Heart Disease Classification

Paige Gonzales

November 10, 2023

Contents

Initial Project

I'll start by giving an overview of the original assignment. Then, I will talk about where the data came from, why it was chosen, and any potential bias or privacy issues with the data set. After, I will move on to describe details about the initial project, including data pre-processing, model training, testing, and evaluation. These components are all important background information before discussing the enhancements I'd like to make to this project. Next, I will talk about some of the enhancements, and then explain the process implemented to achieve my desired results. Finally, I will reflect on any issues I encountered while executing this project and discuss any further work involving this data.

Project Background

This project was originally completed in the Spring of 2023 in my Data Mining class. I worked with two other group members with the goal of addressing the full data analytics life cycle. Our objectives were to find a data set that was interesting to us, explore and visualize the data, apply machine learning algorithms, evaluate performance, and present our findings to our peers. We used R and RStudio for all the data processing, modeling, and evaluation, and PowerPoint to analyze and communicate our results. Several

supervised machine learning algorithms were used, including random forest, naïve Bayes, bagging, decision tree, and boosted logistic regression.

My personal contribution included data cleaning, such as converting coded survey responses to a uniform format and imputing missing values, applying machine learning algorithms, specifically random forest, naïve Bayes, bagging, decision tree, and boosted logistic regression, evaluating model performance, and designing the final presentation.

Data Source and Integrity

The data set used for this project contains responses from the Behavioral Risk Factor Surveillance System (BRFSS) 2021 annual phone survey. The results for the survey are publicly available for download directly from the Center for Disease Control (CDC) website. The data set has records for 438,693 survey participants and 303 features, each representing an individual coded response for each survey question. Questions were asked primarily on demographic information, lifestyle, and medical history. The data is highly structured, with no major anomalies or blank cells. For an analyst, the structure of the data is a bit of a hindrance because all the responses require heavy reference to a code book. All the coded responses with descriptions can be found on the CDC website.

Since this data set is made public by the CDC, it does not contain any names, addresses, or personal contact information. As for data integrity, it is possible the data set has some bias since these responses are from a cohort of individuals who are willing to participant in a lengthy phone survey. In addition, the survey takers might have answered inaccurately in some responses, either because they are unsure of the answer or are unwilling to be truthful.

Motivation and Objectives

My team chose this project because heart disease causes a significant number of deaths in the Unites States and costs billions annually. According to the CDC, about 5% of Americans over 20 years old have coronary artery disease. Further, in 2020, 20% of American deaths were from heart disease, making it the leading cause of death in the United States among all genders and nearly all racial and ethnic groups. From 2018 to 2019, heart disease cost the United States nearly $240 billion in medical and pharmaceutical costs, as well as loss of productivity from those who are diagnosed with the disease (CDC, 2023a).

The purpose of this analysis is to use survey responses from individuals in the United States to predict if the person answered in the survey that they have heart disease. This project serves as an experiment to see if the results from the survey mimic the reality of the current state of heart disease in the United States, as well as if the survey questions can accurately predict how the survey taker answers the questions about heart disease.

We combined two survey questions into one target variable:

1. (Ever told) (you had) angina or coronary heart disease (CAD)?
2. (Ever told) you had a heart attack, also called a myocardial infarction (MI)?

If the person reported having had a MI or CAD, then the feature value is 1, if not it is 2. Otherwise, the response is blank.

Data Exploration and Pre-processing

In our initial data exploration, we found that 8% of individuals reported having CAD or MI, which is 3% more than the reported 5% of people over the age of 20 in the United States provided by the CDC (CDC, 2023a). A possible reason for this difference is the demographic bias in the cohort of individuals who took

the survey. In Figure 1, notice that over 50% of the survey takers were over the age of 55. The same can be said for the distribution of survey takers by region. The highest percentage of survey takers are located in the Midwest and South, with the two lowest percentages being the West and United States territories. The true population, according to the United States Census Bureau, shows the northeast has the lowest percentage of 17.1% and the western population is the second highest, accounting for 23.6% of the total population of the United States (Bureau, n.d.). This indicates there might be further bias based on the area of the country the participant lives in.

