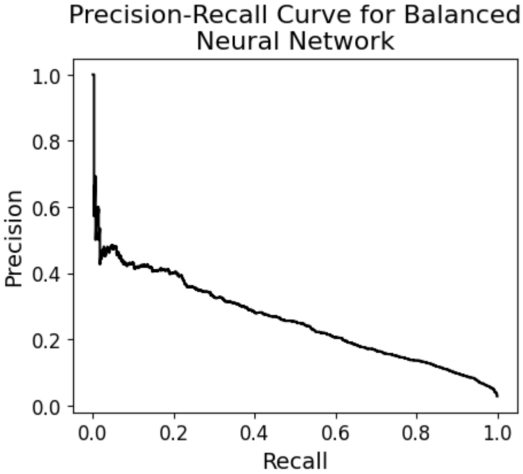

Figure 7: PRC for 50 hidden layer neural network trained on the balanced data.

We extracted thresholds based on the PRC of the neural network with 50 hidden layers. To obtain a recall of 1, a threshold of approximately 0.01 must be used, whereas to obtain a recall of 0.7, thresholds between 0.01 and 0.703 must be used. Using a threshold of 0.00857 to classify, we achieve a recall 1 with an average precision 0.0290 ± 0.000422. Using threshold 0.703 to classify, we obtain an average recall 0.729 with an average precision 0.168 ± 0.0172. As we decrease the recall, we increase the precision, and vice versa, highlighting the challenge in balancing these two metrics to achieve a “perfect” model that accurately predicts all truly unhealthy patients as unhealthy and all truly healthy patients as healthy.

3.4 Comparison

Across the selected algorithms, the models that were most successful in achieving the highest recall of class 1 on the imbalanced data were random forests with 1 minimum sample per leaf, logistic regression, and neural networks with one hidden layer of size 100. The models most successful on the balanced data were random forests with 10 minimum samples per leaf, logistic regression, and neural networks with one hidden layer of 50 neurons, although trees with 100 minimum samples and neural networks with hidden layer sizes 8, 100, and (100, 10) performed equally as well. Figure 8 compares the recall of these successful models and Figure 9 compares their AUPRC.

As can be seen in Figure 9, there is only a significant decrease with the AUPRC of the random forest trained on the imbalanced data, indicating this model was not the most successful. However, there is no significant difference between the AUPRCs of all the other models, implying each model balances precision and recall in a similar manner. Observing the recalls of the models in Figure 8, the models trained on balanced data performed significantly better than the models trained on the imbalanced data. Therefore, the most successful models were the ones trained on balanced data.

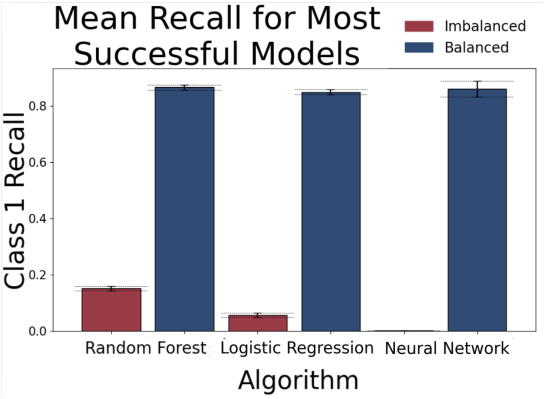

Figure 8: Mean class 1 recall for the most successful models.

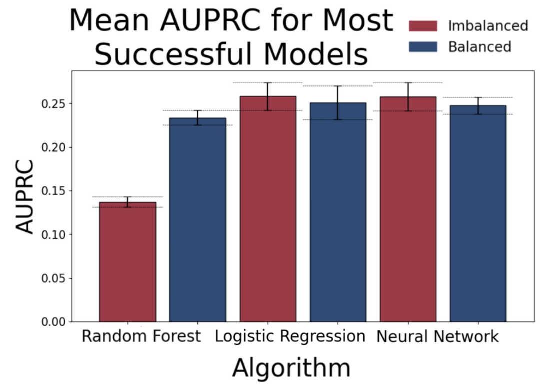

Figure 9: Mean AUPRC for the most successful models.

