Utilizing Machine Learning to Classify Chronic Disease: Quantifying Biopsychosocial Risk and Resilience

Meghan Colosimo, John Obuch, Fatih Catpinar, & Colin Murphy

Drexel University: CS-613 Machine Learning

Abstract

Traditional classification models are used to separate data into groups, with the most simplistic example being a binary-classification model where only two target classes exist. Similarly, we often encounter multi-classification scenarios where $n > 2$ target classes exist. These models are often designed to provide a definitive "Hard" label, where only one label is assigned. However, for many classification applications, the assignment of a single label is unhelpful or misleading. Thus, a multi-label "Soft" classification method is necessary to represent the array of possible target features. The importance of multi-label classification (MLC) can be most easily exemplified by medical screening examples. For classification problems such as these, it is important to identify the various risk factors rather than identifying the most likely. In this paper, we employ both binary and multi-label approaches to classifying chronic disease diagnoses based on the Behavioral Risk Factor and Surveillance System (BRFSS) survey data, provided by the Centers for Disease and Control and Prevention (CDC). Specifically, we apply Logistic Regression, Gaussian Naive Bayes, Decision Trees, Random Forest and Artificial Neural Networks to the task of classifying healthy vs. chronically ill individuals, with a focus on diabetes, heart disease/heart attack, and hypertension. We compare and contrast the performance of these Machine Learning (ML) models, with and without applying Principle Component Analysis (PCA) for dimensionality reduction prior to modeling. Findings suggest that Logistic Regression is superior in regards to precision, accuracy, and F1 scores for classifying diabetes, heart disease/attack, and hypertension. However, where recall is concerned, Random Forest emerged as the superior model for heat disease/attack, while Naive Bayes had the highest recall for diabetes, and Random Forest and Naive Bayes were virtually equivalent in best recall for hypertension.

Background

Chronic illness is persistent (1 year or more symptom duration), functionally impairing, largely incurable, requires ongoing time and cost-consuming management and is alarmingly prevalent across the United States. The Centers for Disease Control and Prevention (CDC)'s National Center for Chronic Disease Prevention and Health Promotion (NCCDPHP) estimates "6 in 10" (133 million) Americans live with at least one chronic illness, while "4 in 10" (100 million) live with two or more comorbid chronic diagnoses (4) (5). Despite the fact that over half of the adult U.S. population is considered chronically ill (60% as of 2014), prevalence continues to increase with conservative projections for the year 2025 predicting 164 million afflicted nationally (6).

Among the most common and costly chronic illnesses are hypertension, lipid disorders including high cholesterol, diabetes mellitus (type II), and heart disease. According to analysis of the 2014 Medical Expenditure Panel Survey (MEPS) conducted by Buttorff, Ruder and Bauman (2017), 27.0% of U.S. adults are diagnosed with hypertension, 21.6% have a lipid disorder/high cholesterol, 10.4% are living with diabetes (5% of which have type 1 diabetes), and 4.8% are diagnosed with some form of heart disease (including coronary atherosclerosis) (7). Heart disease, the most common being coronary heart disease (CHD), and by association heart attack are the primary cause of death in the U.S. across genders and most ethnicities, amounting to 610,000 heart disease deaths/year (1 in 4 deaths) and 790,000 heart attacks/year (8). Together, the cost of heart disease and heart attack total $200 billion in direct (treatment, medication) and indirect (loss of productivity, disability) costs per year (9). Hypertension is deemed the primary contributing cause of death among Americans, given high blood pressure is a major risk factor for the development of heart disease and stroke, causing or contributing to 410,000 deaths/year (9). Annual combined direct and indirect costs for hypertension in the U.S. are $48.6 billion (9). Diabetes is the seventh leading cause of death in America and was responsible for 252,806 deaths in 2015; in 2017, the total cost of diabetes amounted to $327 billion (10).

Given the complex, interconnected nature of these chronic health conditions, the present study will consider many of these diagnoses simultaneously, with a focus on hypertension, diabetes, and heart disease/heart attack given their relatively high prevalence, mortality rates and associated healthcare expenditures. Across age groups, 42% of U.S. adults were living with chronic illness multimorbidity in 2014; however, middle-aged (45 – 64 years) and older adults (65+ years) demonstrate increased prevalence of multimorbidity when compared to their younger adult peers, such that 50% of middle-aged Americans and 81% of older Americans reported multiple chronic illness diagnoses (7). Despite the growing epidemic of chronic disease in the U.S., it is crucial to recognize that the majority of those chronic health conditions which prohibit healthy living and well-being and shorten the average American lifespan are largely preventable. The necessary ingredients for prevention include education, screening, healthcare access and public empowerment. Accurately predicting chronic illness development requires the integration of multiple individual and environmental domains, including demographics, current objective/subjective physical and psychological health status, health-related behaviors, healthcare access and social determinants of health.

