Project 2 – CHD

Author(s):

Gabriel Jackson

Naad Kundu

Thao Nguyen

Kayla Nguyen

Halbert Nguyen

Ben Willoughby

Omar Zeineddine

DS 3001: Foundations of Machine Learning

April 16, 2024

Summary

Our main focus for this project was to build predictive algorithms to determine the likelihood of a person developing coronary heart disease. We utilized many variables such as sex, age, education, current smoker status, cigarettes per day, etc. We then used the provided data to build our training model and to test our model. First, we cleaned our data by updating any NaN values. We needed to clean several variables, including education, glucose, BPMeds, totChol, BMI, cigsPerDay, and heart rate, in terms of NaNs. We then saved these cleaned datasets into updated CSV files. After cleaning the data, we used Linear Regression, K-Nearest Neighbors Classifiers and Decision Tree Classifiers to predict the likelihood of developing coronary heart disease. For our linear models, we used the SKLearn library. Our linear regression model results were interesting, particularly due to our dummy education variable. We decided to compare two linear models: one with all variables and one excluding the education variable. We found that the adjusted R^2 values for the model with all variables were higher than those for the model without the education variable, indicating that the model with all variables had more explanatory power for predicting CHD. Our decision tree model also yielded interesting results—it had an 85% accuracy rating. However, it did end up having a lot of false positives, incorrectly predicting CHD in 143 cases where the person did not have CHD. The importance of education in predicting CHD suggests potential socio-economic factors influencing heart health, warranting further investigation into how these predictors interact. For our last model, we used a K-Nearest Neighbors Classifier. The goal was to classify the data based on the distance to their neighbors. It would find a most common classification among their neighbors, and the model was run with anywhere between 1-150 neighbors. Our results showed that the ideal number of neighbors was 6, giving an accuracy of about 85%. Our next steps will focus on refining the models by integrating additional variables and employing more advanced machine learning techniques to enhance predictive accuracy and reduce false positives. This continued refinement will aim to minimize overfitting while exploring the impact of unexamined variables.

Data

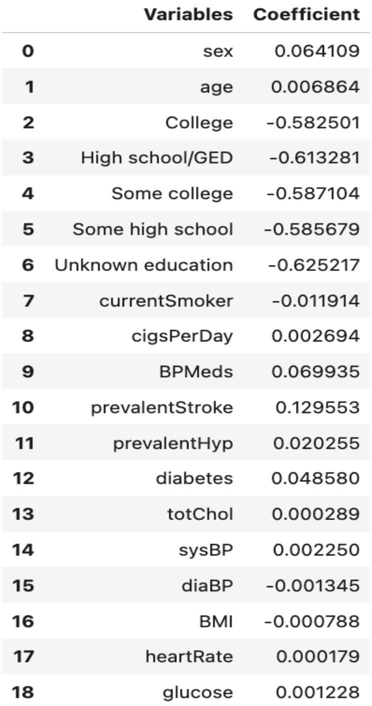
In terms of the data, two CSV files were provided–one for training our model and one to test the model. The variable we are trying to predict is ‘10YearCHD’, which represents the 10-year risk of coronary heart disease (as a 0 or 1). Initially, we had to clean the training dataset. For our education variable, we replaced all NAs with 0s, in order to keep its value numerical. For our glucose variable, we filled all NAs with 85, which is the human average. This was done to keep the data from being skewed for our models. For our BPMeds variable, we replaced all NAs with 0 to indicate that they are not being taken. For our totChol (total cholesterol), BMI (body mass index), cigsPerDay, and heartRate variables, we decided to drop all the NANs. We then saved these cleaned datasets into their respective updated CSV files to prepare them to be used by both our linear regression model and our decision tree. To begin our linear regression models, we first updated our education variable. One big challenge we encountered was with our education variable. The education variable was categorical, but it appeared numerical. We proceeded to change the entries into text, so that we could easily create dummy variables. We replaced its values from 0-5 to ‘Unknown education’, ‘Some high school’, ‘High school/GED’, ‘Some college’, ‘College’, respectively. Overall, our dataset contains multiple dummy variables. For many of the variables, the representation is intuitive, as 1 is true and 0 is false. But for our sex variable, 0 represents female and 1 represents a male. We proceeded to construct a linear model with no intercept, using the variables sex, age, our education dummy variable, and currentSmoker, cigsPerDay, BPMeds, prevalentStroke, and prevalentHyp. Calculating R^2 and adjusted R^2, we got 0.0998 and 0.0943 respectively. We proceeded to make another linear model, this time without the education variable, and our R^2 was 0.0832 and our adjusted R^2 was 0.0790. It didn’t seem like education played much of a factor in coronary heart disease, which is why we decided to create two models- one with the education variable and one without it. Looking at the R^2 values, it seems like the model with all available variables, including education, had more explanatory value due to the higher adjusted R^2 value. We encountered some challenges when trying to make our predictive model. The variable we were trying to predict, 10YearCHD, can only take a binary value of 0 or 1, meaning we have a linear probability model. This implies that the predicted values are understood as probabilities of the event occurring. Also, under linear probability models, the regression line will never fit the data perfectly if the dependent variable is binary and the regressors are continuous. This means the R^2 value, our primary tool for measuring the explanatory power or 'usefulness' of a model, loses its interpretation. Since linear models predict probabilities, we decided to try and use a decision tree as well, to compare our results.

