Data Programming Project 2 Final Report

Team:

Subramanian Arumugam (TL)	sarumugam2@student.gsu.edu
Abinesh Ganesan	ag1@student.gsu.edu
Drona Charyulu Ronanki	dronanki1@student.gsu.edu
Shraddha Joshi	sjoshi22@student.gsu.edu
Tasleem Syed	tsyed1@student.gsu.edu

Objective:

To predict the survival rate of ICU patients based on their vital stats and medical history.

Motivation and Problem:

In a hospital, patients with serious health problems need intensive medical care and monitoring are admitted to an Intensive Care Unit (ICU). Patients are treated by a team of trained doctors, nurses, and other providers. The aim of the ICU is to support patients until the cause of their illness can be treated and resolved. When the number of ICU patients is low, it is relatively easy to manage the resources and equipment. However, as the number of ICU patients increases, it is most appropriate to allocate resources to the patients who have lower chances for survival. This problem is faced by ICU departments in many hospitals. We propose a solution to provide real-time prediction on patients' survival based on their demographic data, vitals, and blood test results. The hospital will be able to better prioritize their resources with our solution and in turn improve the survivability of their patients.

Solution:

The solution mainly consists of creating a predictive model that uses patients' demographic data, vitals, and blood test results to predict whether the patient would survive or not. Typically, ICU department collects these points for diseases related to cardiology, neurology, gastroenterology, gynecology, etc. The project will help ICU departments to improve the chances of survival of their patients by providing the right care and resources to the patients with high risk at the right time. This in-turn will help the hospital and doctors to build a good reputation and boost their efficiency in providing treatments.

The solution focusses on building a ML Pipeline by first collecting the right Dataset, second preparing the data by applying the necessary transformations and feature engineering techniques. Thirdly, 6 predictive modelling techniques were undertaken and trained on the data after sampling the dataset appropriately. Then the right model for the task was selected based on the **Recall** Metric.

Dataset Source:

Mitisha Agarwal. (2021). <i>Patient Survival Prediction</i> [Data set]. Kaggle.

https://doi.org/10.34740/KAGGLE/DSV/2972359

Methodology / Working:

The solution for this problem statement was worked out as explained in the below steps:

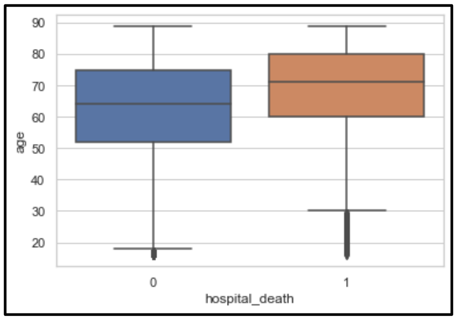
1. **Data Preparation**
   This dataset had 80 features and 91713 records of data. Of the 80 features, 25 were categorical and 55 were numerical. As explained in project 1, several features had the problems of missing values, outliers, varying scales, and varying distributions. We decided to convert the numerical data into categorical by binning them accordingly and later imputed the "NA" string to all missing values. We analyzed the distribution of each of these numeric features and binned these features appropriately to solve for the outlier and scaling issues. Also, the missing values in these features were converted to NA. We realized that this new category 'NA' would be a suitable addition since in the medical world, not all patients are treated the same. For example, a cardiology patient will

be treated differently for a patient with a neurology issue and the corresponding medical tests will be different for both these patients.

2. **Exploratory Data Analysis**

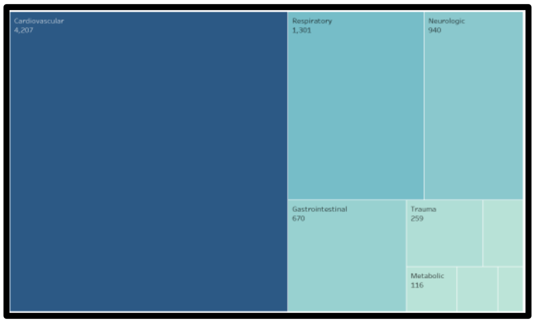
   The exploratory data analysis was discussed for the data was discussed in Projects 0 and 1. The following are some visual insights we would like to discuss in this report:

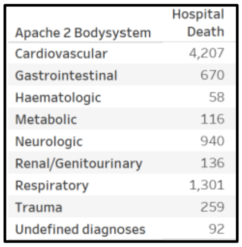
   a. Distribution of Age by the Target Variable (Hospital Death)



   We can see from the above chart that the mean age of those we pass away during ICU admission is 68.5 compared to those who survive at 61.7.

   b. Hospital Death by Disease Distribution





Apache 2 Bodysystem	Hospital Death
Cardiovascular	4,207
Gastrointestinal	670
Haematologic	58
Metabolic	116
Neurologic	940
Renal/Genitourinary	136
Respiratory	1,301
Trauma	259
Undefined diagnoses	92

   We can see that the highest number of patients passed away due to Cardiac problems followed by Respiratory and Neurology diseases.

3. **Sampling**

   This data posed an imbalanced dataset problem as the target value under concern forms only 8.6% of the total samples. The target value under concern is the Hospital Death = 1 sample which means that the patient passed away during his ICU admission.