Related Work

In recent years, there has been a growing, multidisciplinary movement to leverage the powerful analytic tools of ML in order to build complex, multivariate models of chronic illness risk and resiliency. A variety of classification-based ML models and techniques, including but not limited to Logistic Regression, K-Nearest Neighbors, Support Vector Machines, Decision Trees/Random Forrest, Naïve Bayes, Gradient Boosting, and Neural Networks, have been applied to predict chronic illness diagnoses based on both self-report and electronic health record (EHR) data sources. Patil and Tamane (2018) conducted a comparative analysis of 8 ML methods to predict Diabetes diagnoses using features available in the Pima Indians Diabetes Database, including Logistic Regression, K-Nearest Neighbors, Naïve Bayes, and Multilayer Perceptron (MLP). The reported accuracy of all models evaluated by Patil and Tamane ranged from 0.641 (MLP) to 0.79 (Logistic Regression, Gradient Boosting) (11). Gradient Boosting and Logistic Regression models emerged as superior in their performance as diabetes classifiers, given that these models were consistently associated with the highest precision (0.774; 0.762), recall (0.747; 0.716), F1 scores (0.754; 0.730), and Area Under the Receiver Operating Curve (ROC AUC) (0.75) (11). Conversely, the Linear SVM and MLP neural network consistently demonstrated the poorest performance, as indicated by relatively low accuracy (0.689; 0.640), precision (0.340; 0.612), recall (0.50; 0.635), F1 scores (0.405; 0.613) and ROC AUC (0.50; 0.62) (11).

Lopez-Martinez et al. (2018), Ye et al. (2018), and LaFreniere et al. (2016) each applied ML models to the classification of hypertension diagnoses. Lopez-Martinez and colleagues applied a binary Logistic Regression to the National Health and Nutrition Examination Survey (NHANES) 2007-2016 datasets, based on seven pre-selected features – gender, age, race, BMI, kidney disease status, tobacco use, and hypertension status. Lopez-Martinez et al. (2018) reported that their final Logistic Regression model performed with 84% precision, 70% recall, and F1 score of 0.74, and a ROC AUC of 73% (12). Ye and colleagues (2018) applied XGBoost, a gradient boosting ML algorithm, to EHRs from the Maine Health Exchange Network in order to predict 1-year risk for incident essential hypertension. Ye and collages reported superior model performance when compared to the existing literature reviewed here, boasting ROC AUCs of 0.917 and 0.870 across cohorts (13). LaFreniere et al. (2016) applied a 3-layer 11-7-2 Artificial Neural Network (ANN) to the Canadian Primary Care Sentinel Surveillance Network (CPCSSN) dataset with the aim of predicting hypertension diagnosis, based on eleven features – birth year, gender, BMI, systolic BP, diastolic BP, HDL, LDL, Triglycerides, Cholesterol, Micro-Albumin, and Urine Albumin-Creatine Ratio. LaFreniere et al. (2016) reported that their ANN achieves an overall accuracy of 82.3% (14). Taken together, the existing literature supports the application of ML algorithms to public health concerns of detection and prevention of chronic illness.

Existing research utilizing the CDC's Behavioral Risk Factor Surveillance System datasets overwhelmingly focuses on a specialized subset of the total sample to answer a narrowly defined research question, many with limited implications and/or no direct functional applications. As such, the proposed project leverages the powerful analytic tools of ML to build comprehensive, multivariate models of chronic illness risk, with results that many one day be immediately accessible and applicable to the average American in the form of a web-based application.

Methodology

Data Source

Since 1984, the CDC has successfully conducted the largest cross-sectional survey of demographics, chronic illness diagnoses, health-related behaviors and perceived psychophysical wellbeing worldwide – the Behavioral Risk Factor Surveillance System (BRFSS). This standardized survey is administered annually via telephone by interviewers in demography research centers across each of the 50 United States and the U.S. territories of Guam, Puerto Rico and the District of Columbia. The BRFSS datasets and associated documentation are publicly available and published annually on the CDC's website at `https://www.cdc.gov/brfss/annual_data/annual_data.htm`. The present study utilizes the 2017 BRFSS dataset. The complete responses of 377, 658 and partial responses of 72,358 "noninstitutionalized" adults 18 years of age or older and residing in the U.S. at the time of data collection are included in the 2017 BRFSS database (N = 450,016). Participants were randomly selected for participation by their telephone numbers, yielding a nationally representative sample of adults by design.

Pre-processing Pipeline

A comprehensive pipeline was developed for the purposes of properly pre-processing the 2017 BRFSS dataset for subsequent modeling. The following steps were implemented as part of the pre-processing phase:

1. Remove features deemed irrelevant to the project aims, including a) survey metadata, b) open-ended response data, c) inconsistently scaled continuous features (e.g., response in months, days, and/or years), d) redundant features (e.g., height in meters vs. height in inches), and e) redundant outcomes (e.g., multiple versions of the question 'Have you ever been diagnosed with X disease?').

