Introduction to Statistical Models and Data Mining

STATS 101C Final Project

Lecture 2

Predicting Alcoholic Status Using Person's Vitals

Daniel Khurgin, Yue Que, Benyamin Uribe, Santsibyllah Seitz, Krystal Ki

1. ABSTRACT

The objective of the Kaggle project is to utilize classification models on the provided dataset to predict alcoholic status using a person's vitals. The report offers a comprehensive overview of the process of building our final model, encompassing introduction, data imputation, classification methods, hero model, as well as discussions on the limitations and potential improvements of our work.

Our hero model is built on neural networks. The following predictors were used in the model: sex, age, waist, DBP, BLDS, HDL_chole, LDL_chole, triglyceride, hemoglobin, urine_protein, serum_creatine, SGOT_AST, SGOT_ALT, gamma_GTP, and Smoking.Status. Achieving a final accuracy score of 0.73120, our model ranked 23rd in the Kaggle competition.

2. INTRODUCTION

Alcohol is often discussed as a core symbol of social engagement and bonding in various cultures. However, excessive alcohol consumption poses a substantial risk to health, manifesting in the most severe form of alcohol abuse known as alcoholism. Also known as alcohol use disorder, alcoholism involves the inability to control drinking habits and often causes harmful side effects. Globally, approximately 107 million people have an alcohol use disorder, with the prevalence peaking among those aged 25 and 34, and over twice as prevalent among males compared to females. Moreover, alcohol consumption causes 2.8 million premature deaths globally every year.

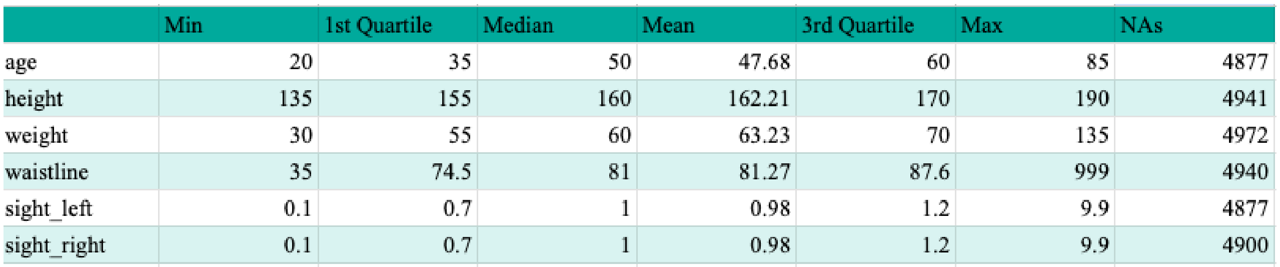
Conventional diagnostic approaches for alcoholism heavily rely on self-reporting and clinical assessments, which may be subject to biases and limitations. To address this, our project utilizes individuals' vitals, such as cholesterol, triglyceride, and hemoglobin levels, to predict alcoholic status. In this Kaggle project, we used the dataset from the National Health Insurance Service in Korea which contains 26 numerical and categorical predictors. The training dataset contains 70,000 observations, while the testing dataset contains 30,000 observations. The goal of the project is to use this alcohol drinking dataset to build and predict models to classify the individuals' alcoholic status.

3. DATA ANALYSIS

Before beginning the process of fitting a model to the data, we chose to analyze the variables in the dataset in order to gain a better understanding of each one as well as what their relationship with the response variable may be. This analysis can be split into that of the numerical variables and that of the categorical variables.