Referring to the previously extracted thresholds can help evaluate the three algorithms trained on balanced data. For the recall to be 1, we obtained a 0.05 threshold for random forests, 0.02 for logistic regression, and 0.01 for neural networks. For the recall to be 0.7, we obtained a 0.65 threshold for random forests, 0.705 for logistic regression, and 0.703 for neural networks.

A higher threshold indicates a model is more conservative in its classification, as it requires a higher degree of confidence to classify a sample as class 1. However, one cannot conclude that a higher threshold leads to a lower false positive rate, as the false positive rate depends on how well the model is able to distinguish between classes 0 and 1. Therefore, we observed the precision obtained when using the extracted thresholds to classify test samples.

Using the recall 1 thresholds to classify, we achieved a 0.0304 precision for the random forest, 0.0315 for logistic regression, and 0.0290 for the neural network. When using the recall 0.7 thresholds to classify, we achieved 0.163 precision for the random forest, around 0.170 for logistic regression, and 0.168 for the neural network. Figure 10 shows that logistic regression had the highest precision for a recall of 1, but Figure 11 shows there is no significant difference in the precisions between models when the recall was set to 0.7.

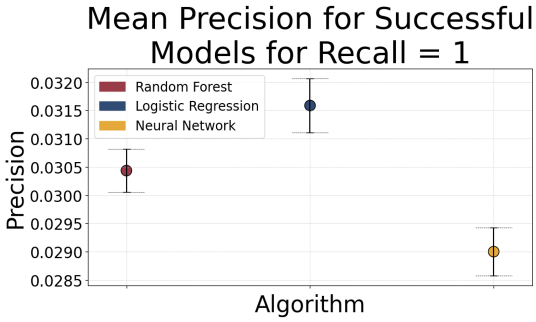

Figure 10: Mean precision for successful models using their respective thresholds, yielding a recall of 1

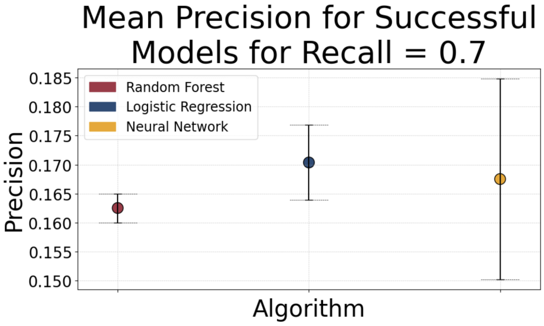

Figure 11: Mean precision for successful models using their respective thresholds, yielding a recall of 0.7

There is an inverse relationship between precision and recall, because as can be seen in Figure 12, the classifications using the thresholds yielding a recall of 1 attained a significantly lower precision than the classifications using the 0.7 recall thresholds. Figure 12 is also reminiscent of the downward shape of the PRC's for every model, highlighting how balancing the precision-recall trade off has been a prominent challenge throughout model training. When we maximize recall, we are doing so at the expense of precision. The effectiveness of the models thus depends on selecting the parameters that provide the best balance between precision and recall, essentially reducing the strength of the inverse relationship between precision and recall. Extensive feature engineering is required, which could become very time consuming. For example, neural networks use back-propagation of the error to find optimal parameters; we are not as interested in the training and test errors, but rather are interested the recall of class 1. Although it is possible to find models maximizing both recall and precision despite the discrepancy in our preferred metric and the algorithm metric, this discrepancy makes it more difficult and time consuming to find such parameters.

