Take Home Exam

December 18, 2023

Question 1

1.1 White Blood Cells Distribution

Figure 2 shows the density of the white blood cell count, including all observations from the *PatientExamination* dataset. The median and the mean value are, respectively, 10.6 and 12.3. By looking at the graph, a doctor should be able to tell whether a certain number of white blood cells is normal or not by comparing it with the population distribution. If the number is around 10 (+-5), it should considered a normal value.

Question 2

2.1. Model: XGBoost

Feature Engineering

First of all, I modified the PatientExamination table to obtain separate columns that showed the values for each type of measurement. I then merged the three datasets (Patient, PatientExamination, and Study) on the patient ID code. After doing this, I was able to look at the full set of variables available for prediction.

I decided to turn the date variables into separate variables accounting for the day, month and year of admission and study entry for later use. Additionally, noticing that some of the variables were categorical, I proceeded by encoding them and scaling them using a function previously defined.

Plotting the distribution of each variable in both the classification set and the training and testing set, I concluded that the two populations were pretty similar. I dropped all the variables that were missing or meaningless for prediction in the classification set (death_recorded_after_hospital_discharge, days_of_follow_up, study_entry_day, days_in_hospital_before_study, admission_day, Zodiac Sign, study_entry_year, study_entry_month, language).

Looking at the distribution of NA values in each variable, shown in Figure 3, I then decided to keep all observations and to impute all of the missing values with the median value (since we are considering medical measures, I thought it would be appropriate to simply impute the most common value in the population when missing).

To test if this was a good choice, I tried leaving the missing values instead of imputing them. The performance and the predictions of the model seemed to be quite unchanged. Therefore, the imputation should not be a fault of the model.

Model Selection

I tried to use different types of models before making a choice. I tried using a simple logistic regression, Random Forest and XGBoost. The model with the best accuracy level and the least misclassified values ended up being XGBoost. The accuracy of the model is 0.83.

The performance levels obtained by the alternative models are in Table 4.

Tuning

The hyperparameters that I ended up chosing are the following: learning_rate, max_depth, n_estimators, subsample, colsample_bytree, gamma. To choose the best value of each of them, I utilized grid search.

Evaluation and Testing

To evaluate my model, I relied primarily on its accuracy and the recall and precision values. Moreover, I looked at the learning curve (shown in Figure 5) to check whether the training sample was large enough to allow the model to learn without excessive overfitting.

2.2. Confusion Matrix

The confusion matrix obtained by the model in the test set is shown in Table 6.

True classes of outcomes are not evenly distributed, as we can see in the matrix. Of the deaths predicted, 75% are actual deaths. The model is able to correctly predict 61% of the actual deaths. While these are not outstanding results, I believe that being able to predict 61% of the actual deaths can be very useful, especially considering that the proportion of actual deaths in the population is low and therefore hard to identify.

2.4. Important Features

The most important positive features of the model are:

- Missing values: the model is able to handle possible missing values in future prediction sets
- Early stopping and hyperparameters' tuning: designed to prevent overfitting or underfitting
- Time variables: the model accounts for special periods like Covid19 pandemic and controls for years and months of admission
- Accuracy and precision have high values
- Recall has a lower value, but the model can still be useful in real life predictions

Question 3

3.1-3. Profitability Model

In order to choose which claims to sell, I evaluated three scenarios: doing nothing, selling all the claims and only selling the claims associated with predicted deaths (as predicted by the model created in Question 2).

I computed the total cost in each of the three scenarios as:

Total Cost of Doing Nothing = (TP[1] + FN[2]) $*$ Compensation Cost

Total Cost of Selling All = Current Patients[3] $*$ Selling Cost

Total Cost of Selling Predicted = (TP + FP[4]) $*$ Selling Cost + FN*Compensation Cost

In Figure 7, I plotted the total cost per strategy as the number of predicted deaths increases. As we can see, when the number of predicted deaths is lower than 195, it is more profitable to sell the claims of individuals predicted to die. On the contrary, when the number of predicted deaths is higher than 195, it is more profitable to sell all claims.

In our particular case, out of 1,000 patients, the model predicts 130 deaths. Therefore, the most profitable strategy is to only sell the claims associated to the predicted deaths (cost is approximated to millions):

Total cost of doing nothing	$= -129,000,000$
Total cost of selling all	$= -150,000,000$
Total cost of selling predicted	$= -99,000,000$
Strategy (3) savings compared to (1)	$= +30,000,000$
Strategy (3) savings compared to (2)	$= +51,000,000$

Table 1: Confusion Matrix

3.4. Type 1 and 2 Errors:

As we can see from Figure 10, the threshold varies depending on the recall (and therefore precision) values of our model. As the recall increases (and therefore precision decreases, as from Figure 8), the threshold above which selling all claims becomes more profitable increases.

Recall is closely related to the concept of type 2 errors. When recall increases, the likelihood of type 2 errors should decrease. On the other hand, precision is related to type 1 errors. As precision increases, the likelihood of making type 1 errors decreases.