   We decided to address this issue by randomly under sampling the majority class by bringing it down to 8.6% samples. This step allows the ML models to learn better relationships within the dataset and in turn perform better in real-time.

4. **Dimensionality Reduction**

   Initially we had 80 features and post feature engineering, all the features have been transformed to categorical features. There were 55 numerical features which had different scales, outliers, and missing values.

   This resulted in a curse of dimensionality when we had to perform one hot encoding for modelling purposes. After one-hot encoding, the 80 features were transformed into 308 columns.

   Given the low row count, 308 columns would be problematic, and we decided to reduce the dimensionality by using the Random Forest Algorithm's variable importance method.

We first trained a random forest algorithm using all the features and later obtained the feature importance score for each independent variable/feature.

Looking at the feature importance score, we decided to take the top 20 features and used these features alone for the rest of the predictive modelling process.

5. **Predictive Modelling**

In this step we decided to build 5 models and get the best model or ensemble the predictions as in a voting classifier. The various models we trained are Decision Trees, Naïve Bayes, XG Boost, ANN, and Support Vector Machines.

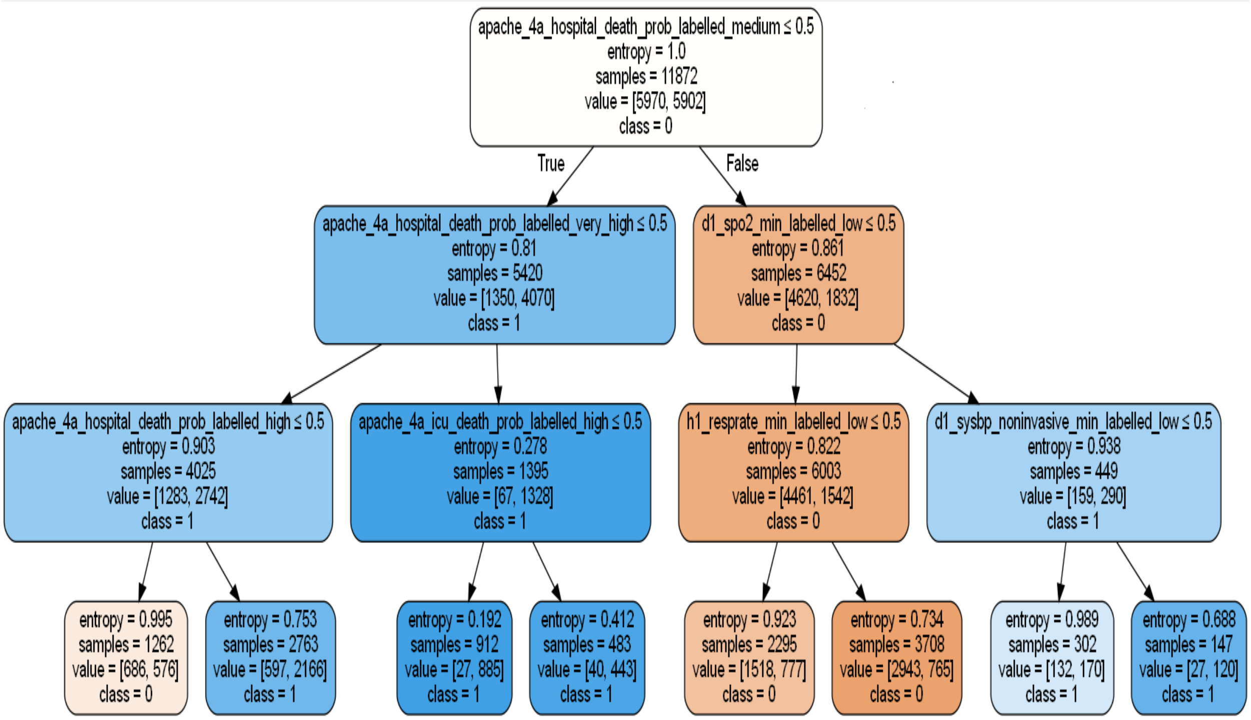
We have chosen the Recall Metric as the deciding factor for choosing between the various models. Recall score focusses on classifying the truly positive samples as positive. This is very important to us as we do not want to predict any dying person as a survivor.

I. Decision Trees

Decision trees are used for classification and regression problems and the main advantage of using this model is the benefit of understanding and interpretation of the model results. The Decision tree uses Gini Impurity or other entropy measures to make splits within features when deciding to create each node.

