WARFARIN DOSING

Introduction

Blood clots are semi-solid or gel like masses made up of platelets and fibrin (a blood protein), they form in the arteries and veins to help control bleeding, however they can also cause serious medical issues.

One complication of blood clots is deep vein thrombosis (DVT) is when a blood clot also known as a thrombus form in one or more of the deep veins in the body, usually in the legs. This can cause swelling or leg pain. This condition affects about 200,000 people each year. A complication of DVT is pulmonary embolism. Pulmonary embolism is caused by blood clot (which travels from the legs) blocking the blood flow in one or more arteries in the lungs. A pulmonary embolism can be life-threatening without medical intervention.

Another serious medical complication of blood clots is a heart attack, which is also known as a myocardial infarction, and it occurs when the blood flow to the heart is blocked. Lack of blood flow to the heart can damage or destroy part of the heart muscle. Blood clots can also narrow one or more of the arteries leading to the brain thereby causing a stroke. Strokes are the 5th leading cause of death in US men

Warfarin is an FDA approved prescription-only medication that serves as an oral anticoagulant commonly used to treat and prevent blood clots. Warfarin, a blood thinner, works by competitively inhibiting an essential enzyme required for vitamin K activity, (vitamin K is essential for blood clotting). Therefore, warfarin makes it so that it takes longer for blood to clot. Warfarin is now the most widely used anticoagulant in the world.

Individual Warfarin dosage varies from 0.5 mg/day to over 20 mg/day. There are several factors involved in deciding how much warfarin an individual is prescribed. These factors include concomitant medications, diet, alcohol intake, and VKORC1, vitamin K [ep]oxide reductase.

There is a problem with Warfarin dosage as patients starting warfarin therapy are most likely to overdose during the initial weeks of therapy. This is because determining the optimal dosage of Warfarin for individuals is challenging for clinicians due to its narrow therapeutic index and high inter- and intra-individual variability among patients in dose requirements. So, in present-day clinicians use the trial-and-error method with dosing, they start patients on a certain dose and require several follow-up treatments to adjust the dose. These follow-up treatments can be burdensome to the healthcare system and the patients, especially financially.

Our project aims to solve the Warfarin dosage problem by using data from over 5,000 patients to create a machine learning model that can predict the dose of Warfarin for any individual given their patient information.

Methods

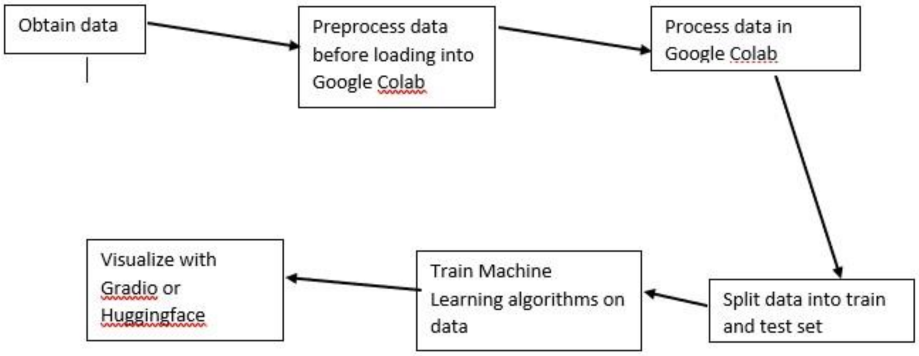
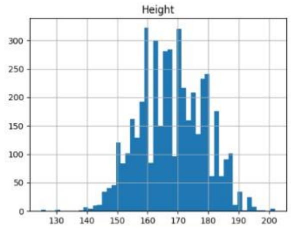

Obtain data
Preprocess data before loading into Google Colab
Process data in Google Colab
Split data into train and test set
Train Machine Learning algorithms on data
Visualize with Gradio or Huggingface