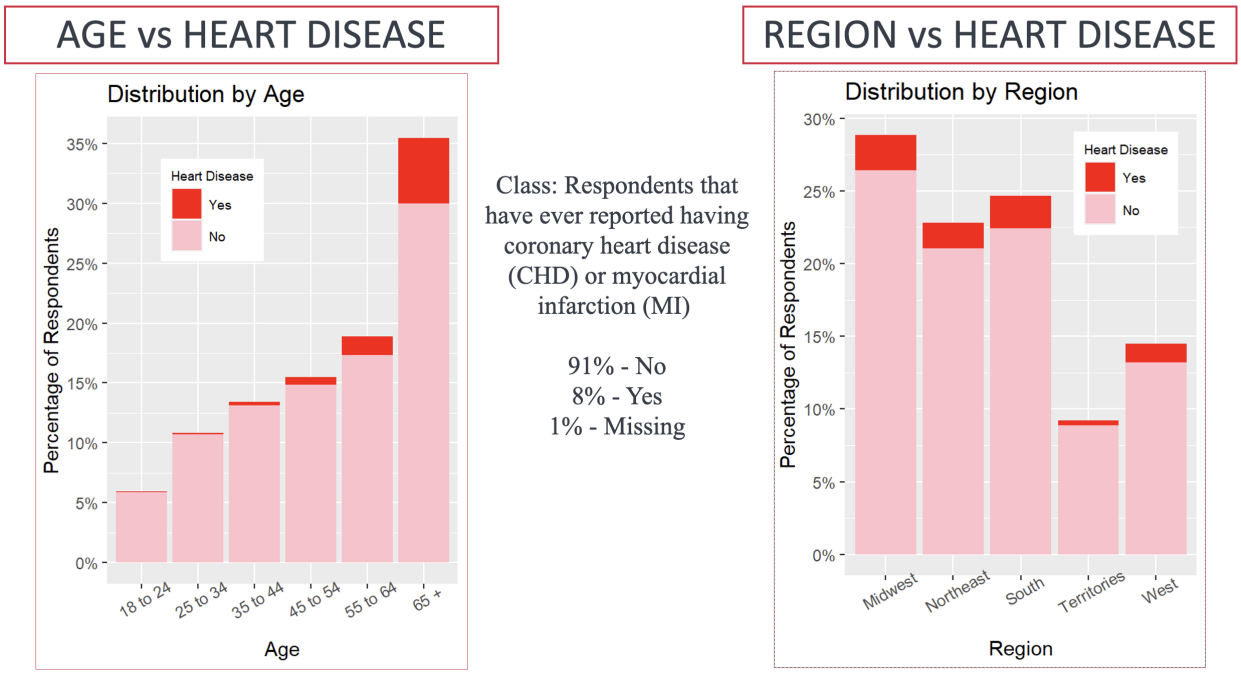

Figure 1: Demographics of Survey Respondents

While my team and I did try our best to complete the data cleaning as thoroughly as possible in the time frame given, the data set is extensive, and we were forced to make a few assumptions and generalizations. Because we started this project in the beginning of our class, we had limited knowledge on how to handle high dimensional data and missing values. We chose to select our features based on subject matter knowledge from the American Heart Association (AHA). Some important factors that contribute to heart disease, according to the AHA, are high cholesterol, diabetes, obesity, as well as others (AHA, 2022). Selecting only these survey questions reduced the data set from 303 features down to 21. Even with this significant reduction in variables, the data set is still quite large for working on a personal computer and requires high computational power and processing time.

It must be heavily emphasized that this data is difficult to pre-process. The raw data set is coded into numeric values representing categorical responses to survey questions. Each of the questions has different code values, so there is not a straightforward or simple way to convert all the numbers to their respective descriptions. Figure 2 shows an example of the variation from survey question to survey question. Notice the variable *PHYS14D* has code values that represent entirely different responses than those of *SMOKER3*. In the question asking if the person's smoker status, the code value 1 represents that they currently smoker. In the question about physical health, value 1 equates to the person saying they have had zero days in the past thirty days that their physical health is not good.