Therefore, we can conclude that: when type 1 errors increase (lower precision, higher recall) the decision threshold increases; when type 2 errors increase (lower recall, higher precision) the decision threshold decreases.

[1] Estimated number of true positives according to the model's recall and precision.

[2] Estimated number of false negatives according to the model's recall and precision.

[3] The total number of current patients, 1000 in our case.

[4] Estimated number of false positives according to the model's recall and precision.

Appendix

Question 1

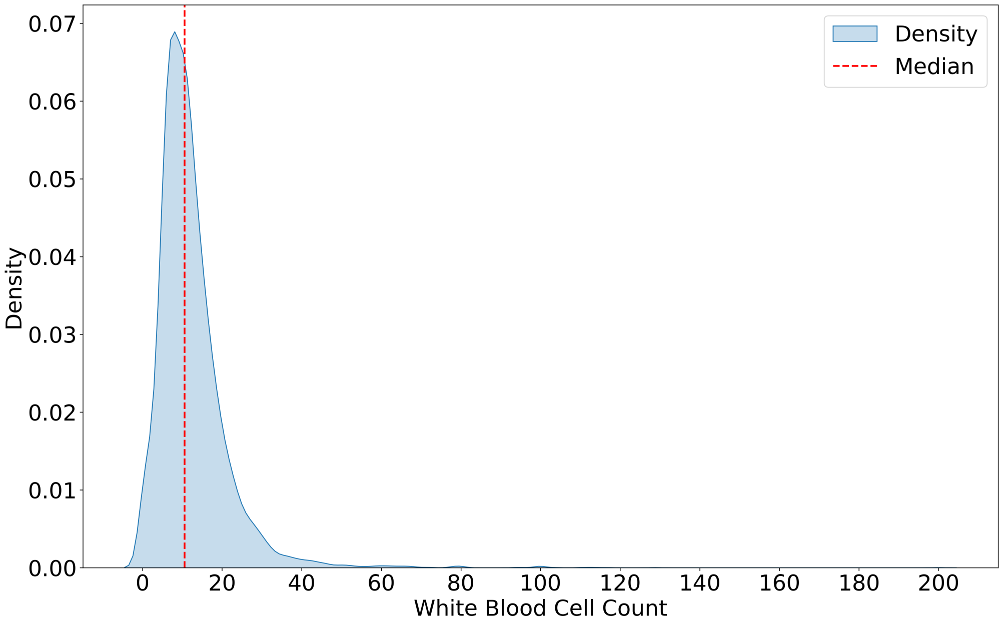

Figure 2: White Blood Cells Kernel Density

Question 2

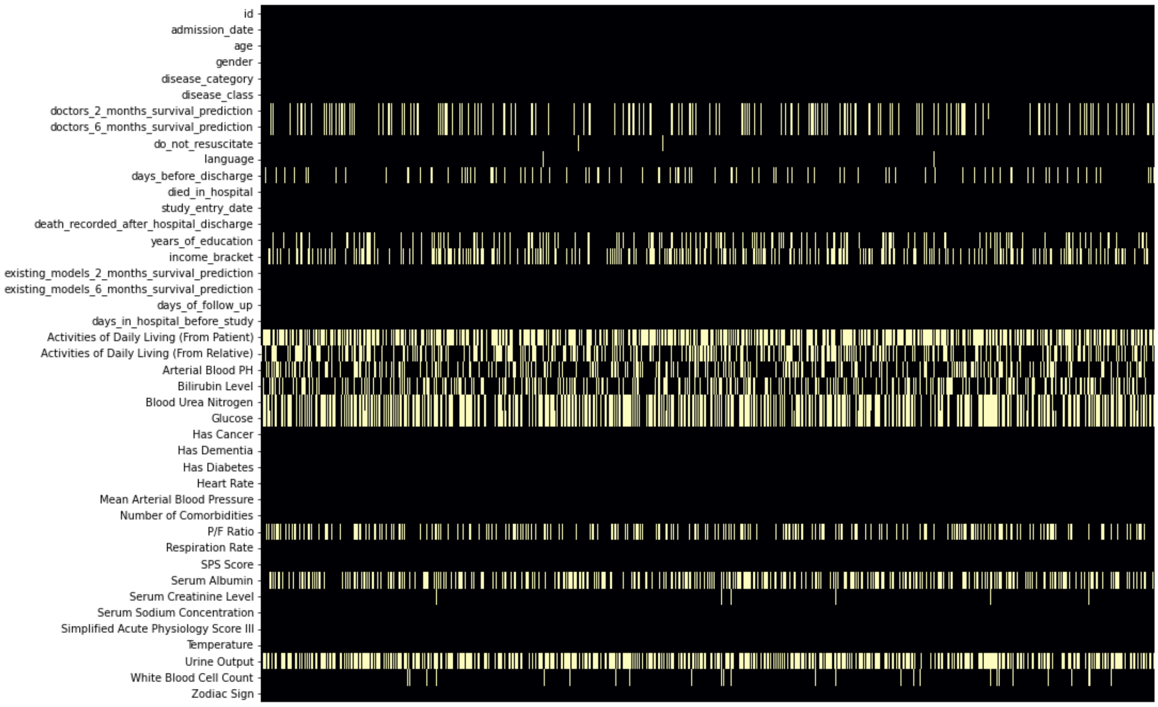

Figure 3: Missing values for each variable

	Logistic Regression	Random Forest	XGBoost
Accuracy	0.80	0.82	0.83
Recall	0.46	0.51	0.61
Precision	0.72	0.79	0.75