2. Rescale variables for consistency (e.g., 0 = no diagnosis, 1 = has diagnosis, for outcomes variables).

3. Address missingness by a) collapsing multiple missing data categories into one (e.g., 'Don't Know' = 'Refused' = 'Missing'), b) dropping features missing $>$ 60% of the total sample, and c) applying sklearn SimpleImputer for mode imputation of features missing $<$ 60%.

4. Dummy coding categorical features and standardizing continuous features.

5. Applying Synthetic Minority Over-sampling Technique (SMOTe) to address imbalanced representation of the minority class (cases with the diagnosis) in the data. This allows us to generate synthetic data for the minority class (3).

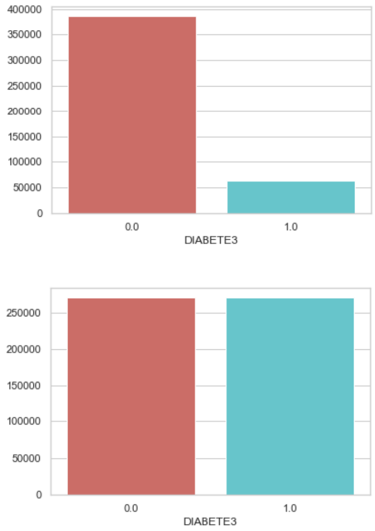

Figure 1: Group Frequencies before and after SMOTe for Diabetes Outcome

For binary classification, the final, pre-processed datasets consisted of 522 features for diabetes and heart disease/attack, and 521 features for hypertension. After pre-processing the data as demonstrated in our source code, we approach the classification of disease likelihood outcome(s) by implementing various models and examining the performance of each to determine our ideal model type that most accurately classifies the an individuals likelihood/risk of contracting the illness of interest. The ML models evaluated include Logistic Regression, Gaussian Naive Bayes, Decision Trees, and Neural Networks. A 70/30 train-test split was employed. We also explored the approach of applying PCA to reduce the dimensionality of the data and feed the models accordingly to compare performance results. The following subsections outline the modeling approaches taken and provide a high level overview of the underlying mathematics performed by Python's sklearn machine learning algorithms.

Logistic Regression

With logistic regression, we assume a binary classification with the goal of providing a probability of the positive class such that $0 \leq P(y = 1) \leq 1$. Unlike linear regression where we computed $g(x, \theta) = x\theta$, we focus our attention on classifying the outcome value with respect to where it falls on the sigmoid. We do so by computing the following probability:

$$P(y = 1 \mid x, \theta) = g(x, \theta) = \frac{1}{1 + e^{-x\theta}}$$

Similarly, we can compute the probability of the an observation belonging to the negative class by computing $1 - g(x, \theta)$. It is important to observe that the probability function defined above is differentiable at every point. This is useful because with logistic regression, we seek to find the parameters θ that minimize the classification error or maximize the correct class likelihood. To do so, we need to take a derivative. To explain further, given a supervised observation (x, y), we wish to compute the likelihood of a correct classification using the following equation:

$$l(y \mid x, \theta) = (g(x, \theta))^y (1 - g(x, \theta))^{(1-y)}$$

Applying this approach to the entire data set and assuming conditional independence across the observations, we can write the above equation in the following form:

$$l(Y \mid X, \theta) = \prod_{t=1}^{N} (g(X_t, \theta))^{Y_t} (1 - g(X_t, \theta))^{(1-Y_t)}$$

From here, we desire to take the derivative. However, we will first take the *log* of the above formulation. Once we do this, we will maximize the log-likelihood by taking the derivative as alluded to above. The following derivation shows how we arrive to the formulation of computing the parameters θ:

$$l(Y|X, \theta) = \sum_{t=1}^{N} Y_t \ln(g(X_t, \theta)) + (1 - Y_t) \ln(1 - g(X_t, \theta))$$

Proceeding with maximizing the log-likelihood we have:

$$\frac{\partial}{\partial \theta_j} l(y|x, \theta) = \frac{\partial}{\partial \theta_j} (y \ln(g(x, \theta)) + (1 - y) \ln(1 - g(x, \theta)))$$

$$= \frac{y}{g(x, \theta)} \frac{\partial}{\partial \theta_j} (g(x, \theta)) + \frac{(1 - y)}{1 - g(x, \theta)} \frac{\partial}{\partial \theta_j} (1 - g(x, \theta))$$

$$= \frac{y}{g(x, \theta)} \frac{\partial}{\partial \theta_j} (\frac{1}{1 + e^{-x\theta}}) + \frac{(1 - y)}{1 - g(x, \theta)} \frac{\partial}{\partial \theta_j} (1 - \frac{1}{1 + e^{-x\theta}})$$