To make a decision tree, writing the code was fairly simple. We also used the sklearn library for the decision trees. We used the same cleaned data files as we did for our linear models. A couple of challenges did arise though- the biggest one was attempting to limit the amount of false positives the decision tree kept giving us. The decision tree incorrectly predicted yes for 143 cases when they had no coronary heart disease. The accuracy was 85%.

Results

In order to predict the likelihood of a person developing Coronary Heart Disease, three predictive models were considered using training and test data from the Framingham Heart Study. This data includes the following variables: sex, age, currently a smoker, cigarettes per day, use of blood pressure medication, stroke prevalence, prevalent hypertension, diabetic, total cholesterol, systolic and diastolic blood pressure, body mass index, heart rate, glucose, and the 10 year risk of coronary heart disease. Overall, the three predictive models used were K-Nearest Neighbors Classifier, Decision Trees Classifier, and linear regression.

The first predictive algorithm used was linear regression. The linear regression model used the training data to create an equation where each variable and its associated weight are used to make a prediction of whether or not a person is likely to develop Coronary Heart Disease within the next 10 years. The code for the linear regression model can be found under ‘linear_regression.ipynb’. This code generated a table for each variable and their respective weights (under ‘coefficients’), which can be seen in Figure 1:



	Variables	Coefficient
0	sex	0.064109
1	age	0.006864
2	College	-0.582501
3	High school/GED	-0.613281
4	Some college	-0.587104
5	Some high school	-0.585679
6	Unknown education	-0.625217
7	currentSmoker	-0.011914
8	cigsPerDay	0.002694
9	BPMeds	0.069935
10	prevalentStroke	0.129553
11	prevalentHyp	0.020255
12	diabetes	0.048580
13	totChol	0.000289
14	sysBP	0.002250
15	diaBP	-0.001345
16	BMI	-0.000788
17	heartRate	0.000179
18	glucose	0.001228

Figure 1: The linear regression table of each variable and each variable’s coefficient.

As can be seen from Figure 1, there were some variables that impacted the risk of developing Coronary Heart Disease the most: prevalence for stroke had a coefficient of roughly 13%, taking blood pressure medication had a coefficient of 6.9%, and being male had a coefficient of 6.4%. The higher magnitude coefficients show which of the variables have a greater impact in developing CHD in the next 10 years. Additionally, it seems that the education variables (‘High School/GED’, ‘College’, ‘Some college’, ‘Some high school’, and ‘Unknown Education’) all had a coefficient of roughly -0.6 with some variation among the coefficients. One theory as to why this happened could be how every person fits one of the categories for education which causes the model to shift the risk of any person by roughly -0.6%. This high coefficient is likely to reflect how each person is relatively safe against developing CHD before considering the other variables. The linear regression model had an adjusted R^2 value of roughly 0.094, which normally means that the model does not fit the data well. One explanation for this poor R^2 value is that the regression line can never fit the data perfectly when the dependent variable is binary and the regressors are continuous. This means that the R^2 value, which is typically used for measuring the explanatory power or ‘usefulness’ of this model, loses its interpretation.