Label: Computed Smoking Status
Section Name: Calculated Variables
Module Number: 11
Question Number: 1
Column: 2009
Type of Variable: Num
SAS Variable Name: _SMOKER3
Question Prologue:
Question: Four-level smoker status: Everyday smoker, Someday smoker, Former smoker, Non-smoker

Value	Value Label	Frequency	Percentage	Weighted Percentage
1	Current smoker - now smokes every day Notes: SMOKE100 = 1 and SMOKEDAY = 1	38,913	8.87	8.83
2	Current smoker - now smokes some days Notes: SMOKE100 = 1 and SMOKEDAY = 2	14,919	3.40	3.77
3	Former smoker Notes: SMOKE100 = 1 and SMOKEDAY = 3	113,247	25.81	22.17
4	Never smoked Notes: SMOKE100 = 2	246,644	56.22	58.90
9	Don't know/Refused/Missing Notes: SMOKE100 = 1 and SMOKEDAY = 9 or SMOKE100 = 7 or 9 or Missing	24,970	5.69	6.32

Label: Computed Physical Health Status
Section Name: Calculated Variables
Module Number: 2
Question Number: 1
Column: 1900
Type of Variable: Num
SAS Variable Name: _PHYS14D
Question Prologue:
Question: 3 level not good physical health status: 0 days, 1-13 days, 14-30 days

Value	Value Label	Frequency	Percentage	Weighted Percentage
1	Zero days when physical health not good	287,796	65.60	66.53
2	1-13 days when physical health not good	90,419	20.61	20.70
3	14+ days when physical health not good	50,984	11.62	10.77
9	Don't know/Refused/Missing	9,494	2.16	2.00

Figure 2: Variation in Coded Responses

The data set also has a high number of unanswered survey questions. If we chose to remove all surveys that were incomplete for the selected features, the data set would have been reduced to 177,019 instances, which is 40% of the original size. This could remove significant patterns in the data, which would compromise the validity of the modeling. To handle this issue, we chose to impute missing values based on mean, median, and mode, depending on whether the feature was categorical or numerical. For numerical variables, the data was imputed using the mean or median, whereas categorical variables were imputed using the mode. Prior to calculating these statistics, we grouped the variables by major demographics, such as sex, age, and region of residence to calculate a more accurate imputation value. This method required a significant amount of time to complete because of the coded format of the data. Each variable had to be cross-referenced to the CDC code book, manually changed, examined, and imputed. Figure 3 shows an example of one of the variables we manually cleaned up and imputed. To go through this process for each of the 21 variables took hours of effort and over 500 lines of code.

```
```{r codeEx, eval = FALSE, echo = TRUE}

round(prop.table(table(heart$X_PHYS14D, useNA = "always")) * 100, digits = 2)

#Mode by groupings to check which mode to use
heart %>% group_by(X_AGE_G) %>% summarise(mode = getmode(X_PHYS14D))
heart %>% group_by(SEXVAR) %>% summarise(mode = getmode(X_PHYS14D))
heart %>% group_by(X_REGION) %>% summarise(mode = getmode(X_PHYS14D))

#Impute using mode
heart <- heart %>% mutate(X_PHYS14D = ifelse(X_PHYS14D == 9, NA,
                                             X_PHYS14D))
heart <- heart %>% mutate(X_PHYS14D = ifelse(is.na(X_PHYS14D), getmode(X_PHYS14D),
                                             X_PHYS14D))

```
```

Figure 3: Example of manual cleanup required for one variable

Data Modeling and Evaluation

When we finally completed data pre-processing, we began the data modeling and evaluation process. There were two evaluation metrics used to evaluate model performance:

1. **Sensitivity**: Sensitivity is the true positive rate, or how many predicted positives are correct out of all true positives (Ajitesh, 2023). We are predicting an outcome that involves human life, so we want a low number of false negatives. In implementation it would be of high risk to falsely diagnose a person as not having heart disease when they do have the disease. We chose this value as our most important evaluation metric.

2. **Kappa**: This data set has a class imbalance, so accuracy is not the best measure for model appropriateness. For example, in this particular data set, 8% of the respondents said they don't have heart disease, as shown in Figure 1. That means if we always predicted they do **not** have heart disease, then we would be 92% accurate. When evaluating the supervised learning models, the accuracies will be high, but it doesn't tell us anything about the model effectiveness. The kappa statistic is a way to avoid this problem because it takes into account the probability of being correct due to chance (Scientist, n.d.).

We trained five different supervised learning algorithms: random forest, naïve Bayes, bagging, decision tree, and boosted logistic regression. Each model was trained using custom parameter tuning and 10-fold cross validation with sampling up to account for the class imbalance. Our initial results are shown in Figure 4. Out of the five machine learning algorithms trained, the logistic regression model was the best fit in predicting heart disease based on the chosen features. This model was 10% less accurate than a no information rate,

the rate at which we would be correct by always choosing that the person did not have heart disease (Lantz, 2019). However, the sensitivity was 83%, which we considered a success for such a complex data set.

