ECS 111 Final Project Report

Ruhi Aggarwal, Siya Jain, Tisha Kathrani

Introduction

We are applying a research re-implementation on the Smoker Status Prediction using Bio-Signals dataset. There are 22 features (age, height (cm), weight (kg), waist (cm), eyesight (left), eyesight(right), hearing(left), hearing(right), systolic, relaxation, fasting blood sugar, cholesterol, triglyceride, HDL, LDL, hemoglobin, urine protein, serum creatinine, AST, ALT, Gtp, dental caries) of their bio-data that we are analyzing to see how they can be used to predict smoker status where 0 is non-smoker and 1 is smoker. We found this dataset from a Kaggle competition where the submissions we saw all implemented classification models and created an output for the probability of the individual being a smoker or not. We want to build the best possible classification prediction model for the variables given to predict whether an individual is past the threshold for being in danger due to smoking. We want to pick the top 8-10 variables that show predictiveness of smoker status. This project could be used to assess one's health. For example, if someone is a non-smoker but they are classified as a smoker, it could be a signal that their bio-signals are at unhealthy levels and they may need to get their health checked.

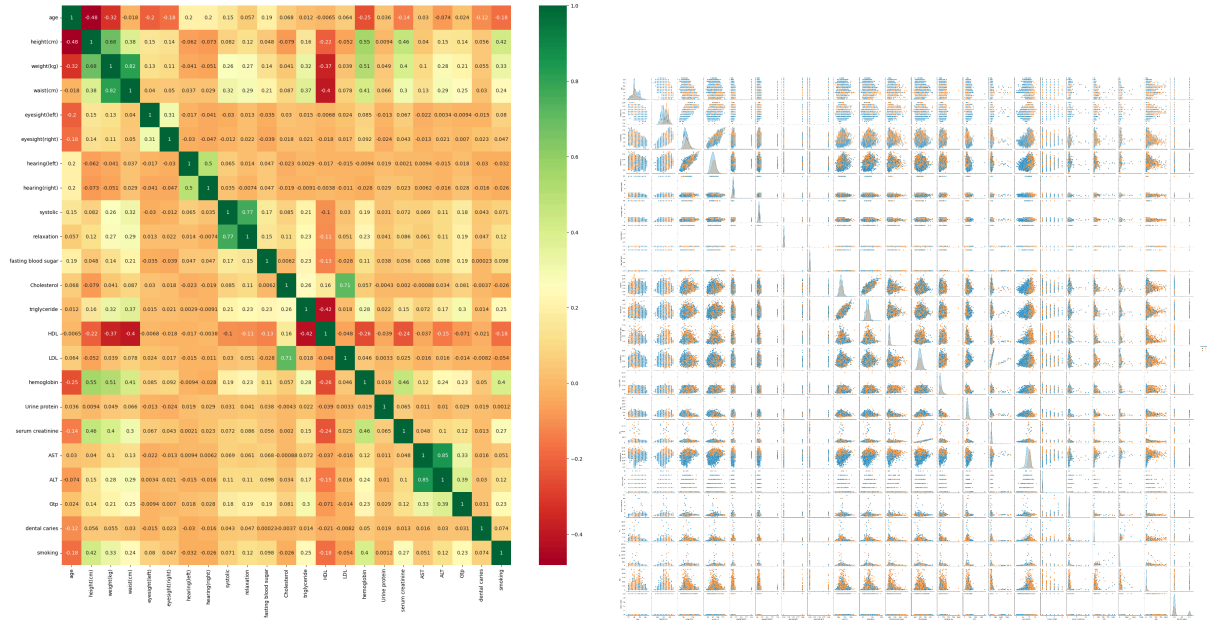
Background

We used various python libraries with important machine learning tools to complete our project including Pandas, NumPy, Matplotlib, Seaborn, Scikit-learn, Keras, and PyTorch, which are crucial for data manipulation, numerical operations, visualization, and machine learning model development. Classification, a type of supervised learning, is used to predict categorical class labels, such as smoker or non-smoker, based on past observations. Training and testing models involve splitting the data into subsets to train the model and evaluate its performance on