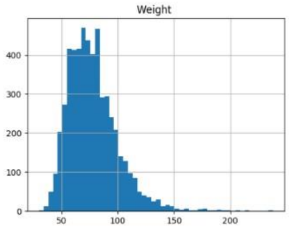
Image 1: Methods Workflow Flowchart

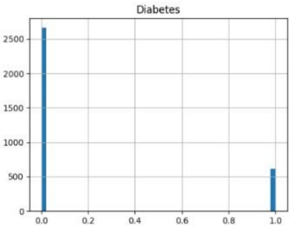
The data was obtained from the PharmGKB website under the International Warfarin Pharmacogenetics Consortium (IWPC) data set. The data was processed prior to uploading it into the Google Colab Notebook by selecting for only columns that are necessary for the analysis. These columns are Age, Race, Gender, Height, Weight, Diabetes, Simvastatin, Amiodarone, INR on Reported Therapeutic Dose of Warfarin, VKORC1 genotype and Therapeutic Dose of Warfarin (the target column).

We chose to treat the Warfarin dose problem as a supervised regression problem.

We visualized all the numerical columns with histograms and all the categorical columns with pie charts. These are some of the columns we visualized:







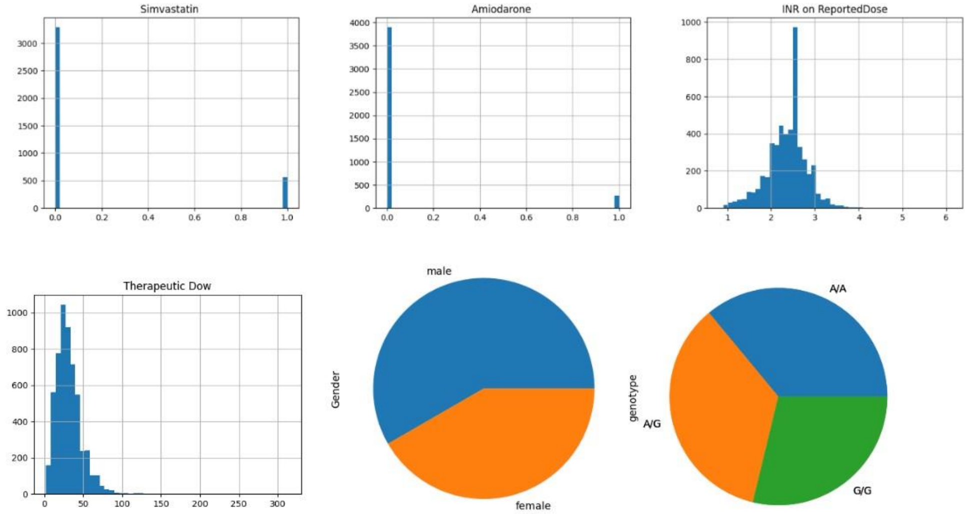


Image 2: Histograms and Pie Charts of columns in the dataset

We converted the categorical text columns to numerical columns using label encoder and then replaced missing values with their K-nearest neighbors using KNNImputer with K = 3

We checked for multicollinearity by using a heatmap to draw the correlation map of all columns. We set a threshold of 0.7 and found that no columns were highly correlated with each other. The image below shows our correlation map:

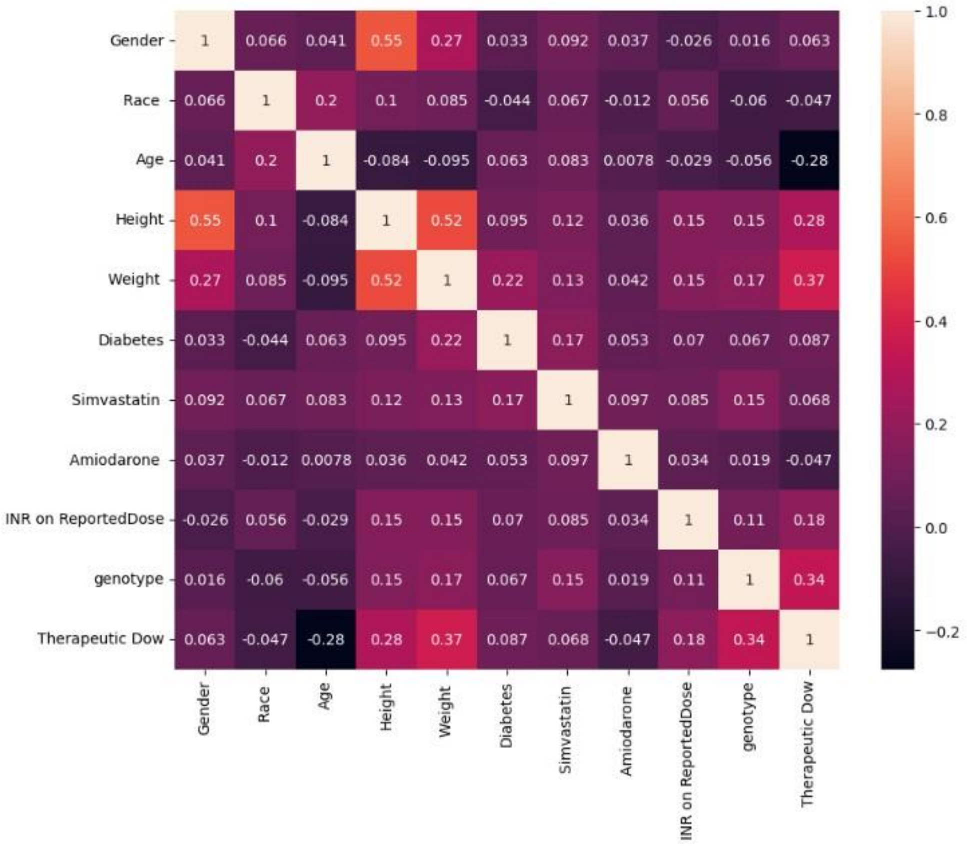


Image 3: Correlation Map

We scaled the inputs to the machine learning algorithms with the Standard Scaler. We then chose 6 Machine Learning algorithms: Decision Tree, Random Forest, KNN, Neural Network, Gradient Boost and AdaBoost. These algorithms are explained below:

The Decision Tree algorithm builds regression or classification models in the form of a tree structure. In our project we used the Decision Tree Regression model. It breaks down a dataset into smaller subsets while an associated decision tree is incrementally developed. The result is a tree with decision nodes and leaf nodes.

Random Forest is an ensemble of Decision Trees, it combines the output of multiple decision trees to reach a single result. The trees are trained on different parts of the training set with the goal of reducing variance, the mean or average prediction of the individual trees is returned

KNN or K-Nearest Neighbor is a non-parametric, supervised learning classifier, which uses proximity to make classifications or predictions about the grouping of an individual data point. In regression analysis the average the k nearest neighbors is taken to make a prediction about a classification.

Neural Networks are computing systems inspired by the biological neural networks that constitute animal brains. Neural networks are comprised of node layers, containing an input layer, one or more hidden layers, and an output layer. Each node, or artificial neuron, connects to another and has an associated weight and threshold. If the output of any individual node is above the specified threshold value, that node is activated, sending data to the next layer of the network. Otherwise, no data is passed along to the next layer of the network.

Gradient Boost is an ensemble method that works by sequentially adding predictors to an ensemble, each one correcting its predecessor. This method tries to fir the new predictor to the residual errors made by the previous predictor