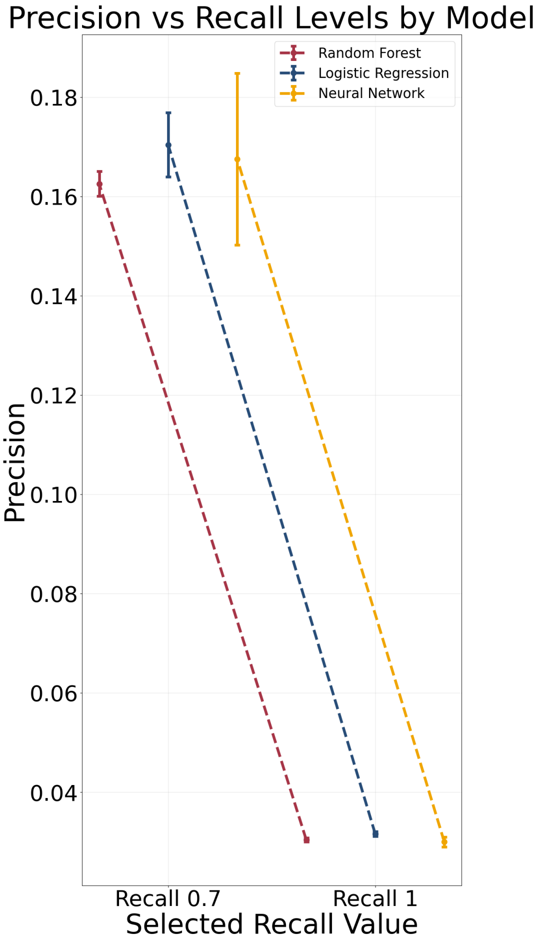

Figure 12: Precision values for extracted thresholds at given recall values, for the three most successful models.

Although logistic regression performed slightly better than the other models at recall 1, and the top three models performed equally on the data when the recall was 0.7, their inability to diminish the inverse relationship between precision and recall led them to perform poorly in the context of the investigation. If we set the recall to 1, precision decreases to less than 4%, as seen in Figure 10, indicating that many healthy patients will be wrongly classified as unhealthy. However, if we increase the precision, recall decreases due to the inverse relationship the models were unable to eliminate. This would lead to more false negatives: classifying unhealthy patients as healthy, preventing at-risk patients to get the medical attention they require.

3.5 Further Discussions

A binary classification approach is somewhat limitative in the healthcare context. Broadly classifying patients as "healthy" and "unhealthy" does not provide doctors extensive insights, as there is a discrepancy between a moderately unhealthy patient and a severely at risk patient. Thus, we briefly attempted a multi-class approach, categorizing patients on a scale of 0 (healthiest) to 8 (unhealthiest).

The multi-class approach was harder than the binary classification approach. Due to the imbalanced nature of the dataset, there were few class 8 samples (181 total samples) compared to class 0 samples (87895 total samples). Models trained on the imbalanced data struggled to predict class 8, whereas models trained on the balanced under-sampled data predicted more instances of class 8 but the values were not centered in the leading diagonal. Models predicted many false negatives and few true positives. Moreover, in the binary classification approach we wished to maximize class 1 recall, as identifying at-risk patients is more important than misclassifying healthy ones. In a multi-class approach, we focus on assigning each patient a health score, where scores 5, 6, 7, and 8 represent the unhealthiest patients. We must maximize the recall of all these four classes to ensure unhealthy patients are aware of their health risks. It becomes more challenging to find parameters that simultaneously maximize the recalls of four classes, while ensuring that precision is not drastically decreased.

Changes to the dataset could increase the success of random forests, logistic regression, and neural networks in health classification. Firstly, class imbalance was a prominent issue in the dataset. Although under-sampling was used to balance the data, gathering more samples of unhealthy patients and thus increasing the number of data points can allow a model to capture general trends in the data better, potentially lowering false positive and false negative rates, increasing recall and precision. Additionally, our dataset contained only 6 features; although introducing new features could increase the complexity of the models, adding features such as physical activity levels, dietary habits or restrictions, and genetic predispositions, which tend to be closely related to one's health, could provide more predictive power than the features currently selected, leading to more efficient classification. Moreover, the dataset does not provide access to who the patients are nor their demographic distribution. Potential lack of diversity in the dataset could affect model generalizability, reducing the model utility when applied to populations that may be underrepresented in the data. Data with more demographic-specific information could be more useful to healthcare professionals.

4 Conclusion

This investigation provided insight into the complexity of simultaneously maximizing precision and recall in healthcare models. In many disease prediction algorithms, maximizing the recall of the diseased class is important, as accurate diagnoses of sick patients prevent their conditions from worsening. Precision in such predictive models ensures healthy patients are not classified as sick and potentially cured for diseases they do not have.

In this investigation, we found random forests with 10 and 100 minimum samples per leaf, logistic regression, and neural networks with hidden layer sizes 8, 50, 100, and (100,10), trained on under-sampled balanced data, to be the most successful models. However, the inverse relationship between precision and recall hindered these models, reflected in the steep, downward shape of the models' PRC's. It was challenging to achieve AUPRC's larger than 0.3, meaning models performed slightly better than random chance.