We trained a Decision Tree model using hyperparameter tuning and obtained the following metrics:

Recall: 72.9%, Precision: 78.9%, F1 Score: 75.8%, Accuracy: 75.7%



II. Naïve Bayes

Naive Bayes is a probabilistic machine learning algorithm based on the Bayes Theorem. It is used in a wide variety of classification tasks, it is not only simple algorithm but also fast, accurate, and reliable. It works particularly well with

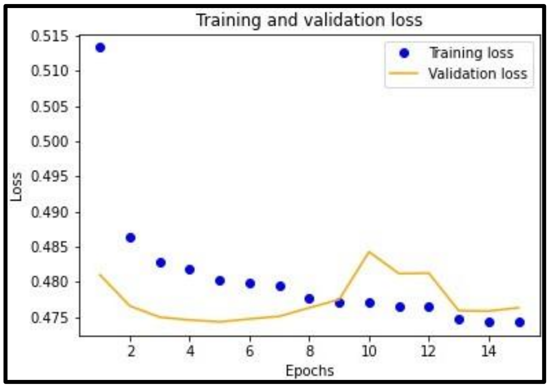
natural language processing (NLP) problems. We have a trained data set of patients details and corresponding target variable 'hospital_death'. We obtained the following metrics post training the model:
Recall: 66%, Precision: 81%, F1 Score: 73%, Accuracy: 75%

III. XGBoost

XGBoost is a type of Gradient Boosting model that is widely used in classification and regression tasks.
We trained an XGBoost model with our dataset and obtained the following metrics on the test set:
Recall: 72%, Precision: 79%, F1 Score: 75%, Accuracy: 76%

IV. Artificial Neural Networks

ANNs mimic the working of the human brain in the sense of neurons getting triggered by an impulse. We tried various versions of the ANN models by tweaking the hidden layer count and the number of hidden layers in the model. Later we finalized a model with 2 hidden layers both running the 'relu' activation function with 16 nodes each. The output layer had a single node with a sigmoid activation function to provide the probability of prediction.
Recall: 74%, Precision: 79%, F1 Score: 76%, Accuracy: 76%
The training process of the ANN is as illustrated below:



V. Random Forest

Random forest model utilizes the decision tree algorithm at its core but enhances it by implementing Bagging and Boosting. Bagging and Boosting is basically creating various decision trees that uses different subsets of samples for training. Subset of samples are created both using different feature sets and subsets of samples.
We trained Random Forest model on our data and obtained the following metrics:
Recall: 75.7 %, Precision: 80.6 %, F1 Score: 78.1 %, Accuracy: 78.3 %

VI. Support Vector Machines

Support vector machine (SVM) employs classification techniques to solve two-group classification problems. An SVM model can classify new text after being given sets of labeled training data for each category.
They offer two key advantages over more recent algorithms like neural networks: greater speed and improved performance with fewer samples (in the thousands). As a result, the approach is excellent for text classification issues, where it's typical to only have access to a dataset with a few thousand tags on each sample. SVM has three main Parameters. They are C, Kernel, and gamma. C is regularization parameter and is used to set the tolerance of the

model to allow misclassification of data points to achieve lower generalization error. Higher the value of C the lesser is the tolerance. The C value controls the penalty of misclassification. Kernel is a set of mathematical functions used by the SVM to find the relationship between two observations. Kernel takes the input and transforms it into the required form (usually higher dimensions). RBFs are the most popular kind of kernel functions. since it responds locally and infinitely throughout the entire x-axis. The inner product between two locations in an appropriate feature space is returned by the kernel functions. Thus, even in very high-dimensional areas, with low computational expense, by defining a notion of similarity. A single training example's influence is determined by the gamma parameter, with low values denoting "far" and large values denoting "near."

We used hyperparameter tuning with a suitable tuning grid and identified the best parameters as follows:

C=0.5, gamma= 'scale', kernel='rbf'

The final metrics that we obtained are:

Recall: 80.38 %, Precision: 71.43 %, F1 Score: 75.64 %

VII. Voting Classifier

To enhance the prediction capabilities of our models, we experimented with the option of Voting/ Ensemble Classifier by combining the predictions of all the 5 models. The voting method was used to identify the test sample. For instance, if a sample is voted as 1 (or Death) by 3 or more models, then we will predict this sample as 1, else all other samples will be classified as 0.

This voting classifier was implemented on our dataset and the following metrics were obtained:

Recall: 71 %, Precision: 80 %, F1 Score: 76 %, Accuracy: 77 %

6. Conclusion

Model	Accuracy	Precision	Recall	F1 - Score
Decision Tree	75.7	78.9	72.9	75.8
Random Forest	78.3	80.6	75.7	78.1
XG Boost	76	79	72	75
ANN	76	79	74	76
SVM	76	72	80.4	75.6
Voting Classifier	77	80	71	76

After training all the 5 models, we can understand that the Support Vector Machine Model has performed the best by offering the highest recall score of 80%. We can conclude that the SVM model works best for this dataset.

Evaluation, Outcomes and Discussion

We evaluated the dataset using 5 different robust Machine learning models and can confidently claim that we will be able to predict if a patient in likely to pass away with a recall of 80%.

This solution gives us the confidence that if we scale this solution to bigger datasets, we will get better results. This solution is just a Proof-of-Concept and if we can find significant patterns within these samples, then given an exhaustive training approach with several more samples will provide us with a better model.

We feel that the various data points collected for a patient in the ICU can explain the outcome of the treatment for that patient. Also, what is to be noted is, that these data points only offer us a snapshot of the patient's medical condition during admission and these data points do not talk about the day-to-day changes in the patient’s health. We feel that if we can access the daily details of the patient’s health stats and vitals will be even more useful to us to create a much better robust ML model.