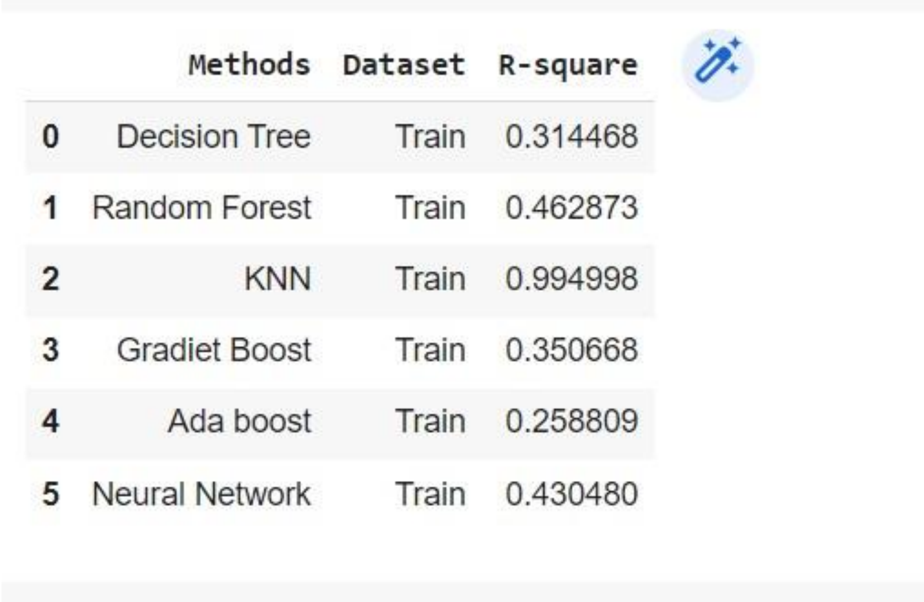
AdaBoost is an ensemble method that combines several weak learners into a strong learner. This method tweaks the instance weights at every iteration.

All our machine learning algorithms are evaluated by their mean square error and Rsquare.

Lastly, we used Gradio to launch our machine learning model.

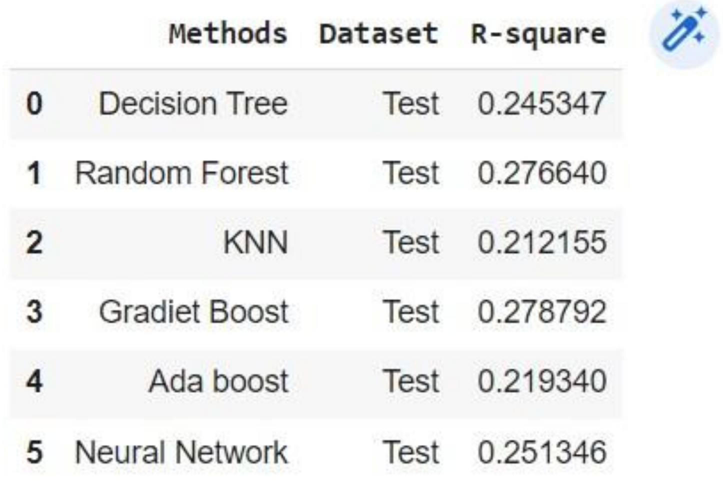
Results

Train R square metric results:



	Methods	Dataset	R-square
0	Decision Tree	Train	0.314468
1	Random Forest	Train	0.462873
2	KNN	Train	0.994998
3	Gradiet Boost	Train	0.350668
4	Ada boost	Train	0.258809
5	Neural Network	Train	0.430480

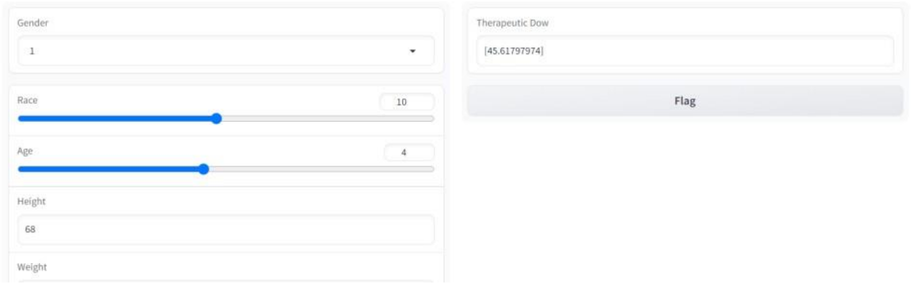
Test R-squared metric results:



	Methods	Dataset	R-square
0	Decision Tree	Test	0.245347
1	Random Forest	Test	0.276640
2	KNN	Test	0.212155
3	Gradiet Boost	Test	0.278792
4	Ada boost	Test	0.219340
5	Neural Network	Test	0.251346

Based on the R-squared values for the train and test datasets, we can draw the following conclusions:

- Random Forest and Gradient Boost models outperform other models on both train and test datasets, demonstrating that they have higher generalization ability and can avoid overfitting.
- The KNN model has an extremely high R-squared value for the train dataset but performs poorly on the test dataset. This shows that the KNN model is overfitting to the training data and may not generalize well to fresh data.
- The Neural Network model has a reasonable R-squared value for train and test datasets, indicating it has good generalization ability but may not be the ideal solution for this situation. So, to tried improving mlp neural n/w I did parameter tuning by looping layers and learning rate. And stored the best model.
- The R-squared values for Gradient Boost and AdaBoost for the training dataset are 0.350354 and 0.259983, respectively. R-squared values for the test dataset are 0.277077 and 0.224837, respectively.
- We can see from these results that Gradient Boost surpasses AdaBoost in terms of R-squared values for both the training and test datasets. However, the Rsquared values for Gradient Boost are still quite low, indicating that the model may not be capturing all the variability in the data. Further hyperparameter tuning or feature engineering may improve the performance of these models.
- We implemented the Gradio application for the algorithms to predict wafrin dosage.


Gender: 1

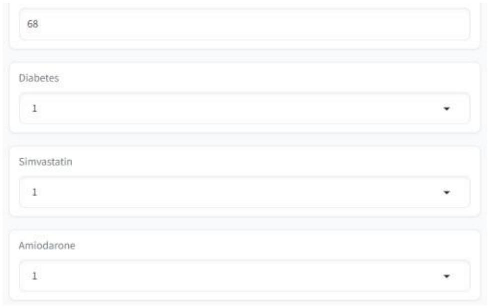
Race: 10

Age: 4

Height: 68

Weight

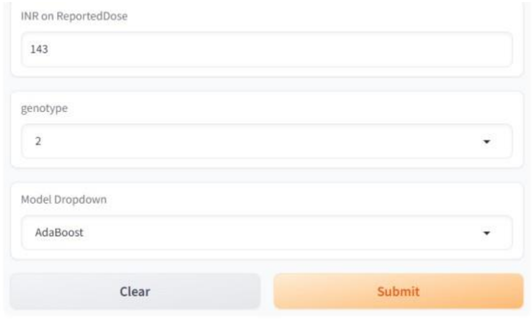
Therapeutic Dow: [45.61797974]


68

Diabetes: 1

Simvastatin: 1

Amiodarone: 1


INR on ReportedDose: 143

genotype: 2

Model Dropdown: AdaBoost

Course Summary

This course taught us the fundamental concepts, algorithms, and applications of machine learning. We learned supervised and unsupervised models, classification, and regression tasks. We also learned how to implement various common machine learning algorithms while receiving hands-on experience with programming and data analysis. Throughout the training, emphasis was made on the need for data preparation and feature engineering in generating accurate and useful findings. The course also looked at ethical issues in machine learning, such as prejudice and fairness. By the end of the course, we understood the fundamental ideas of machine learning and applied their knowledge to real-world challenges.

References

- https://my.clevelandclinic.org/health/diseases/17675-blood-clots
- Patel S, Singh R, Preuss CV, et al. Warfarin. [Updated 2023 Feb 25]. In: StatPearls [Internet]. Treasure Island (FL): StatPearls Publishing; 2023 Jan-. Available from: https://www.ncbi.nlm.nih.gov/books/NBK470313/
- Mayo Clinic
- Pirmohamed, Munir. "Warfarin: almost 60 years old and still causing problems." British journal of clinical pharmacology vol. 62,5 (2006): 509-11. doi:10.1111/j.13652125.2006.02806.x
- https://www.saedsayad.com/
- https://scikit-learn.org/
- https://en.wikipedia.org
- https://www.ibm.com/