The next predictive model used was K-Nearest Neighbor Classification. The code for this model can be found under the ‘knn.ipynb’ Jupyter Notebook file. The KNN model was used to classify the data based on their distance to a certain number of their neighbors. It will then classify data based on the most common classification among these neighbors. To determine the correct number of neighbors to use, the KNN model was run from 1 to 150 neighbors, where the ideal numbers of neighbors can be selected based on that model’s accuracy. From these iterations, the ideal number of neighbors was found to be 6, which gave the highest accuracy score of roughly 85%.

The final predictive model used was Decision Tree Classification, which presented a series of decisions in the form of a tree of nodes that predict if a person is likely to develop Coronary Heart Disease (‘Yes CHD’) or not (‘No CHD’). The code used to create this decision tree can be found under the ‘decision_trees.ipynb’ Jupyter Notebook file. The training and test data were used to create a 5-layer Decision Tree Classifier from SKLearn, which can be seen in Figure 2:

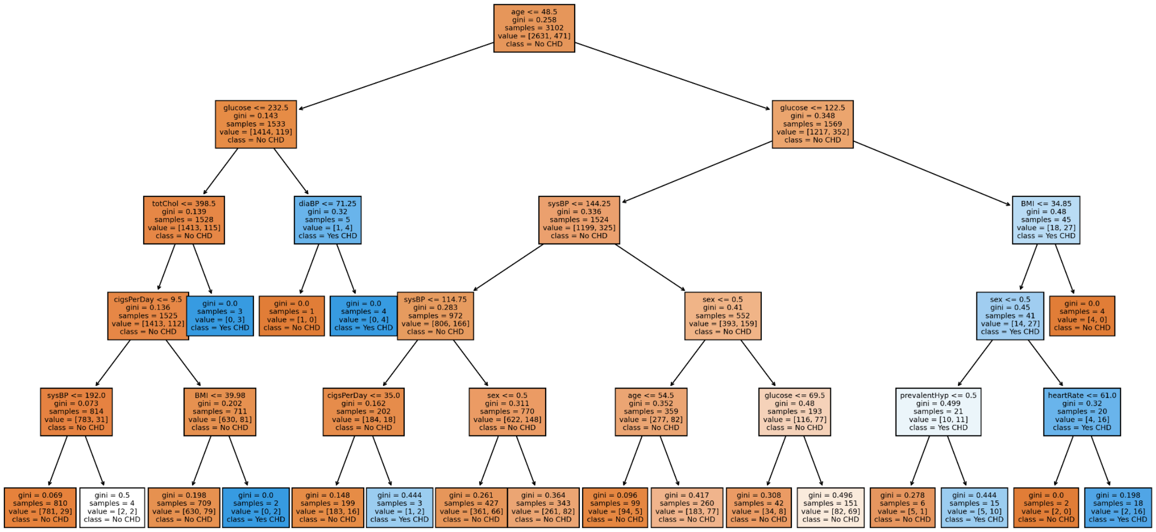

age <= 48.5 gini = 0.258 samples = 3102 value = [2631, 471] class = No CHD
glucose <= 232.5 gini = 0.143 samples = 1533 value = [1414, 119] class = No CHD
glucose <= 122.5 gini = 0.348 samples = 1569 value = [1217, 352] class = No CHD
totChol <= 398.5 gini = 0.139 samples = 1528 value = [1413, 115] class = No CHD
diaBP <= 71.25 gini = 0.32 samples = 5 value = [1, 4] class = Yes CHD
sysBP <= 144.25 gini = 0.336 samples = 1524 value = [1199, 325] class = No CHD
BMI <= 34.85 gini = 0.48 samples = 45 value = [18, 27] class = Yes CHD
cigsPerDay <= 9.5 gini = 0.136 samples = 1525 value = [1413, 112] class = No CHD
gini = 0.0 samples = 3 value = [0, 3] class = Yes CHD
gini = 0.0 samples = 1 value = [1, 0] class = No CHD
gini = 0.0 samples = 4 value = [0, 4] class = Yes CHD
sysBP <= 114.75 gini = 0.283 samples = 972 value = [806, 166] class = No CHD
sex <= 0.5 gini = 0.41 samples = 552 value = [393, 159] class = No CHD
sex <= 0.5 gini = 0.45 samples = 41 value = [14, 27] class = Yes CHD
gini = 0.0 samples = 4 value = [4, 0] class = No CHD
sysBP <= 192.0 gini = 0.073 samples = 814 value = [783, 31] class = No CHD
BMI <= 39.98 gini = 0.202 samples = 711 value = [630, 81] class = No CHD
cigsPerDay <= 35.0 gini = 0.162 samples = 202 value = [184, 18] class = No CHD
sex <= 0.5 gini = 0.311 samples = 770 value = [622, 148] class = No CHD
age <= 54.5 gini = 0.352 samples = 359 value = [277, 82] class = No CHD
glucose <= 69.5 gini = 0.48 samples = 193 value = [116, 77] class = No CHD
prevalentHyp <= 0.5 gini = 0.499 samples = 21 value = [10, 11] class = Yes CHD
heartRate <= 61.0 gini = 0.32 samples = 20 value = [4, 16] class = Yes CHD
gini = 0.069 samples = 810 value = [781, 29] class = No CHD
gini = 0.5 samples = 4 value = [2, 2] class = No CHD
gini = 0.198 samples = 709 value = [630, 79] class = No CHD
gini = 0.0 samples = 2 value = [0, 2] class = Yes CHD