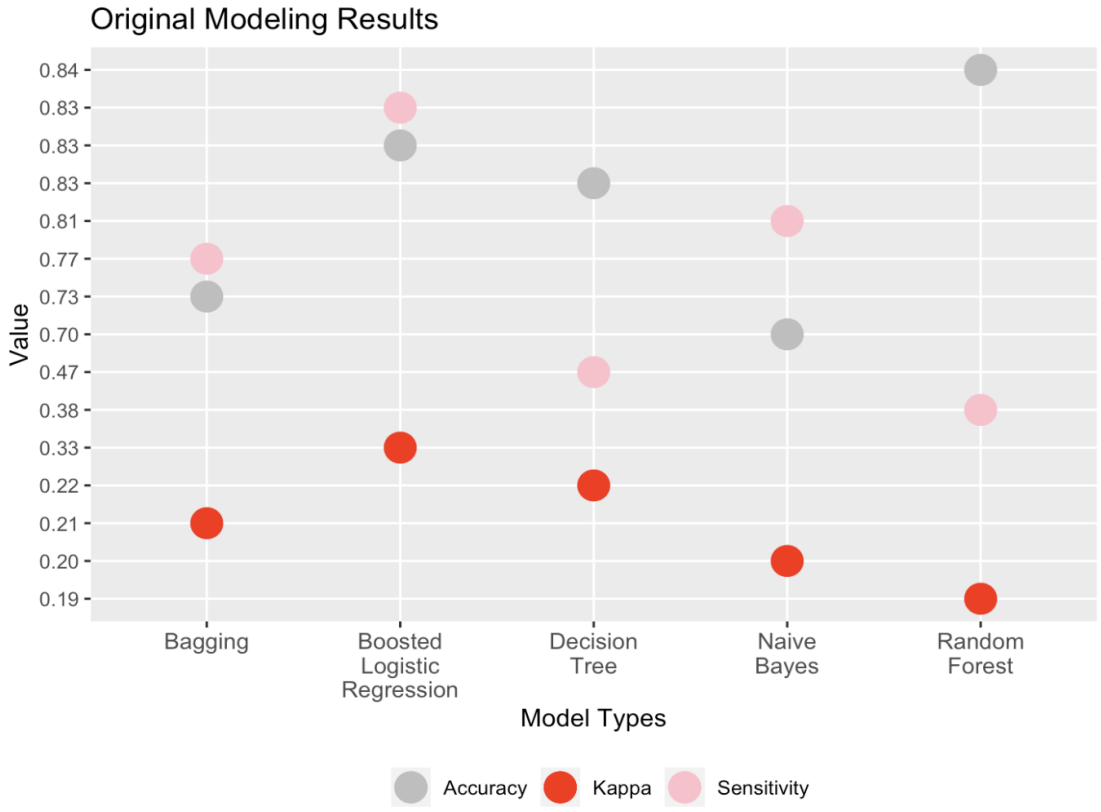

Figure 4: Initial Model Results

Project Expansion

Goals for Improvement

While this project was a helpful practice in data modeling and data pre-processing, there were some major hurdles we encountered that could have been handled using alternative methods. There are three improvements I will be doing for this project:

1. As mentioned previously, it is difficult to understand what the variables represent when they are left as coded values. This makes it difficult for interpretation and creation of visualizations. My first goal is to automate the re-coding process to reduce manual cleanup time. If possible, this automation will be generalized enough so that it can be applied to different data sets.

2. When we first cleaned this data, some values were listed as "Don't know/not sure" or "Refused." We considered these missing values to be "No" because going through each variable for the coded value required more time than we had available. In hindsight this was a large assumption. It is possible that a person who answers they don't know or refused to answer could be a certain type of individual who may have heart disease. For example, it is possible people who are knowingly unhealthy refuse to answer because of embarrassment. It is also possible that people who don't know the answers to personal health questions are uneducated on their health or don't often visit the doctor. This could

have skewed the data and caused modeling to be less successful in predicting outcomes. I want to see if the performance of the model improves when these values are kept in the training data rather than removed. The automation of de-coding the variables will allow these values to be kept in the data set and considered as values when training models.