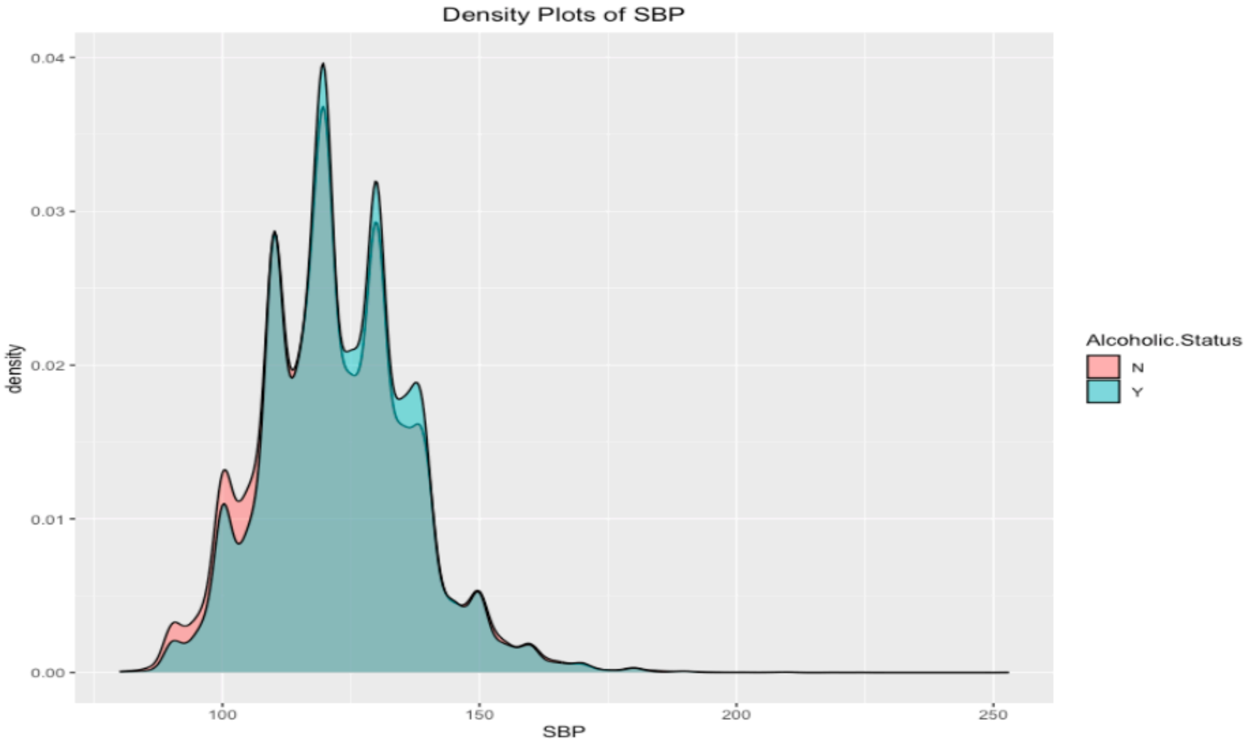
A. Numerical Variables

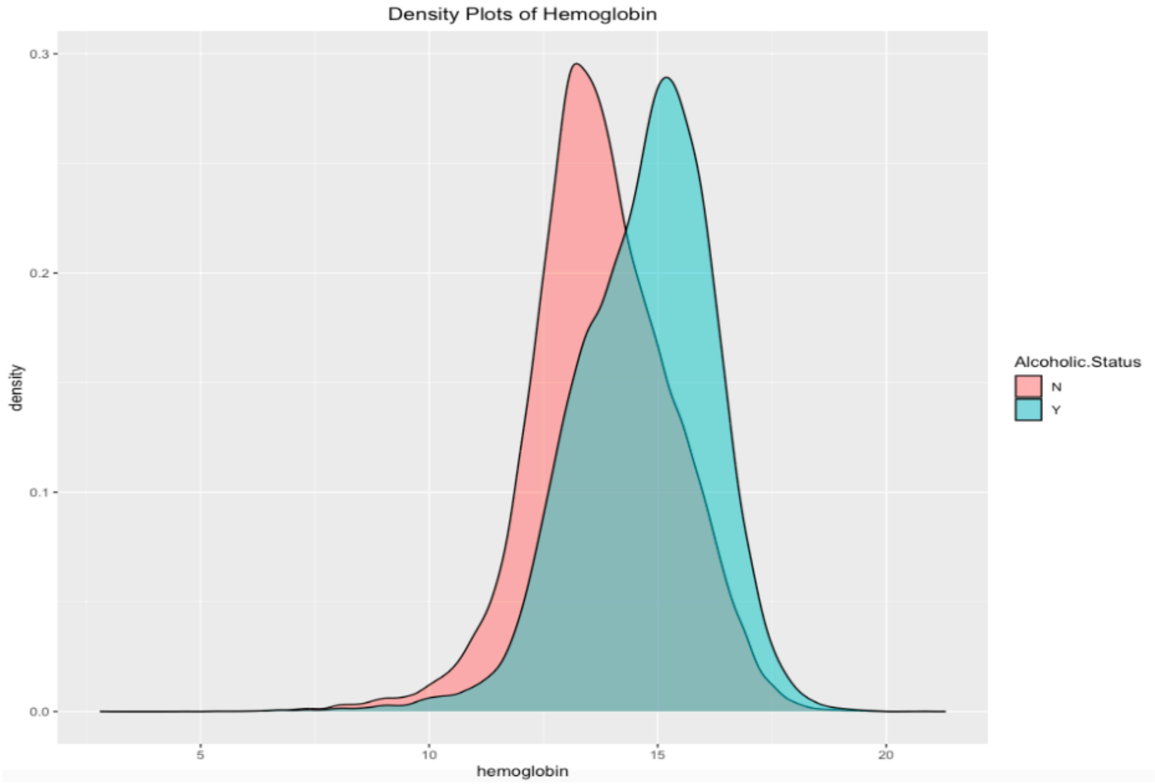
Within this dataset, the numerical predictors included age, height, weight, waist circumference (waistline), visual acuity in the left eye (sight_left), visual acuity in the right eye (sight_right), systolic blood pressure (SBP), diastolic blood pressure (DBP), fasting blood glucose (BLDS), total cholesterol concentration (tot_chole), HDL cholesterol (HDL_chole), LDL cholesterol (LDL_chole), triglyceride, hemoglobin, serum creatinine concentration (serum_creatine), SGOT AST, SGOT ALT , gamma-GTP, and BMI. To start, we calculated the summary statistics of each variable as well as the number of NAs in the original dataset (before imputation). The summary statistics of the first six numerical variables as listed above are displayed in the table below (summary statistics were calculated for all numerical variables but only some will be shown for brevity).



	Min	1st Quartile	Median	Mean	3rd Quartile	Max	NAs
age	20	35	50	47.68	60	85	4877
height	135	155	160	162.21	170	190	4941
weight	30	55	60	63.23	70	135	4972
waistline	35	74.5	81	81.27	87.6	999	4940
sight_left	0.1	0.7	1	0.98	1.2	9.9	4877
sight_right	0.1	0.7	1	0.98	1.2	9.9	4900

Additionally, we created density plots for each of the numerical variables in order to visualize the relationship between these variables and Alcoholic Status. Two example density plots are shown below. The first density plot is that of SBP, which was one of the predictors we later dropped from the model. From the density plot we can see that there is little separation between an alcoholic status of yes and an alcoholic status of no, which indicates that this variable will likely be unhelpful in classification. The second density plot is that of hemoglobin, which is one of the predictors we kept in our final model. This density plot displays a much clearer separation, which indicates that this variable may be more helpful in classification.