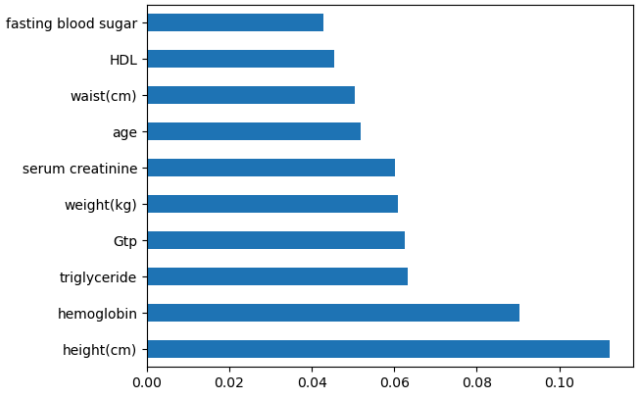
unseen data. Feature selection is the process of identifying the most predictive variables for the target variable, which in this case is smoker status. Familiarity with machine learning algorithms such as Logistic Regression, Neural Networks, K-Nearest Neighbors (KNN), Gradient Boosting, and Random Forest Classifier is essential, as these are used to build and evaluate the classification models in the project. Understanding these concepts will enable readers to grasp the methodologies and approaches used to predict smoker status using bio-signals.

Methodology

We began by doing exploratory analysis to understand the dataset and any possible underlying relationships between the variables. Since the dataset is quite large with 22 different features, we realized that we may need to select only a subset of the features for the final model to reduce complexity. We decided to try different visualization techniques to see the relationships between the features and also with the “smoking” output variable. We started with a correlation matrix or a heatmap which showed the correlation factor between every pair of variables possible. We saw that smoking didn’t have a strong correlation with any one of the variables as we saw in the heatmap, so we couldn’t make a decision based on this graph. We decided to try the pairwise plots instead between all the variables and then categorizing the color of each data point based on smoker status, so we could see which combination of variables showed a clear distinction between both smoker status groups.



From this, we listed down some clear predictive variables where the smoker and non-smoker points were clearly distinguishable. Using this method, we identified hemoglobin, weight, waist, HDL, age, and triglyceride to be potential predictors. We also used a feature importance function to see which top features have the best correlation with the output variable, smoking. This affirmed our conclusions about including hemoglobin, triglyceride, waist, weight, HDL, and age from our manual selection previously, and also influenced our decision to add height, relaxation, systolic, and serum creatine



Afterwards, we started working on building our logistic regression model. Since the test data provided from Kaggle didn't have the output variable we needed ("smoking"), we used the training data and split it into a training and validation set for creating a model and then evaluating the model performance. We tested the model on both the full set of features (left) as well as the subset of features (right). They achieved an accuracy of 0.73 and 0.67, respectively.

```
Logistic Regression Results with all features:
Accuracy: 0.73
Classification Report:
              precision    recall  f1-score   support

           0       0.77      0.81      0.79      4975
           1       0.63      0.57      0.60      2822

    accuracy                           0.73      7797
   macro avg       0.70      0.69      0.70      7797
weighted avg       0.72      0.73      0.72      7797

Confusion Matrix:
[[4040  935]
 [1200 1622]]
```

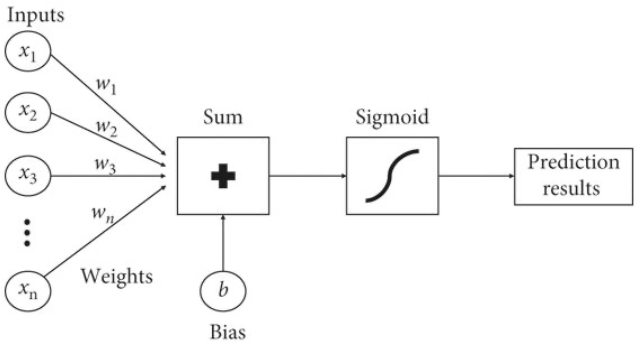
```
Logistic Regression Results with hand-picked features:
Accuracy: 0.67
Classification Report:
              precision    recall  f1-score   support

           0       0.70      0.84      0.76      4975
           1       0.56      0.37      0.45      2822

    accuracy                           0.67      7797
   macro avg       0.63      0.60      0.60      7797
weighted avg       0.65      0.67      0.65      7797

Confusion Matrix:
[[4159  816]
 [1775 1047]]
```

This diagram shows a neural network diagram that displays how our logistic regression model works. There are multiple inputs which represent the features that we included, with individual weights. Then the inputs are added up and generate an output from the sigmoid activation function. We thought our logistic regression model had a decent model performance, but that we should explore further types of models, especially other neural networks.