From here, let us simplify our computation by computing $\frac{\partial}{\partial \theta_j} (\frac{1}{1+e^{-x\theta}})$

$$\frac{\partial}{\partial \theta_j} (\frac{1}{1 + e^{-x\theta}}) = x_j g(x, \theta)(1 - g(x, \theta))$$

Substituting this result into our flow above and simplifying we yield the following:

$$\frac{\partial}{\partial \theta_j} l(y \mid x, \theta) = x_j (y - g(x, \theta))$$

Vectorizing this result for all the parameters we have the following form:

$$\frac{\partial l}{\partial \theta} = x^T (y - g(x, \theta))$$

Next, we wish to vectorize this result to be the mean gradient across all observations which gives us:

$$\frac{\partial l}{\partial \theta} = \frac{1}{N} x^T (y - g(x, \theta))$$

From here, we apply gradient decent such that our parameters θ converge to their true values. We do this by updating θ as follows:

$$\theta = \theta - \frac{\eta}{N} X^T (g(x, \theta) - Y)$$

Where η is the learning rate. After convergence is reached, we will have obtained our final model utilizing logistic regression.

Gaussian Naïve Bayes

A Naïve Bayes Classifier is a machine learning algorithm based on Bayes theorem. The Bayes theorem can be represented as following:

$$P(A \mid B) = \frac{P(B \mid A) P(A)}{P(A)}$$

A and B are events. $P(A)$ and $P(B)$ are the probabilities of events and are independent from each other. $P(A \mid B)$ is the probability of A occurring given the condition B. And $P(B \mid A)$ is the probability of B happening given the condition A. As touched on above, in Naïve Bayes, the features are assumed to be independent of each other. By using the basis of Bayes theorem, the Naïve Bayes Classifier formula can be written as follows:

$$P(y = i \mid f = x) = \prod_{k=1}^{D} P(f_k = x_k \mid y = i)$$

where x is a set of D features and y is the class. Since a large amount of our data is represented as continuous values and we can't calculate the probability of the value, we applied Gaussian Naïve Bayes algorithm that uses the Gaussian Distribution Function to calculate the probabilities. The Gaussian Distribution Function is shown below.

$$f(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

where μ is the mean and σ is the standard deviation. We applied a binary classification. We calculated the probabilities for both classes, and the class that had the highest probability was declared the predicted result.

Decision Trees

The underlying idea behind decision tree classification is that of measuring the impurity of a node. "A node having multiple classes is impure whereas a node having only one class is pure" (2). Given data with a set of features, we desire to identify the feature(s) that can split the data into subsets that ideally contain observations from a single class. To achieve this, we leverage an approach of feature selection, namely *information gain*. We should note here that there are two approaches for computing the *information gain*. A commonly used approach is to compute the *entropy* of the system which measures the randomness of the data. Another approach (Python's *sklearn* default) is to compute the *Gini Score/Index*. The following equations illustrate the difference of *entropy* versus the *Gini Score*:

Entropy

$$H(P(v_1), \ldots, P(v_n)) = \sum_{i=1}^{n} (-P(v_i) \log_n P(v_i))$$

Gini Score/Index

$$H(P^2(v_1), \ldots, P^2(v_n)) = 1 - \sum_{i=1}^{n} P^2(v_i)$$

where $P(v_i)$ is the probability of event v occurring in a given node corresponding to feature i.

We choose to leverage the *entropy* approach for the purposes of providing the reader with a high-level overview of the underlying math associated to decision tree modeling. In reference to the *entropy* equation outlined above, we determine the attribute to split on by computing the *information gain* associated to a given feature. Let us assume a discrete binary classification. We identify the feature(s) of interest by computing the *information gain*. As mentioned above, we assume a binary classification where we notate p as the number of samples with a classification label one and n as the number of samples with classifications labeled zero. We utilize these figures to compute the prior probability. Lastly, we will let p_i and n_i represent the number of samples associated to each subset E_i with labels one and zero respectively. Note that E_i is the subset associated to each observations value $X_{i,j}$. The idea here is to minimize the *entropy* after the split on the feature(s). Under these assumptions, we can compute the average *entropy* as follows:

$$\mathbb{E}(H(A)) = \sum_{i=1}^{k} \frac{p_i + n_i}{p + n} H(\frac{p_i}{p_i + n_i}, \frac{n_i}{p_i + n_i})$$

We now have everything we need to compute the *information gain* associated to a given feature. The following equation illustrates how we achieve our conclusions of which feature/attribute to split on:

$$\text{IG}(A) = H(\frac{p}{p+n}, \frac{n}{p+n}) - \mathbb{E}(H(A))$$

Based on the resulting outputs from the equation above, we will select the feature with the largest *information gain*. One thing to keep in mind when implementing a decision tree model is the concept of over-fitting. Since decision tree algorithms are an iterative process, it is common to use a validation set and report the accuracy score associated to a given tree. Given this aspect of the algorithm, we chose to explore the implementation of a *random forest* approach to our data for comparison (*see results section for comparisons*).