3. Lastly, I want to revisit our approach to dimensionality reduction. As mentioned, we chose a few questions from the survey that contribute to heart disease according to the American Heart Association. In class, we briefly discussed the use of decision trees to reduce the number of features to only those that most contributed to predicting the target variable. I will start with the larger data set and reduce the number of dimensions by first creating a decision tree and choosing the variables that have the most information gain. I will use only these variables for machine learning algorithms to see if the models improve.

Expanded Results

Dimensionality Reduction and Data Cleaning The original data set has 303 features, all containing coded responses. To visit each feature individually would be a tedious task. Additionally, many of the features in the data set are derived calculations from the other variables, redundant, or not asked to everyone in the cohort of participants. Before doing any data cleaning, I removed all features that are derived from other variables. There are three exceptions: age, body mass index (BMI) and the target variable. The variable for age range is an important factor in determining heart disease, and there was no other feature in the data set that represented age, most likely due to privacy for the individuals. Additionally, I kept BMI out of convenience rather than spending time calculating the BMI based on the height and weight provided in two other variables. As mentioned previously, the target variable is a calculated variable which combines two survey questions: a) (Ever told) you had a heart attack, also called a myocardial infarction? b) (Ever told) (you had) angia or coronary artery disease?

Additionally, I removed any features that are for recording purposes, such as the date the interview took place, whether it was taken on a land line or mobile phone, and survey identification numbers. This removed a significant chunk of features, but still too many to justify manually filling in all the codes. My solution first was to remove any variables that have more than 50% of the responses missing. All missing values will be later imputed, and I didn't want to train a supervised learning model based on features that have over half of the values extrapolated.

Now that many of the features had been removed, I downloaded a SAS file on the CDC website that has all the variables, code values, and code descriptions for this data set (CDC, 2023b). I then performed data cleaning on this code file in RStudio so that I could use it to loop through and apply the descriptions of the codes to the data set. Some variables required manual cleaning even with this more generalized approach. Figure 5 shows a before and after of some of the variables from the data set. Notice that the bottom data set is easier to interpret for the reader, as well as easier to explore and analyze for the analyst.

PRIMINSR	PERSDOC3	MEDCOST1	CHECKUP1	EXERANY2	BPHIGH6
3	1	2	2	2	3
1	2	2	1	1	1
2	2	2	1	2	1
2	1	2	1	1	1
3	1	2	1	1	4

PRIMINSR	PERSDOC3	MEDCOST1	CHECKUP1	EXERANY2	BPHIGH6
Medicare	Yes, only one	No	Within past 2 years (1 year but < 2 years ago)	No	No
A plan purchased through an employer or union (incl...	More than one	No	Within past year (anytime < 12 months ago)	Yes	Yes
A private nongovernmental plan that you or another f...	More than one	No	Within past year (anytime < 12 months ago)	No	Yes
A private nongovernmental plan that you or another f...	Yes, only one	No	Within past year (anytime < 12 months ago)	Yes	Yes
Medicare	Yes, only one	No	Within past year (anytime < 12 months ago)	Yes	Told borderline high or pre-hypertensive or elevated ..

Figure 5: Before and After Recoding Process

The next step was imputation. I would still impute values based on mean, median, or mode, grouped by age, sex, and region of residence, but would create a function to create a systematic process for doing so, instead of writing code for each variable manually. For categorical variables, I created a function that would calculate the mode of each feature grouped by either the region the person lives in in the Unites States, their gender, or their age group. If the mode is the same across all groupings, the mode of the entire feature would be used to impute the missing values. If only one of these groupings has variance, I computed the mode based on that group. By the end of this process, there were only three variables that required manual examination, which I assessed to determine the best grouping for imputation.

The numeric variables were grouped by age, gender, and region, but instead of mode, I used median or mean, depending on the distribution of the data. The data set has only 5 numeric variables and each represents different types of information, so I did this manually.