Next, we tested a K-nearest neighbors classifier. We tested it on both the subset of the features (right) as well as all the features (left) and got a pretty decent result. The accuracy was roughly 0.71 for both.

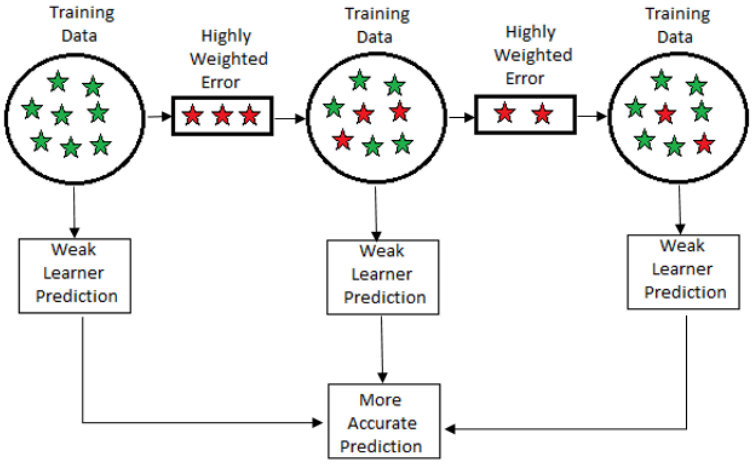
	precision	recall	f1-score	support
0	0.73	0.70	0.71	2953
1	0.69	0.72	0.71	2775
accuracy			0.71	5728
macro avg	0.71	0.71	0.71	5728
weighted avg	0.71	0.71	0.71	5728

```
0.7114175977653632
[[0.70199797 0.29800203]
 [0.27855856 0.72144144]]
```

	precision	recall	f1-score	support
0	0.72	0.70	0.71	2953
1	0.69	0.72	0.70	2775
accuracy			0.71	5728
macro avg	0.71	0.71	0.71	5728
weighted avg	0.71	0.71	0.71	5728

```
0.7075768156424581
[[0.69928886 0.30071114]
 [0.2836036  0.7163964 ]]
```

We also experimented with gradient boosting. It is a special type of ensemble learning technique which combines several weak learners into a strong learner. With gradient boosting, each predictor improves on the previous one by minimizing the residuals of the training point.



This diagram explains the process of how gradient boosting works in more detail. As seen in the diagram, each individual weak predictor builds off the previous to form one stronger predictor. The results of our testing with gradient boosting are shown below.

	precision	recall	f1-score	support
0	0.84	0.71	0.77	2953
1	0.73	0.86	0.79	2775
accuracy			0.78	5728
macro avg	0.79	0.78	0.78	5728
weighted avg	0.79	0.78	0.78	5728

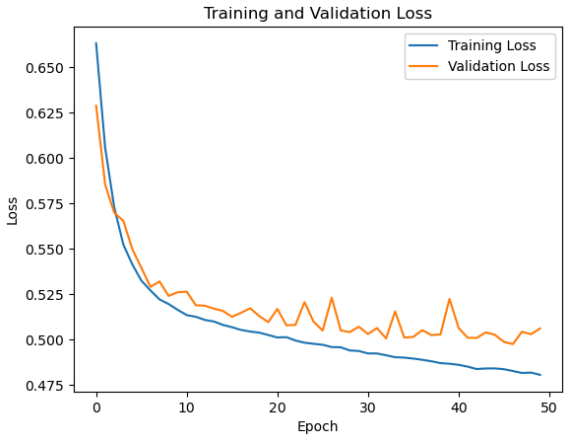
```
0.78125
[[0.70572299 0.29427701]
 [0.13837838 0.86162162]]
```