Neural Network Architecture

The intricacies of MLC problems may be better captured by models with higher orders of complexity, thus we have constructed feed-forward artificial neural networks (NN) for simultaneous classification of various types of chronic diseases. The width and density of this network was kept to a minimum (relative to many other neural network architectures) to easily investigate the training methods of multi-label classification — a 183-100-10 neural network was constructed.

When considering the activation of our NN, we cannot represent our class label outputs as a probability distribution amongst all labels as in a typical logistic regression model (assigning the label with the highest probability). Rather, the output nodes of possible class labels must be represented by a sigmoid activation function.

Additional attention was also given to the method of learning for our NN. The multi-label nature of our model requires that the model learn and propagate error for each label output independently; therefore, our loss function is simply binary cross entropy (or binary log loss). The use of this loss function will ensure that each label output is modeled as independent Bernoulli Distributions.

Experiments and Results

Binary Classification Results

The tables below display results associated with the various models implemented, namely Logistic Regression (LR), Gaussian Naïve Bayes (GNB), Decision Tree (DT), and Random Forest (RF). Each model was run with and without applying PCA prior to modeling. PCA selected a total of 95 features for diabetes and heart disease/attack, and 99 features for hypertension. For every outcome/model combination examined, results with all features retained were superior to the results obtained with dimensionality reduction applied. Performance metrics highlighted in bold are associated with the highest achieved performance across models for each outcome - diabetes, heart disease/attack, and hypertension. Overall, Logistic Regression emerged as the model with superior performance for binary classification of diabetes, heart disease/attack, and hypertension, as measured by accuracy, precision, and F1 scores. Given that high recall (minimizing risk of false negatives) is prioritized in medical diagnosis classification, we note that Naive Bayes was associated with the highest recall for diabetes, while Random Forest was associated with the highest recall for heart disease/attack. Both Naive Bayes and Random Forest provided comparable, highest recall rates for hypertension.

Diabetes Model Results				
Metrics	LR	GNB	DT	RF
Accuracy	**0.8701**	0.6776	0.7984	0.7025
Precision	**0.5476**	0.2375	0.3511	0.2498
Recall	0.4628	**0.5802**	0.5038	0.5524
F1-Score	**0.5016**	0.3370	0.4138	0.3441
True Pos	8824	11063	9606	10533
False Pos	7290	35523	17750	31627
True Neg	108647	80414	98187	84310
False Neg	10244	8005	9462	8535

Diabetes PCA Model Results				
Metrics	LR	GNB	DT	RF
Accuracy	0.8465	0.6451	0.7936	0.7110
Precision	0.4440	0.1734	0.2092	0.2230
Recall	0.3438	0.4017	0.1660	0.4212
F1-Score	0.3875	0.2423	0.1851	0.2916
True Pos	6555	7659	3165	8032
False Pos	8208	36500	11966	27987
True Neg	107729	79437	103971	87950
False Neg	12513	11409	15903	11036

Heart Disease/Attack Model Results				
Metrics	LR	GNB	DT	RF
Accuracy	**0.9046**	0.7150	0.8738	0.7441
Precision	**0.4314**	0.1469	0.3018	0.1702
Recall	0.2883	0.4704	0.3381	**0.4876**
F1-Score	**0.3456**	0.2239	0.3189	0.2536
True Pos	3401	5549	3988	5870
False Pos	4482	32223	9227	28625
True Neg	118727	90986	113982	94584
False Neg	8395	6247	7808	5926

Heart Disease/Attack PCA Model Results				
Metrics	LR	GNB	DT	RF
Accuracy	0.8907	0.7595	0.8211	0.7284
Precision	0.3391	0.1293	0.1756	0.0962
Recall	0.2648	0.3056	0.2835	0.2512
F1-Score	0.2974	0.1817	0.2168	0.1392
True Pos	3123	3605	3344	2963
False Pos	6086	24284	15704	27828
True Neg	117123	98925	107505	95381
False Neg	8673	8191	8452	8833

Hypertension Model Results				
Metrics	LR	GNB	DT	RF
Accuracy	**0.7491**	0.6119	0.7062	0.5522
Precision	**0.6883**	0.5108	0.6201	0.4678
Recall	0.6872	**0.8268**	0.6949	**0.8269**
F1-Score	**0.6878**	0.6314	0.6554	0.5976
True Pos	37302	44882	37720	44886
False Pos	16889	42991	23105	51059
True Neg	63833	37731	57617	29663
False Neg	16981	9401	16563	9397