Despite being able to create models that correctly predict all unhealthy patients as unhealthy, we were unable to create models that simultaneously predict all healthy patients as healthy. Our model could be useful to doctors because although having many false positives may lead to more expensive testing and monitoring, having a model that can correctly classify all unhealthy patients ensures no sick patients are fatally left untreated.

Logistic regression and random forests highlighted that diabetes and smoking are prominent risk factors; educational campaigns on sugar consumption and tobacco use could potentially reduce the prevalence of health-related issues.

To successfully combine the medical and ML fields, future investigations should focus on fitting models that work purely based on maximizing the precision-recall ratio, reducing the inverse relationship between precision and recall and attaining the "perfect" PRC shape. Datasets with more class balance, a larger variety of features, and more demographic information could be utilized to assess the impact of new variables on health classification as well as improving generalizability to various demographic groups. In addition, future investigations could address a multi-class case, providing valuable insights into the risk factors influencing patients' health levels rather than simply looking at the risk factors that influence whether patients are healthy or not.

Extending this investigation, models could focus on disease prediction, using a longitudinal approach to track patient health over time to predict health levels and disease development. Doctors could use these predictive models to prevent a disease before it occurs, rather than curing it once it develops as the cross-sectional models of this investigation allow.

Despite the inability to attain perfect precision and recall in this investigation, ML has the potential to develop classification and prediction models which revolutionize curative and preventative healthcare.

5 References

1. Thomas, Mike. "Ultra-Modern Medicine: Examples of Machine Learning in Healthcare." Built In, 2019, `builtin.com/artificial-intelligence/machine-learning-healthcare`. Accessed 3 Nov. 2024.

2. "2020 NHIS Questionnaires, Datasets, and Documentation." Centers for Disease Control and Prevention, `www.cdc.gov/nchs/nhis/documentation/2020-nhis.html#cdc_data_surveillance_section_2-using-our-data`. Accessed 1 Nov. 2024.

3. James, Gareth, et al. An Introduction to Statistical Learning : With Applications in Python, Springer International Publishing AG, 2023. ProQuest Ebook Central, `https://ebookcentral.proquest.com/lib/bibliotecaie-ebooks/detail.action?docID=30614337`.

4. Sanderson, Grant. But What Is a Neural Network? — Deep Learning Chapter 1. 5 Oct. 2017, `www.youtube.com/watch?v=aircAruvnKk`. Accessed 20 Nov. 2024.

5. "Datasets: Imbalanced Datasets." Google for Developers, 2024, `developers.google.com/machine-learning/crash-course/overfitting/imbalanced-datasets`. Accessed 10 Nov. 2024.

6. "3. Under-Sampling — Version 0.9.0." Imbalanced-Learn.org, `imbalanced-learn.org/stable/under_sampling.html`. Accessed 11 Nov. 2024.

7. Selvaraj, Natassha. Confusion Matrix, Precision, and Recall Explained." KDnuggests, 9 Nov. 2022, `www.kdnuggets.com/2022/11/confusion-matrix-precision-recall-explained.html`. Accessed 12 Nov. 2024.

8. Fukui, Louis. "Imbalanced Data? Stop Using ROC-AUC and Use AUPRC Instead." Towards Data Science, 2020, `towardsdatascience.com/imbalanced-data-stop-using-roc-auc-and-use-auprc-instead-46af4910a494`. Accessed 14 Nov. 2024.

9. "Sklearn.ensemble.RandomForestClassifier — Scikit-Learn 0.20.3 Documentation." Scikit-Learn.org, 2018, `scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html`. Accessed 11 Nov. 2024.

10. "Feature Importances with a Forest of Trees." Scikit-Learn, 2024, `scikit-learn.org/1.5/auto_examples/ensemble/plot_forest_importances.html`. Accessed 11 Nov. 2024.

11. Scikit-Learn. "LogisticRegression." Scikit-Learn, 2024, `scikit-learn.org/1.5/modules/generated/sklearn.linear_model.LogisticRegression.html`. Accessed 11 Nov. 2024.

12. "MLPClassifier." Scikit-Learn, `scikit-learn.org/dev/modules/generated/sklearn.neural_network.MLPClassifier.html`. Accessed 11 Nov. 2024.