gini = 0.148 samples = 199 value = [183, 16] class = No CHD
gini = 0.444 samples = 3 value = [1, 2] class = Yes CHD
gini = 0.261 samples = 427 value = [361, 66] class = No CHD
gini = 0.364 samples = 343 value = [261, 82] class = No CHD
gini = 0.096 samples = 99 value = [94, 5] class = No CHD
gini = 0.417 samples = 260 value = [183, 77] class = No CHD
gini = 0.308 samples = 42 value = [34, 8] class = No CHD
gini = 0.496 samples = 151 value = [82, 69] class = No CHD
gini = 0.278 samples = 6 value = [5, 1] class = No CHD
gini = 0.444 samples = 15 value = [5, 10] class = Yes CHD
gini = 0.0 samples = 2 value = [2, 0] class = No CHD
gini = 0.198 samples = 18 value = [2, 16] class = Yes CHD

Figure 2: The Decision Classification Tree

The decision tree will make a decision at each node, which after making a series of decisions in the tree leads to a terminal node that will make a prediction on whether or not a person is likely to develop Coronary Heart Disease. The terminal nodes that end in a 'Yes CHD' prediction are colored from a range of white to blue depending on how many samples were used to make this terminal decision. Additionally, the terminal nodes that end in a 'No CHD' prediction are colored from a range of white to orange, depending on how many samples were used to make this terminal decision.

Along with creating this decision tree diagram, the decision tree was tested with the testing data to produce a confusion matrix that determined accuracy, specificity, sensitivity, and MCC score of the tree. The confusion matrix showed 85% accuracy, 0.90% specificity, 98.76% sensitivity, and a MCC score of 10.68%. This means that with a specificity rate of 0.90% there was a high false positive rate and that with a sensitivity of 98.76% that it was good at detecting actual Coronary Heart Disease cases. Therefore, the low specificity and high sensitivity leads to the model favoring 'Yes CHD' strongly. This can be seen in its low false negative rate and its high false positive right. Additionally, the MCC score of 10.68% showed that it was not good at balancing errors made within the tree. However, it did have an accuracy of roughly 85% in the testing data, which demonstrates its ability to correctly classify a person as likely to develop CHD. While this decision classification tree was correct 85% of the time, it can only account for up to 5 nodes (5 decisions in the tree). Using a higher number of layers is possible, but makes the diagram (Figure 2) harder to interpret. Therefore, the researchers chose to use 5 layers to balance readability with performance. This comes at a cost, where roughly 5 decisions can be made to determine if a person is likely or not to develop Coronary Heart Disease (despite there being more than 5 variables). While the tree does not consider all variables (at each decision node), it does indicate which variables are most likely to determine developing CHD, such as a higher age, glucose level, etc. It's also fairly straight-forward to read these variables as they are presented in mostly-human readable form presented as conditions.