Hypertension PCA Model Results				
Metrics	LR	GNB	DT	RF
Accuracy	0.7329	0.6701	0.6876	0.6253
Precision	0.6668	0.5737	0.6029	0.5310
Recall	0.6713	0.6987	0.6532	0.5821
F1-Score	0.6690	0.6301	0.6271	0.5554
True Pos	36438	37928	35458	31599
False Pos	18212	28180	23351	27907
True Neg	62510	52542	57371	52815
False Neg	17845	16355	18825	22684

Multi-Label Neural Network Results

For MLC problems, it becomes difficult to evaluate a model using traditional binary or multi-class metrics

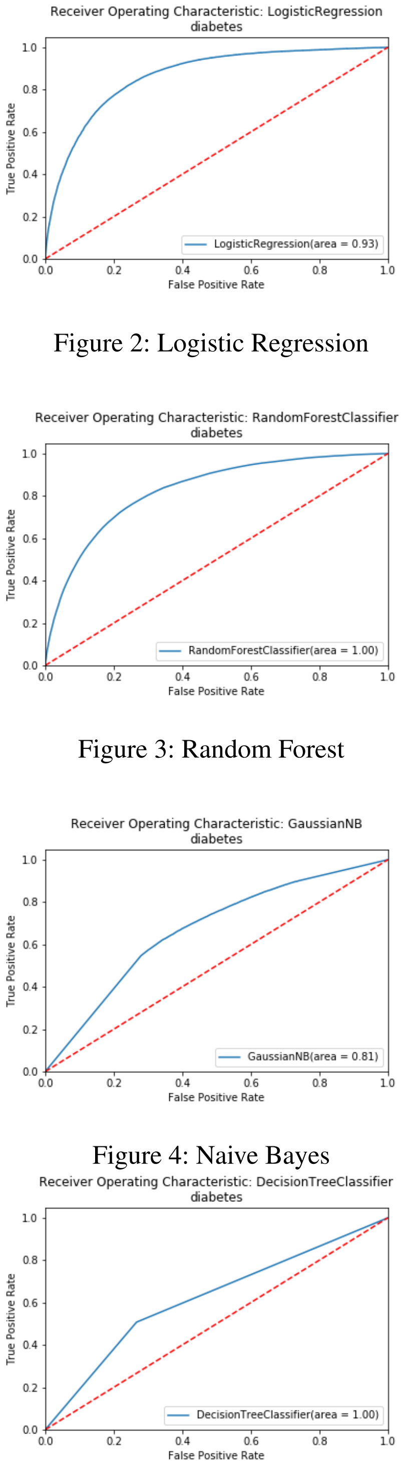

Figure 2: Logistic Regression

Figure 3: Random Forest

Figure 4: Naive Bayes

Figure 5: Decision Tree

Figure 6: ROCs for Diabetes Classification Models

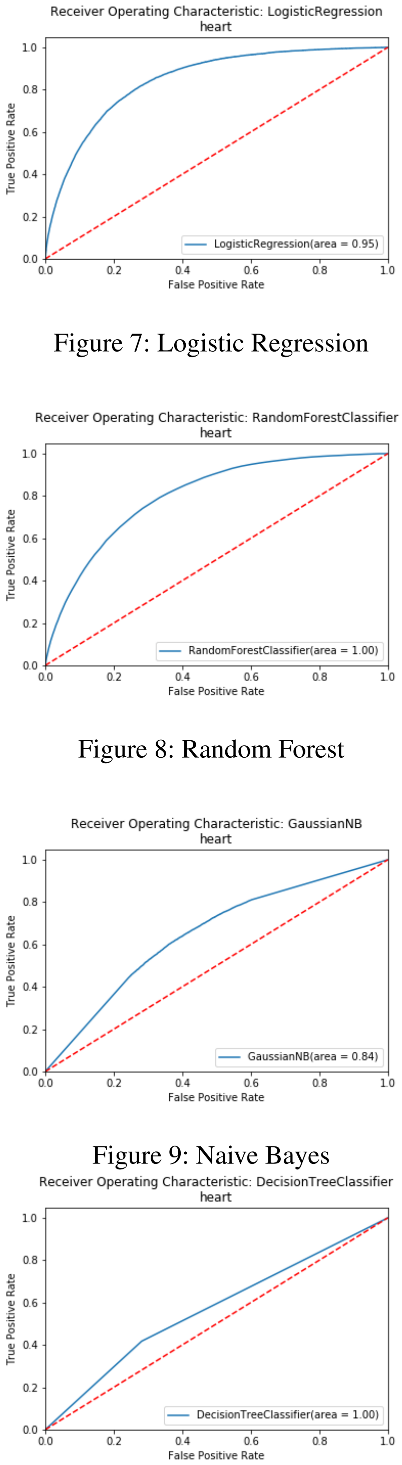

Figure 7: Logistic Regression

Figure 8: Random Forest

Figure 9: Naive Bayes

Figure 10: Decision Tree

Figure 11: ROCs for Heart Disease/Attack Classification Models

such as a contingency table or a confusion matrix. The context of the modeling goal becomes important as we may want to understand the labels' dependencies to one another — conditional and marginal (unconditional) dependency (1). The focus of this study will be on the marginal dependencies of the labels.