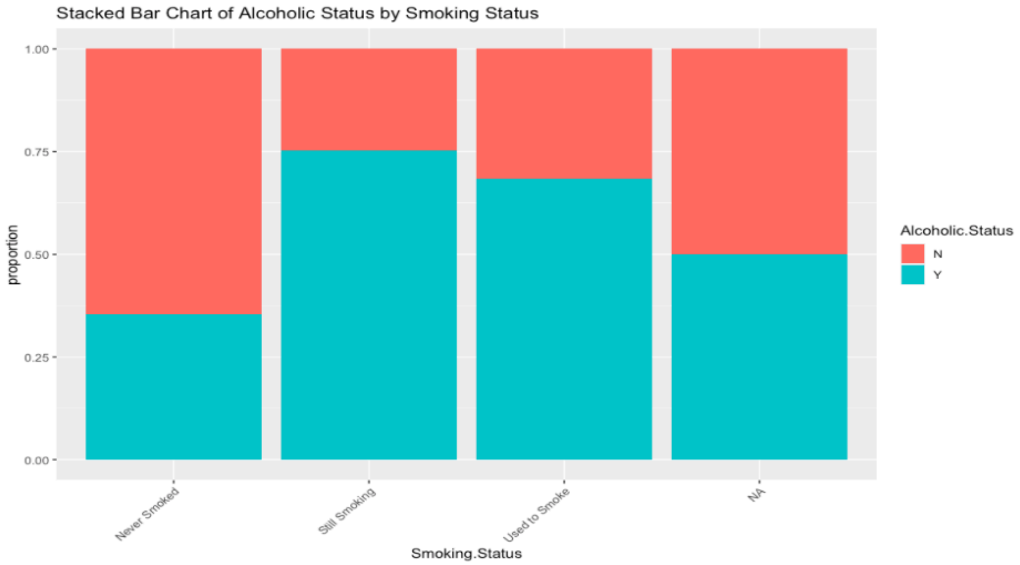


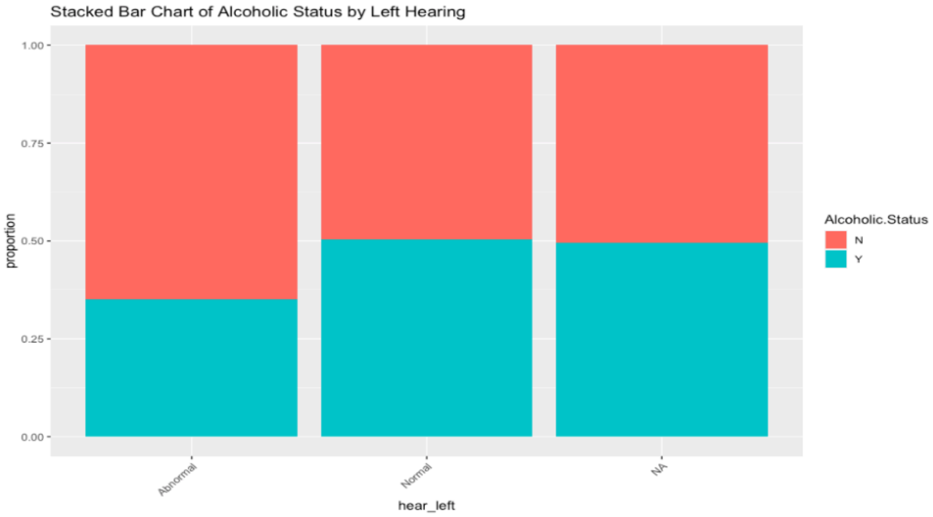
B. Categorical Variables

The categorical variables in this dataset are sex, left ear hearing (hear_left), right ear hearing (hear_right), urine protein, BMI category, age category, and smoking status. Since we could not calculate the summary statistics of these variables, we instead determined the number of observations within each category as well as how many missing values each variable contained in the initial dataset. The total missing values for each of the categorical variables can be seen in the table below.

	sex	hear_left	hear_right	urine_protein	BMI.Category	Age.Category	Smoking Status
NAs	4962	4833	4877	4899.00	4874	8313	4879

Additionally, we made stacked bar charts for each categorical variable in order to visualize the proportion of alcoholics and non-alcoholics within each category of each variable. Two of such plots are shown below. The first plot is of the proportion of alcoholics and non-alcoholics split by smoking status. The proportion of alcoholics and non-alcoholics is different across the different levels of smoking status, which indicates that this may be a useful predictor. We ended up keeping smoking status in our final model. The second set of stacked bar charts are those for left side hearing, which was not a variable we ended up keeping in our model. While there is some difference between the proportion of alcoholics with normal hearing and abnormal hearing, there is not as large of a difference as we saw in the bar charts for smoking status.