6 Appendix

The following confusion matrices are examples of matrices produced in one fold of 5-fold CV. Confusion matrices are read as follows:

$$\begin{bmatrix} TN & FP \\ FN & TP \end{bmatrix}$$

where TN is true negatives, FP is false positives, FN is false negatives, and TP is true positives. Row 1 represents the number of samples that truly belong to class 0, and row 2 represents the number of samples that truly belong to class 1. Column 1 represents the number of samples that the model predicts as class 0, and column 2 represents the number of samples that the model predicts as class 1.

6.1 Confusion Matrices for Imbalanced Random Forests

$$\begin{bmatrix} 46210 & 0 \\ 1316 & 0 \end{bmatrix}$$

Matrix 1: 1000 Minimum Samples per Leaf

$$\begin{bmatrix} 46161 & 0 \\ 1365 & 0 \end{bmatrix}$$

Matrix 2: 100 Minimum Samples per Leaf

$$\begin{bmatrix} 46088 & 54 \\ 1330 & 54 \end{bmatrix}$$

Matrix 3: 10 Minimum Samples per Leaf

$$\begin{bmatrix} 45416 & 726 \\ 1168 & 216 \end{bmatrix}$$

Matrix 4: 1 Minimum Sample per Leaf

6.2 Confusion Matrices for Balanced Random Forests

$$\begin{bmatrix} 30103 & 6842 \\ 149 & 927 \end{bmatrix}$$

Matrix 5: 1000 Minimum Samples per Leaf

$$\begin{bmatrix} 29454 & 7491 \\ 129 & 947 \end{bmatrix}$$

Matrix 6: 100 Minimum Samples per Leaf

$$\begin{bmatrix} 29753 & 7192 \\ 130 & 946 \end{bmatrix}$$

Matrix 7: 10 Minimum Samples per Leaf

$$\begin{bmatrix} 29112 & 7836 \\ 196 & 877 \end{bmatrix}$$

Matrix 8: 1 Minimum Sample per Leaf

6.3 Confusion Matrix for Imbalanced Logistic Regression

$$\begin{bmatrix} 46036 & 80 \\ 1351 & 59 \end{bmatrix}$$

Matrix 9: Imbalanced Logistic Regression

6.4 Confusion Matrix for Balanced Logistic Regression

$$\begin{bmatrix} 30461 & 6498 \\ 152 & 910 \end{bmatrix}$$

Matrix 10: Balanced Logistic Regression

6.5 Confusion Matrices for Imbalanced Neural Networks

$$\begin{bmatrix} 46274 & 0 \\ 1252 & 0 \end{bmatrix}$$

Matrix 11: Hidden Layer Size 1

$$\begin{bmatrix} 46116 & 0 \\ 1410 & 0 \end{bmatrix}$$

Matrix 12: Hidden Layer Size 8

$$\begin{bmatrix} 46196 & 0 \\ 1330 & 0 \end{bmatrix}$$

Matrix 13: Hidden Layer Size (10, 10)

$$\begin{bmatrix} 46314 & 0 \\ 1212 & 0 \end{bmatrix}$$

Matrix 14: Hidden Layer Size 50

$$\begin{bmatrix} 46119 & 4 \\ 1401 & 2 \end{bmatrix}$$

Matrix 15: Hidden Layer Size 100

$$\begin{bmatrix} 46123 & 0 \\ 1403 & 0 \end{bmatrix}$$

Matrix 16: Hidden Layer Size (100, 10)

6.6 Confusion Matrices for Balanced Neural Networks

$$\begin{bmatrix} 46142 & 0 \\ 1384 & 0 \end{bmatrix}$$

Matrix 17: Hidden Layer Size 1

$$\begin{bmatrix} 37576 & 8566 \\ 230 & 1154 \end{bmatrix}$$

Matrix 18: Hidden Layer Size 8

$$\begin{bmatrix} 34552 & 11692 \\ 153 & 1129 \end{bmatrix}$$

Matrix 19: Hidden Layer Size (10, 10)

$$\begin{bmatrix} 37593 & 8549 \\ 231 & 1153 \end{bmatrix}$$

Matrix 20: Hidden Layer Size 50

$$\begin{bmatrix} 37698 & 8444 \\ 227 & 1157 \end{bmatrix}$$

Matrix 21: Hidden Layer Size 100

$$\begin{bmatrix} 35355 & 10787 \\ 180 & 1204 \end{bmatrix}$$

Matrix 22: Hidden Layer Size (100, 10)