After these first steps, there were 434,058 rows and 48 variables left, which is a significant reduction in size from the original data, but still requires a lot of computational memory when training machine learning models. Additionally, such a high dimension of data could also create a high variance when modeling and erroneous results. As stated, my solution to this issue is to create a decision tree. In decision tree algorithms, information gain is a calculation that indicates how much the target variable shows homogeneity when split upon a particular feature (Lantz, 2019). If the information gain is very low, then even when you split on this feature, there is no distinction between who has or doesn't have heart disease, effectively making it a useless variable. The decision tree showed 16 variables with high information gain, which I will use to train my models. Figure 6 shows the decision tree created using the *rpart* and *rpart.plot* package (Milborrow, 2022) (Therneau, Terry [aut], Atkinson, Beth [aut, cre], Ripley, Brian [trl], 2022). The first feature to split upon is the question, "Are you currently taking medicine prescribed by your doctor or other health professional for your cholesterol?" This question seems intuitive since having high cholesterol is one of the leading contributors to heart disease according to the American Heart Association (AHA, 2022). Some other features kept in the data set are if the person has good general health and their employment status. While some of these features were selected for the initial project, such as the cholesterol question, many were not, so it will be interesting to see if the modeling improves at all.

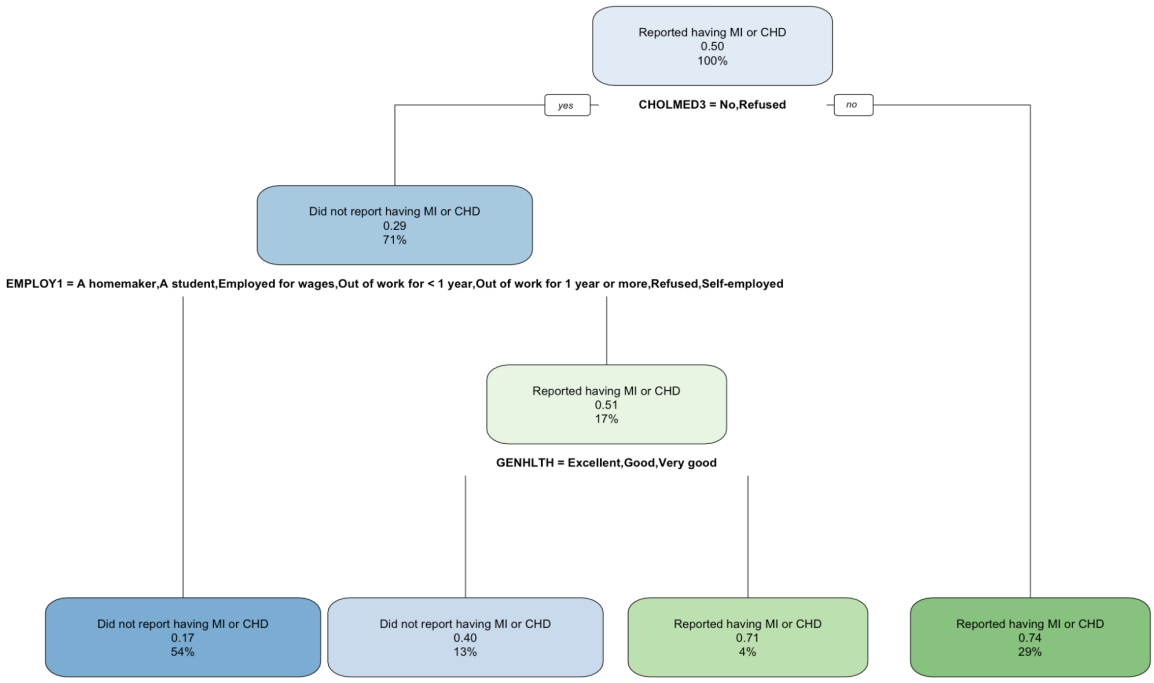

Figure 6: Decision Tree for Dimensionality Reduction

The final data set used for training and testing consisted of 434,059 instances and 16 variables. To summarize, Figure 7 shows the workflow taken to clean up this data set. I successfully was able to reduce the dimensions strategically, decode the values, and keep the "Don't know/not sure" and "Refused" responses for modeling.