To measure the performance of our MLC model we have considered the error cost of a false negative (missing a diagnosis) and false positive (assigning an incorrect diagnosis) to be high. Thus, our MLC model metrics have focused on the model's sensitivity ("recall" or "true positive rate") and specificity ("selectivity" or "true negative rate"). Additionally, the hamming-loss of the model is provided to understand the overall fraction of incorrect labels amongst all labels. Thus, we would seek a model with high recall. The issue with solving by decomposition into several binary classifiers (binary relevance training) is that this approach may ignore important interdependencies of the labels. In the table below, the sensitivity and specificity metrics of the model for each class label are given — with the average across all labels being 0.5036 and 0.7511 for sensitivity and specificity, respectively. Furthermore, the overall hamming loss was calculated to be 0.2817.

Neural Network MLC Model Results		
Class Label	Sensitivity	Specificity
Heart Attack	0.4352	0.8894
Angina	0.5110	0.8926
Stroke	0.5287	0.7342
Asthma	0.1793	0.8039
Skin Cancer	0.5967	0.7038
General Cancer	0.6002	0.6513
COPD	0.5387	0.7450
Arthritis	0.6017	0.7587
Depression	0.4522	0.6335
Kidney Disease	0.5920	0.6986

Note: Threshold for each label was set as the mean of probability rather than default 0.5

The exception of low sensitivity for the Asthma label suggests that critical attributes for identifying the positive instances are missing from our dataset. Further analysis of the original dataset (prior to PCA) may help

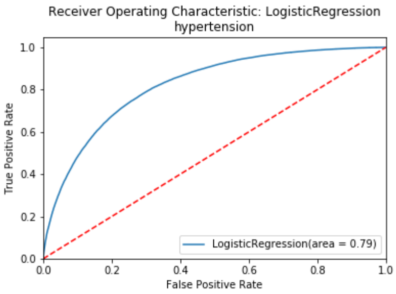

Figure 12: Logistic Regression

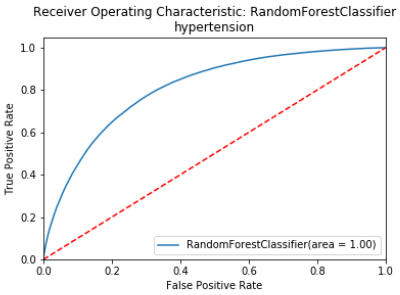

Figure 13: Random Forest

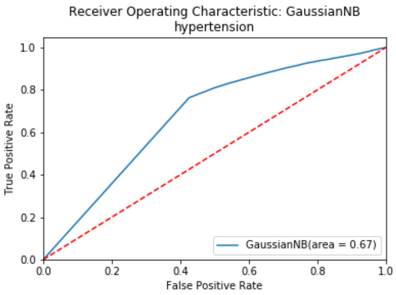

Figure 14: Naive Bayes

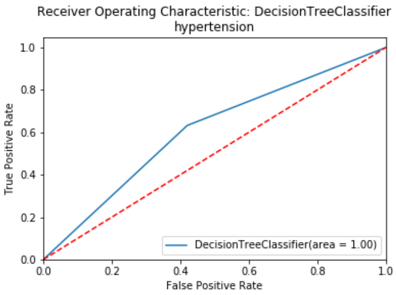

Figure 15: Decision Tree

Figure 16: ROCs for Hypertension Classification Models

to identify the critical information for this particular label.

Conclusions

Analyzing the results above, we observe that the performance of our disease classifier is heavily dependent on the the algorithmic model we choose. Based on accuracy, precision, and F1 scores, the model with the highest performance across disease outcomes was the Logistic Regression binary classifier. Based on the same performance metrics, the Decision Tree classifier emerged as the second best binary classifier across all diseases outcomes. We observed that performance based on recall alone varied across outcomes, such that Naive Bayes was associated with the highest recall for diabetes, Random Forest provided the highest recall for heart disease/attack, and Naive Bayes and Random Forest were comparable in their performance on recall for Hypertension. Models that incorporated PCA performed poorly compared to models utilizing the full set of available features. We posit that the extensive preprocessing which occurred prior to applying PCA in order to rid that dataset of redundant features in combination with the loss of information that occurs as a function of PCA is responsible for these results. Additionally, PCA is best suited for continuous as compared to categorical data. Though dummy coding of categorical features was performed as part of pre-processing, it should be noted that the majority of the features in this dataset were categorical in nature.