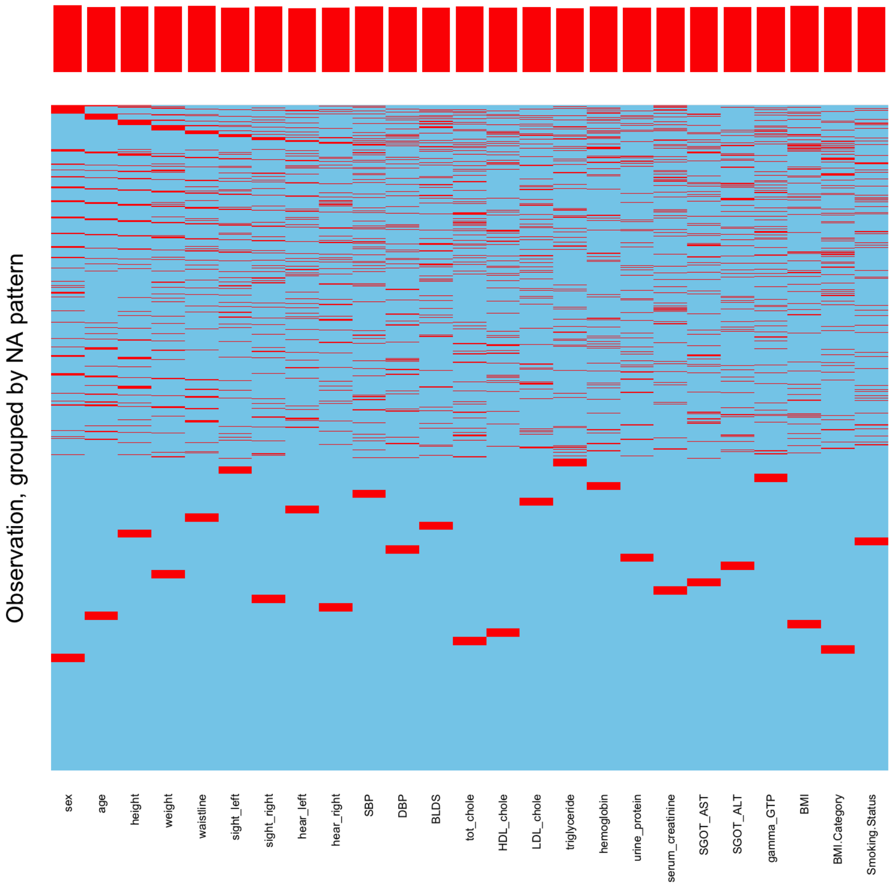




4. MISSING VALUE IMPUTATION

In both the training and testing datasets, about 7% of values were missing for each predictor.[1] Below is a plot of the combination of missing values for each observation, where missing values are represented in red. More common combinations of missing values are towards the bottom of the plot.

[1] Except for “AGE.Category”, which was redundant due to the “age” variable and ignored.



Nearly half of the observations contain no missing values or 1 missing value. Additionally, we see no apparent pattern in the distribution of missing data.