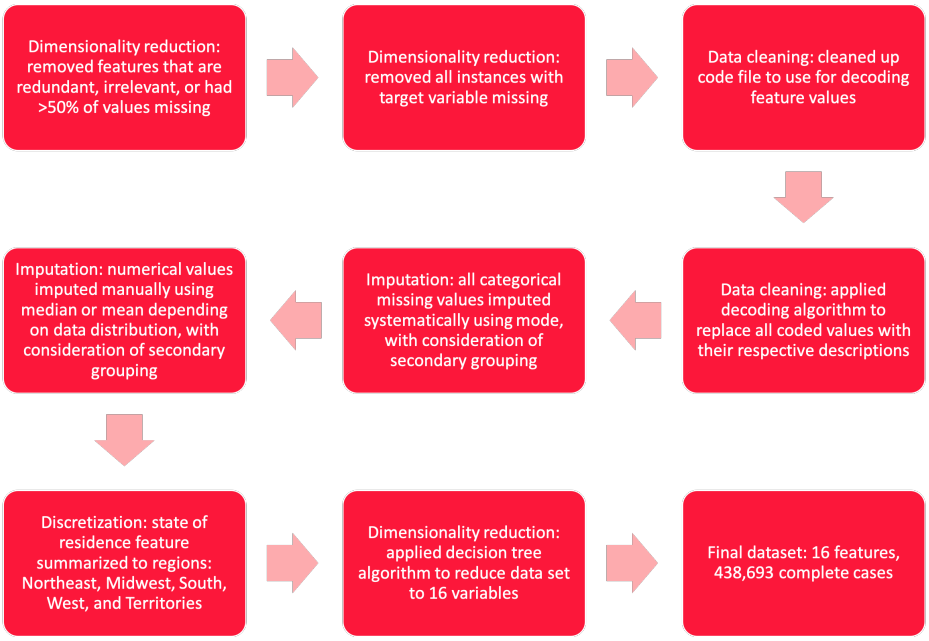

Figure 7: Improved Data Pre-Processing Workflow

Modeling and Evaluation I selected the training data set based on a random sample of 75% of the final data. The remaining 25% would be used for testing, which was the same proportion used in the initial project. The custom parameter tuning was kept the same for four of the five models using the *CARET* package in R (Kuhn, 2019). We chose 10-fold cross validation and model choice within one standard error of the best model to help improve model performance. We also chose sampling up since there is a significant class imbalance to this data set, with the class of interest representing only 8% of the total (Lantz, 2019). I had some issues running the naïve Bayes model using the CARET package, which could not be resolve by the time this report was finished. Instead, I chose a simpler naïve Bayes modeling algorithm from the *e1071* package, which uses no parameter tuning or cross validation (Meyer, 2023). However, the results are still better than the initial project.

In our initial results, the most impressive model was found to be boosted logistic regression. Figure 8 shows the difference in results between the original trained models and the new models. The sensitivity values show improvement in the boosted logistic regression, decision tree, and random forest models. In four out of the five models the kappa statistic improved. Interestingly, the only model that performed worse based on the kappa statistic is the boosted logistic regression model, the one we selected to put into production initially. Accuracy improved for bagging, and naïve Bayes, but none of the other models. The best model, considering the need for high sensitivity and kappa statistic, remains the boosted logistic regression.