Transitioning our attention to the Receiving Operating Characteristics (ROC) and Area Under the Curve (AUC) diagrams, we assumed a threshold of 50%. Under this assumption we glean important information about the performance of our models in addition to the generic performance metrics such as precision and recall. To explain further, let us examine the results of our Diabetes classifiers. We see that although the decision tree classifier has an AUC of one, the ROC curve illustrates a fairly poor performance in regard to separating the classes as it hugs the “red” reference line. Furthermore, even though the accuracy of the decision tree model reports to be higher than that of the random forest model as outlined in the tables above, we see from the ROC/AUC curves that the random forest model is superior when it comes to separating the classes with a true positive rate of approximately 80% and a false positive rate of around 20% as opposed to that of the decision tree model with true positive and false positive rates indicating 40% and 10% respectively. This further alludes to the fact that although the performance metrics portray good performance with respect to the decision tree model, we see that it does a poor job of separating the classes as mentioned above. Using the classification models we implemented to identify chronic disease based on our extensive amount of features within our data, we see that the logistic regression model and random forest model show promising results while minimizing classification error rates. Overall, we were able to improve upon our proposed baseline evaluation results and provide a novel analysis of chronic disease classification. *For additional model evaluation results such as confusion matrices and PR curves, please refer to our source code*.

Future Work/Extensions

We encourage future researchers to expand on our results by refining our implementation of multi-classification. Furthermore, we recognize that there could exist some biases embedded in our results due to imbalanced data. As mentioned in the sections above, we did apply SMOTe to the data to generate synthetic data to control for this, however, we urge future expansion of this to optimize the normality of the data prior to running the machine learning models we implemented above. Relevant, health-related datasources other than BRFSS could also be aggregated to increase the breadth of relevant features and improve performance. Other areas of interest include optimizing the neural network and exploring the application of penalty terms to the models in an attempt to better classify the data. Additionally, feature importance analysis could be applied to the modeling results to determine which features are most influential, signaling the most impactful targets of intervention to improve health. Future directions of this line of research may also include development of a web-based application for disseminating information to the general public about predicted risk for chronic illness to motivate positive, actionable health behavior change.

References

[1] Dembczyǹski, K.; Waege-man, W.; Cheng, W.; and Hüllermeier, 2012. On label dependence and loss minimization in multi-label classification. *Machine Learning* 88(1-2):5–45.

[2] *Gini Index vs Entropy Information gain: Decision Tree: THAT-A-SCIENCE.* (n.d.). Retrieved March 15, 2020, from *https://thatascience.com/learn-machine-learning/gini-entropy/*

[3] Mahendru, K. (2019, June 26). *How to Deal with Imbalanced Data using SMOTE.* Retrieved March 15, 2020, from *https://medium.com/analytics-vidhya/balance-your-data-using-smote-98e4d79fcddb*

[4] National Center for Chronic Disease Prevention and Health Promotion (NCCDPHP). (March, 2019). About chronic diseases. Retrieved from *https://www.cdc.gov/chronicdisease/about/index.htm*.

[5] National Center for Health Statistics. (2014). Health, United States, 2013: With Special Feature on Prescription Drugs. Hyattsville, MD.

[6] Partnership for Solutions. (2004). Chronic Conditions: Making the Case for Ongoing Care. September 2004 Update. Retrieved from *https://www.fightchronicdisease.org*.

[7] Buttorff, C., Ruder, T., Bauman, M. (2017). Multiple chronic conditions in the United States. Retrieved from *https://www.rand.org/content/dam/rand/pubs/tools/TL200/TL221/RAND*$_{T}L221.figurecharts.pdf$.

[8] Benjamin, E.J., Blaha, M.J., Chiuve, S.E., Cushman, M., Das, S.R., Deo, R., et al. (2017). Heart disease and stroke statistics – 2017. American Heart Association Statistical Update, 135, e146-e603.

[9] National Center for Health Statistics. (2016). Health, United States, 2015: With Special Feature on Racial and Ethnic Health Disparities. Hyattsville, MD.

[10] Centers for Disease Control and Prevention. (2017) National Diabetes Statistics Report, 2017: Estimates of Diabetes and its Burden on the United States. Retrieved from *https://www.cdc.gov/diabetes/pdfs/data/statistics/national-diabetes-statistics-report.pdf*.

[11] Patil, R. Tamane, S. (2018). A comparative analysis of the evaluation of classification algorithms in the prediction of diabetes. International Journal of Electrical and Computer Engineering, 8(5), 3966 - 3975.

[12] Lopez-Martinez, F., Schwarcz, A., Nunez-Valedz, E., Garica-Diaz, V. (2018). Machine learning classification analysis for a hypertensive population as a function of several risk factors. Expert Systems with Applications, 110, 206 - 215.

[13] Ye et al. (2018). Prediction of incident hypertension within the next year: Prospective study using statewide electronic health records and machine learning. Journal of Medical Internet Research, 20(1), e22.

[14] LaFreniere, D., Zulkernine, F., Barber, D., Martin, K. (2016). Using machine learning to predict hypertension from a clinical dataset.