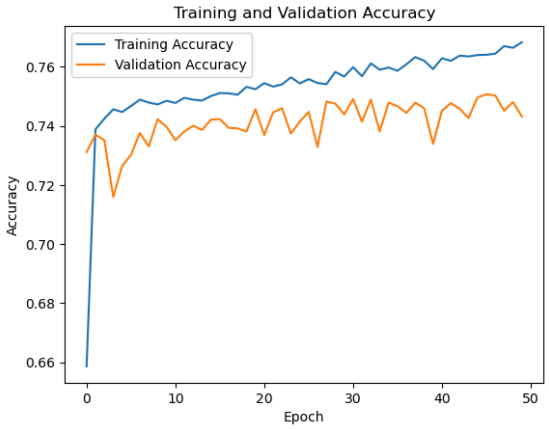
	precision	recall	f1-score	support
0	0.83	0.69	0.76	2953
1	0.72	0.85	0.78	2775
accuracy			0.77	5728
macro avg	0.78	0.77	0.77	5728
weighted avg	0.78	0.77	0.77	5728

```
0.7685055865921788
[[0.69420928 0.30579072]
 [0.15243243 0.84756757]]
```

On the left, we were able to achieve an accuracy of 0.78 using all the features, and on the right, we achieved an accuracy of 0.77 with the subset of features. This method clearly achieved higher accuracies than logistic regression and KNN, so we decided to further explore trees and random forests.

We tried to use a Neural Network on our model as well. We tried different models on our data as well as different subsets of data (full data and hand-picked). We tried a different number of hidden layers with an output layer of a sigmoid activation function, and we finalized on using a glorot distribution initializer for our output layer and normal distribution initializers for the hidden layers. At first we tried performing 10 epochs with a batch size of 32 and then a batch size of 16, but then we decided on doing 50 epochs on a batch size of 32 since we had such a large dataset. This gave us the highest accuracy out of the different neural networks we tried of 0.743 with a loss of 0.506.





Our last model that we tried was a random forest classifier. We tried it on the full dataset as well as the features that we selected on our own. The model that was trained on the full dataset ended up giving us the highest accuracy out of all our models with an accuracy of 0.788. The model trained on the dataset with the features we selected ourselves had a lower accuracy of 0.771.

Full Dataset Random Tree Classifier Metrics:

Accuracy: 0.7836941340782123

Classification Report:

	precision	recall	f1-score	support
0	0.84	0.71	0.77	2953
1	0.74	0.86	0.79	2775
accuracy			0.78	5728
macro avg	0.79	0.79	0.78	5728
weighted avg	0.79	0.78	0.78	5728

Confusion Matrix:

```
[[2100  853]
 [ 386 2389]]
```

Hand-Picked Dataset Random Tree Classifier Metrics:

Accuracy: 0.7707751396648045

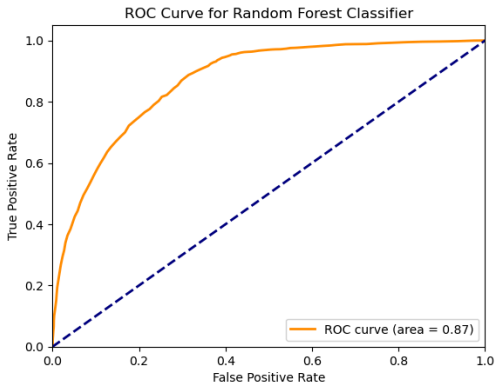
Classification Report:

	precision	recall	f1-score	support
0	0.84	0.69	0.76	2953
1	0.72	0.86	0.78	2775
accuracy			0.77	5728
macro avg	0.78	0.77	0.77	5728
weighted avg	0.78	0.77	0.77	5728

Confusion Matrix:

```
[[2034  919]
 [ 394 2381]]
```

We attempted to use Sequential Feature Selector previously to find the most relevant features for our model so we could try training different types of model on these automated selected features. However, possibly due to the large size of our dataset, SFS took a very long time to run and would end up just crashing on our computers. Since we had no clear way of automating the best features for building a model, we tried to use our Random Forest Classifier model to select the most relevant features for model predicting from which we got the top 18 features. We tried training a model on these 18 features, from which we got a similar accuracy to the full dataset of 0.785.