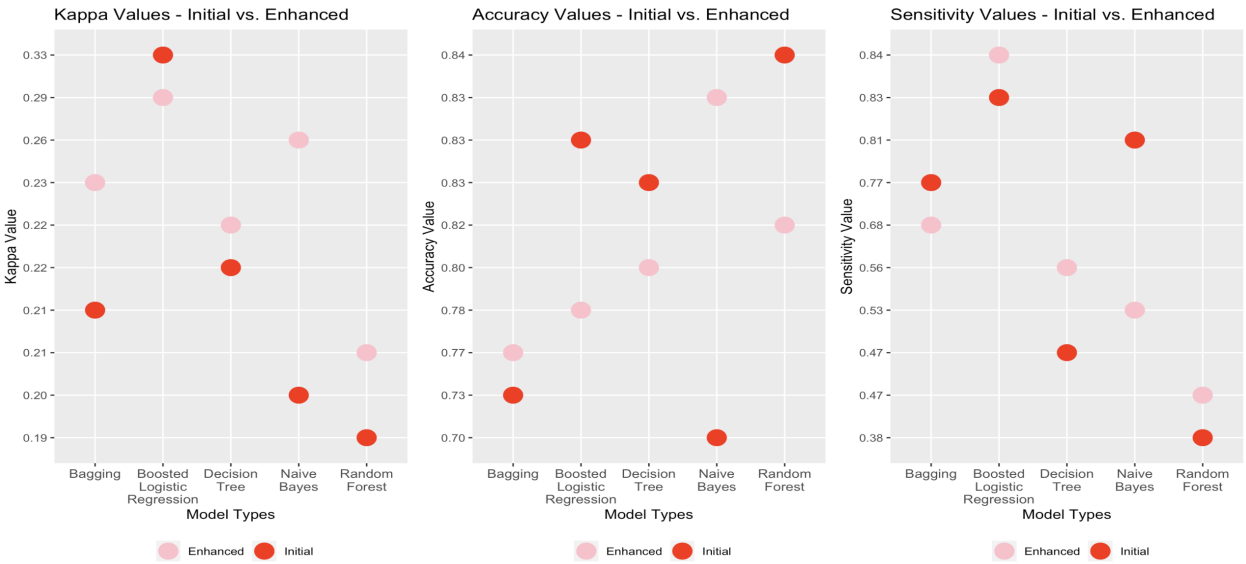

Figure 8: Comparison of Evaluation Metrics: Initial Project vs. Updated Project

Conclusion, Challenges and Further Work

Overall, this project has been a great learning experience for data pre-processing and modeling several supervised learning algorithms. I was successfully able to automate most of the conversion of code values to code descriptions, as well as automate much of the imputation for the categorical features. This project has also been great practice in handling dimensionality reduction for categorical data. I was able to successfully use a decision tree algorithm to reduce the number of features.

A major challenge for this data set has been the data cleaning, taking many hours of coding. Converting the coded values to responses that are easily interpreted by the user required extensive coding and tedious and time-consuming work to manually go through each feature to verify correctness. The codes are not the same for each question, so the formatting had to be done carefully with a significant amount of cross validation. Handling missing values has also been time consuming since there are both numerical and categorical features, and nearly every feature contained some number of missing values.

Dimensionality reduction has been a challenge because despite removing many rows, the data set is still very large, which causes issues with modeling time and accurate predictions. Here we have an example of data set with complex patterns, and although the model shows decent results, it is possible even with the reduction we have an instance of the "curse of dimensionality."

For future work, I would choose to approach this data set instead from a purely data visualization and exploratory data analysis standpoint, as I don't feel the modeling portion can be improved much. However, this survey does provide valuable insight into the behavior of United States citizens year over year and would be interesting to follow in that regard.

References

AHA. (2022). *Understand your risks to prevent a heart attack.* https://www.heart.org/en/health-topics/heart-attack/understand-your-risks-to-prevent-a-heart-attack

Ajitesh, Kumar. (2023). *Machine learning – sensitivity vs specificity difference.* https://vitalflux.com/ml-metrics-sensitivity-vs-specificity-difference/

Bureau, U. S. C. (n.d.). *U.s. And world population clock.* https://www.census.gov/popclock/data_tables.php?component=growth

CDC. (2023a). *Heart disease facts.* https://www.cdc.gov/heartdisease/facts.htm

CDC. (2023b). *LLCP 2021 codebook report overall version data behavioral risk factor surveillance system.* https://www.cdc.gov/brfss/annual_data/2021/pdf/codebook21_llcp-v2-508.pdf

Chronic Disease Prevention, N. C. for, & Health Promotion, D. of P. H. (2023). *2021 BRFSS Survey Data and Documentation.* https://www.cdc.gov/brfss/annual_data/annual_2021.html

Grolemund, H. W., & Garrett. (2017). *R for data science.* https://r4ds.had.co.nz

Kuhn, M. (2019). *The caret package.* https://topepo.github.io/caret/

Lantz, B. (2019). *Machine learning with r.* Packt Publishing. https://learning.oreilly.com/library/view/machine-learning-with/9781788295864/

Meyer, cre]., David [aut. (2023). *e1071: Misc functions of the department of statistics, probability theory group (formerly: E1071), TU wien.* https://cran.r-project.org/web/packages/e1071/index.html

Milborrow, S. (2022). *Rpart.plot: Plot 'rpart' models: An enhanced version of 'plot.rpart'.* https://cran.r-project.org/web/packages/rpart.plot/index.html

Scientist, T. D. (n.d.). *Performance measures: Cohen's kappa statistic.* https://thedatascientist.com/performance-measures-cohens-kappa-statistic/

Therneau, Terry [aut], Atkinson, Beth [aut, cre], Ripley, Brian [trl]. (2022). *Rpart: Recursive partitioning and regression trees.* https://cran.r-project.org/web/packages/rpart/index.html