We assumed that the missing data was missing completely at random (MCAR) and thus felt comfortable imputing missing values. We tried multiple imputation methods, including those found in R packages “mice” and “Amelia”. Ultimately, the most effective method for missing data imputation we found was a simple iterative method in which missing values were estimated using the other predictors in the observation. The algorithm is as follows:

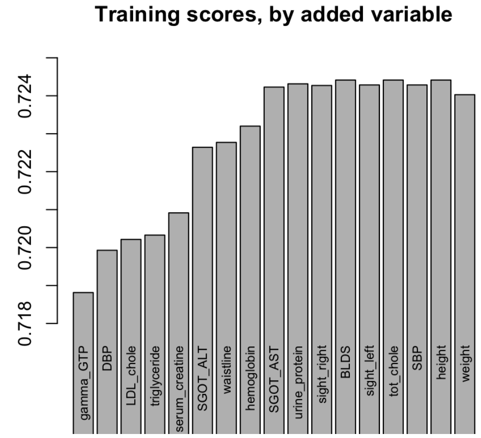
1. Combine the training and testing predictor data, and estimate the missing values by predictor median (for numerical predictors) or predictor mode (for categorical predictors).
2. Re-estimate missing values using all other (selected) predictors in the observation, by a linear model (for numeral predictors) or a logistic/multinomial model (for categorical predictors).
3. Repeat (2) until the estimates are relatively stable.

We ran the above algorithm for 3 iterations. Running the algorithm for more than 3 iterations did not show any improvement in the predictive ability of the estimated dataset. It should be noted that categorical predictor urine_protein was estimated solely by mode, as the vast majority of observations were valued at the mode.

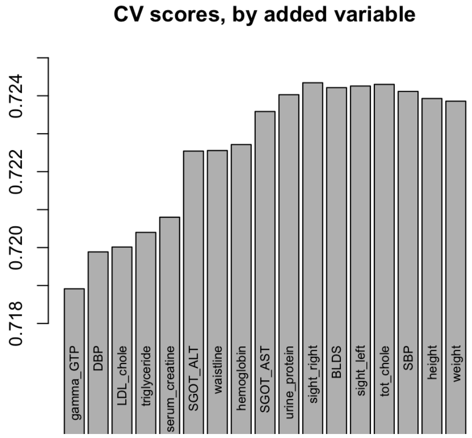
5. VARIABLE SELECTION

The predictors "BMI", "BMI.Category", and "AGE.Category" were dropped due to redundancy. The rest of the predictors were selected based on a forwards stepwise logistic regression. A logistic regression was used due to its fast speed and decent predictive power. The criterion used in the stepwise regression was the "accuracy score", defined as 1 minus the misclassification rate. We thereby obtained a collection of several logistic models, each adding one additional predictor as determined by the forwards stepwise regression.

We then examined the training accuracy scores and 10-fold cross-validational accuracy scores of each model. The first 4 added variables – sex, age, smoking status, and HDL cholesterol – quickly increased the training and cross-validational scores to near 0.72. The scores of the rest of the models are given below.



Training scores, by added variable



CV scores, by added variable

We see no significant improvement in training scores and cross-validation scores for models with extra predictors past "urine_protein". We therefore dropped from consideration the predictors sight_right, BLDS, sight_left, tot_chole, SBP, height, and weight. We found it interesting that height and weight were the very last predictors added by the stepwise regression, and actually appeared to negatively affect the model.

Thus, the final predictors used in our models were: sex, age, waist, DBP, BLDS, HDL_chole, LDL_chole, triglyceride, hemoglobin, urine_protein, serum_creatine, SGOT_AST, SGOT_ALT, gamma_GTP, and Smoking.Status.