Automated Features Dataset Random Tree Classifier Metrics:

Accuracy: 0.7798533519553073

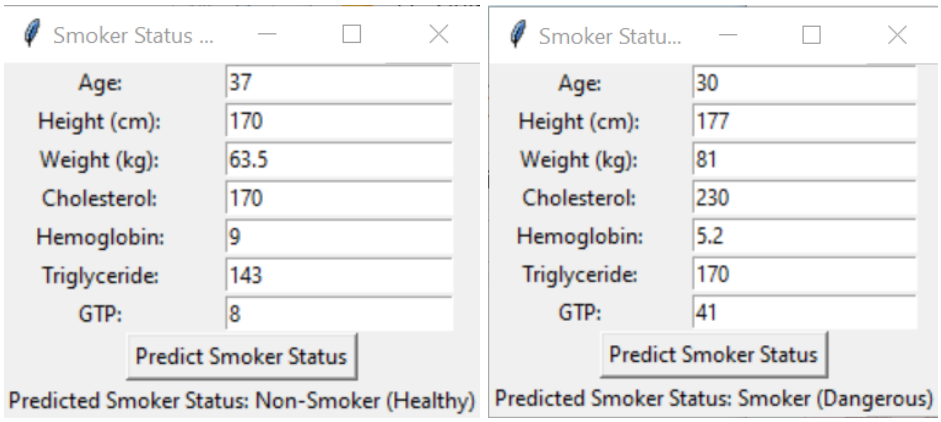
Classification Report:

	precision	recall	f1-score	support
0	0.84	0.71	0.77	2953
1	0.73	0.85	0.79	2775
accuracy			0.78	5728
macro avg	0.79	0.78	0.78	5728
weighted avg	0.79	0.78	0.78	5728

Results

We were able to build multiple models that classified our data with relatively high accuracies. Our highest accuracy was 0.788 in our random forest classifier model trained on the full dataset. Overall, we were successful in achieving our project goals as we can make accurate

predictions of the smoker status of the individual. We created a pop-up demo that takes in some of the bio-data signals of users and gives them a prediction of whether they are at a healthy or dangerous level based on their smoking habits. Here are two examples of different bio-datas from two individuals where our interface was accurately able to predict their health levels.



Discussion

Splitting the data into training and validation sets allowed us to assess model performance accurately. Metrics such as precision and recall provided insights into the trade-offs between correctly identifying smokers and minimizing false positives. Working on this project has been a valuable learning experience in several aspects of machine learning and data analysis. Initially, delving into the dataset and performing exploratory data analysis (EDA) helped us understand the nature of the features and their relationship with the target variable, smoker status. Through EDA, we identified key features that showed promise in predicting smoker status, such as hemoglobin, weight, waist, HDL, height, age, relaxation, systolic, triglyceride, and serum creatine. We were able to learn how to create different models and implement them on a large dataset.

However we ran into some challenges like not being able to automate a subset of features by which ones were the most optimal for predicting smoker status as SFS didn't work for us. Alongside, we also struggled with the amount of time that hyperparameter tuning took especially for neural networks since there are many different combinations of layers, initializers, and batch sizes/epochs that we could have tried. Overall, it is hard to pick just a few features from a dataset of 22 features that would give the highest accuracy for our model. If we want to implement these models in the real world, it is impractical to have to use 22 different features for each individual since a lot of these bio-datas people don't have their information for. We would have to find a

good balance between model accuracy and the performance with the number of features we include.

Conclusion

In conclusion, our project focused on re-implementing research on the Smoker Status Prediction using Bio-Signals dataset, with the aim of building an effective classification model to predict an individual's smoker status based on various bio-signals. We began with exploratory data analysis to understand the dataset and identify potential predictive features. Our analysis led to the selection of a subset of features, including hemoglobin, weight, waist, HDL, age, and triglyceride, which showed promise in distinguishing between smoker and non-smoker groups.

We experimented with several machine learning models, including logistic regression, K-nearest neighbors, gradient boosting, and random forest classifiers. Through model evaluation and validation, we found that the random forest classifier performed the best, achieving the highest accuracy on the full dataset. Our project highlights the importance of feature selection, model evaluation, and the use of various machine learning techniques in predicting smoker status. This work has potential applications in healthcare, where such predictions could be used to assess an individual's health risk related to smoking and guide preventive measures. Overall, this project deepened our understanding of machine learning and data analysis in the context of health-related classification tasks