While each model did present some amount of predictive ability for the test and training data, there were a lot of issues with using each of these models. For instance, using a classifying algorithm like KNN or Decision Trees didn’t make much sense for this data set. This is because classifiers will group a person as either 100% likely or 0% likely to develop Coronary Heart Disease. In the real world, there is a risk of developing Coronary Heart Disease, which is a range from 0% to 100% (not just simply 0% or 100%). This means that linear regression would make the most sense here, over the other models as it can represent the risk (as a continuous probability) of developing Coronary Heart Disease. However, the issue with linear regression is that the R^2 value was very low, which could be due to the binary nature of the risk for developing CHD in the next 10 years . Another issue with the predictive ability for these models is that there were a lot of features included in the model. This made it difficult to create visualizations and determine which specific variable impacted the chance of developing Coronary Heart Disease the most. Overall, the flaws of each predictive model have made it difficult to use the results to draw meaningful conclusions.

Conclusion

Overall, our project embarked on an ambitious task to predict the likelihood of developing coronary heart disease (CHD) using a subset of data from the Framingham Heart Study. Through the methods of linear regression models and decision trees, we explored various factors that could increase risk factors in developing CHD, such as sex, age, education, smoking habits, and medical history. Our approach required us to focus on the fundamentals: meticulous data cleaning, thoughtful variable transformation, and strategic model selection in order to adhere with the project’s objectives of experimenting with machine learning models, and having fun.

One of the project’s significant findings was the varied impact of education on the risk of CHD, revealed through comparing models with and without the education variable. This result highlights the interesting relationship between socio economic factors and health, suggesting that education level can serve as a precursor to activities that impact heart health. Additionally, the decision tree model further complemented our analysis with an 85% accuracy rate; however, with a good amount of false positives.

This goes to show the complications of predicting binary values (yes/no) in data regarding health. It is important to note that while our decision tree model did generate a higher number of false positives than ideal, this is often a challenge in medical prediction models, particularly when prioritizing sensitivity. In future iterations, adjusting the decision threshold and exploring cost-sensitive learning could better balance the trade-off between sensitivity and specificity, reducing false positives without significantly compromising the ability to detect actual cases of CHD.

As for further exploration, we could incorporate other machine learning techniques such as k nearest neighbor (for regression), in hopes of enhancing accuracy and reducing false positives. Further, exploring the relationship between other variables in the dataset can reveal other potentially unexpected predictors of CHD. Another way we can further elaborate on this data is to account the model for change in risk factors over time. This would provide us a different and dynamic point of view on CHD prediction. Moreover, doing further analysis to include more information such as genetic information, dietary habits, or even psychological markers can add depth to our models. These multidimensional analyses allow for an intricate platter of factors influencing CHD, presenting deeper insights and more detailed predictions.

Wrapping everything up, although our project may be not perfect, it gives us a glimpse into the world of health data science. Going from the raw data to cleaned_data to predictive models is complicated and tedious, but the experience gained from this project is what makes it possible for us to adopt a thirst for knowledge, hopefully understanding more of how we can decrease the risk factors of CHD. As technology progresses and machine learning models become even smarter, we are getting closer to a future where predictive models can provide personalized insights on improving health, increasing human lifespan.

Appendix

No additional plots or tables were useful to include in this write-up, however the source code can be found within the '.ipynb' files in the project's repository's root at:

https://github.com/thaonguyyen/project_chd/