6. LESS EFFECTIVE MODELS

Before ultimately landing on our best model, there were numerous models we attempted that were able to get relatively high accuracy, but were mostly unable to reach a 73% accuracy level. These models included Logistic Regression, KNN, LDA, Ridge Regression for Classification, and Random Forest. Logistic regression is a statistical tool that can be used to predict the probability of an event, e.g., whether or not someone is an alcoholic. While logistic regression is a simple and efficient technique that can produce highly accurate results on simpler datasets, its major drawback is that it assumes a linear relationship between the predictor(s) and the log odds of the response probability. Overall, logistic regression is a good approach that has difficulties predicting complex relationships, so it can be outperformed by other models such as neural networks, as shown by the maximum accuracy rate of 0.72243 that we obtained using logistic regression. Additionally, we tried K-Nearest Neighbors (KNN), another model that is very easy to implement; however, it struggles with high dimensionality and large datasets, so we were only able to achieve a maximum accuracy below 70%, using k=25. Linear Discriminant Analysis (LDA) was another model attempted that has relatively similar drawbacks to logistic regression. LDA performed better than KNN but worse than other models when assessed by 10-fold cross validation. We also attempted a Random Forest model to combat overfitting, but again, the results we obtained were not as accurate. Finally, we tried Ridge Regression for Classification. Ridge regression is a useful type of shrinkage method that shrinks regression coefficients towards zero using the tuning parameter lambda in order to reduce prediction MSE. Out of these models, Ridge Regression for Classification had the highest accuracy with that of 0.72333.

A. XGBoost

XGBoost or Extreme Gradient Boosting is a highly popular gradient-boosted decision tree algorithm. For our project, we used XGBoost using the train() function from the R library “caret”. We experimented with different amounts of predictors (all predictors, all significant predictors, and only the most significant predictors). Additionally, we took advantage of the wide assortment of inputs into the train() function and tried different cross validation, bootstrapping, tree building methods, and desired metric for model improvement (i.e. Accuracy so that the model may improve based on accuracy level of different versions). While XGBoost is known for its high accuracy, this is often linked with high levels of overfitting as XGBoost is a very powerful algorithm. To combat this, we tested different tuning parameters (eta, gamma, min_child_weight, nrounds, max_depth, colsample_bytree, and subsample) in order to decrease overfitting while still maintaining a high accuracy. At first, we arbitrarily tried different tuning grids based on our knowledge of what values most strongly influenced overfitting, then we tuned more strategically by creating multiple models where just one or two tuning parameters were changed, then we selected the best sets of tuning parameters from these models using the bestTune output included in XGBoost models. Overall, our best XGBoost model produced an accuracy of 0.73036, which was just slightly below that of our model from neural networks.

B. Neural Network

Neural networks are a type of machine learning that draw inspiration from the activity within the human nervous system. There are numerous types of neural networks, but all types of neural networks consist of an input layer, a hidden layer or layers, and an output layer. The raw input is fed to the input layer, the hidden layers or layers process this input, and the result is produced by the output layer. The primary strength of neural networks stems from the fact that they use parallel processing, and this powerful method can ultimately find hidden patterns, even from highly complex data. Since we ended up using a neural network as our hero model, more details regarding the different parameter inputs and our process for developing our final model are detailed in the section below.

7. HERO MODEL

Our hero model was the nnet() function in the R package nnet. The parameters of our hero model were: size = 20, maxit = 4000, reltol = 1e-14, abstol = 1e-14, entropy = T, decay = 1e-2. The size refers to the number of nodes in the single hidden layer. Literature describes the optimal number of hidden nodes to be around 2/3rds of the sum of input and output nodes, which in our case, is 28, so 20 nodes fits the theoretical value almost exactly. Through iterative testing, we determined 20 to be the best size as well practically. The training can run for a maximum of 4000 iterations, but in the final model, it converged rather quickly, using only 1510. The tolerances describe how much each iteration converges to consider an iteration worthwhile. These were set low because our model was not dependent on taking a low time to train, so we had the luxury of training the model to almost complete convergence. Setting entropy to true, in this function, means the loss function for the neural network is similar to the loss function of logistic regression, which is ideal for our purposes of classification. Finally, the weight decay argument means that the initial weights that the model uses in early iterations decay over time, so new, stronger connections could be forged. We found that increasing the decay helped the accuracy of the model, however, increasing it too much led to over-fitting on the training data. A decay value of .01 was found to lead to the highest testing accuracy.

We explored using another function in the caret package, pcaNNet, in an attempt to lower the complexity of the model and lower training time without losing much data from the training set. Using a variety of thresholds for the principal component analysis, 0.9, 0.95, and 0.99, we were unable to match the accuracy given through the pure nnet() function after using the variable selection techniques outlined above.

8. CONCLUSION

As we bring this Kaggle project to a close, it's essential to reflect on the journey we embarked on to predict alcoholic status using classification models. This report encapsulates the comprehensive process of building our final model, exploring various elements such as data imputation, classification methods, and, ultimately, our hero model, which employed neural networks. Throughout this project, we navigated challenges, analyzed data, and aimed for predictive accuracy.

Our hero model, constructed with the nnet() function from the nnet package, stands out as the pinnacle of our efforts. Incorporating predictors such as sex, age, various vitals, and lifestyle factors, this neural network achieved a commendable final accuracy score of 0.73120, securing the 23rd position in the Kaggle competition. This success reinforces the effectiveness of utilizing advanced machine learning techniques, specifically neural networks, in predicting complex outcomes like alcoholic status.

Alcohol, often symbolized as a social bonding element, comes with significant health risks when consumed excessively. With approximately 107 million people globally struggling with alcohol use disorder, our project sought to address diagnostic limitations by leveraging vital signs for prediction. The dataset from the National Health Insurance Service in Korea, comprising 26 predictors, served as the foundation for our exploration.

Before delving into model fitting, a meticulous analysis of numerical and categorical variables was conducted. For numerical predictors, summary statistics and density plots were employed to visualize relationships with alcoholic status. While some variables were dropped due to limited discriminatory power, others like hemoglobin proved pivotal for classification. Categorical variables, including sex, hearing status, BMI category, and smoking status, were scrutinized through observation counts and stacked bar charts, aiding in predictor selection.

Our hero model's success can be attributed to the carefully chosen parameters: a hidden layer with 20 nodes, 4000 iterations capped for training, and stringent convergence tolerances. The use of maximum conditional likelihood fitting, along with a decay factor for weight adjustments, optimized the neural network for classification purposes.

A noteworthy attempt to simplify the model involved exploring the pcaNNet function from the caret package, incorporating principal component analysis (PCA). Despite experimenting with various thresholds, the accuracy achieved fell short of the hero model. Hence, the decision was made to retain all relevant variables, striking a balance between model complexity and predictive power.

In conclusion, this Kaggle project has been a testament to the potential of advanced machine learning methods in predictive analytics. The successful implementation of a neural network to predict alcoholic status underscores the significance of considering an individual's vitals in diagnostics. As we wrap up this project, the lessons learned, challenges overcome, and insights gained will undoubtedly inform future endeavors in utilizing classification models for predictive healthcare analytics. This project serves as a reminder of the continuous evolution and innovation within the realm of data science, promising exciting possibilities for the future.

9. ACKNOWLEDGEMENT

We would like to express our great gratitude to Professor Akram Almohalwas and TA Kyle McEvoy for teaching us Stats 101C this Fall and all the help, suggestions and encouragement.

---

10. REFERENCES

"Alcohol Use Disorder." *Mayo Clinic*, Mayo Foundation for Medical Education and Research, 18 May 2022, https://www.mayoclinic.org/diseases-conditions/alcohol-use-disorder/symptoms-causes/syc-20369243

Awan, Abid Ali. "Building Neural Network (NN) Models in R." *DataCamp*, 6 Feb. 2023, www.datacamp.com/tutorial/neural-network-models-r.

Ritchie, Hannah, and Max Roser. "Alcohol Consumption." Our World in Data, Apr. 2018, https://ourworldindata.org/alcohol-consumption

"What Is XGBoost?" *NVIDIA Data Science Glossary*, NVIDIA, 2023, www.nvidia.com/en-us/glossary/data-science/xgboost/.