Table 4: Models' Performance (Recall and Precision Refer to Predicted Deaths)

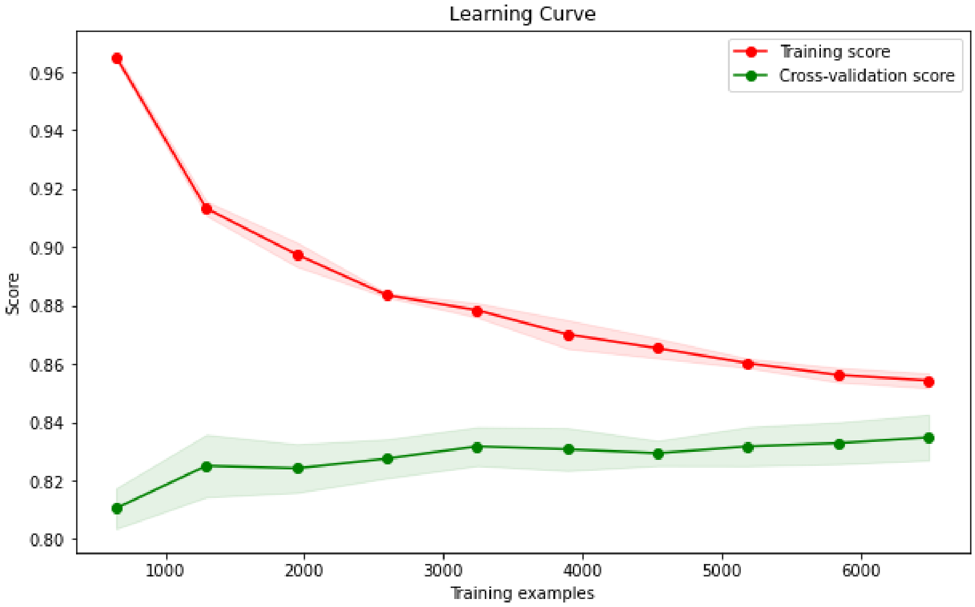

Figure 5: Learning Curve

	Actual Negative	**Actual Positive**
Predicted Negative	1613	138
Predicted Positive	264	417

Table 6: Confusion Matrix

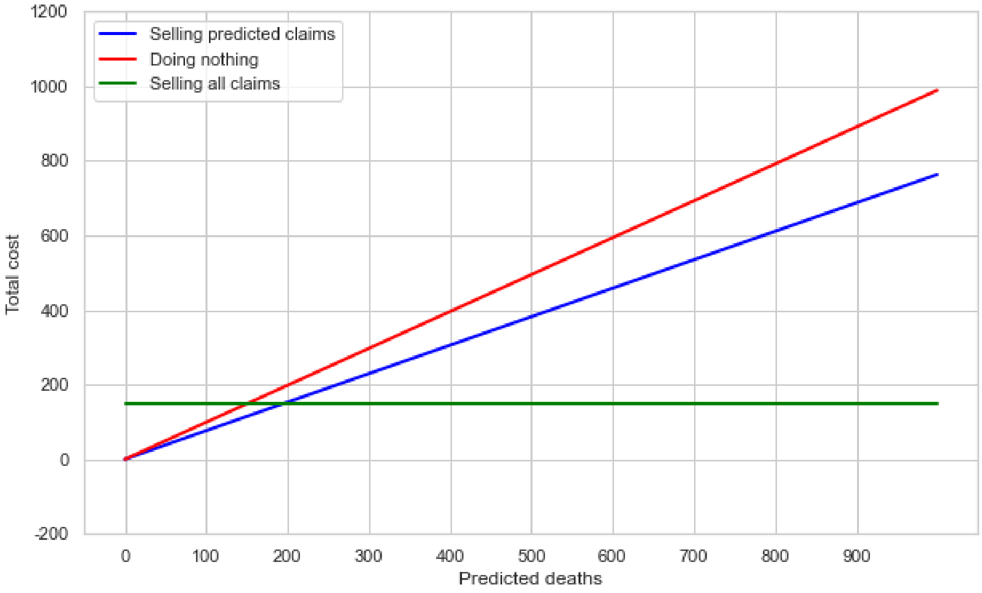

Figure 7: Cost Strategies

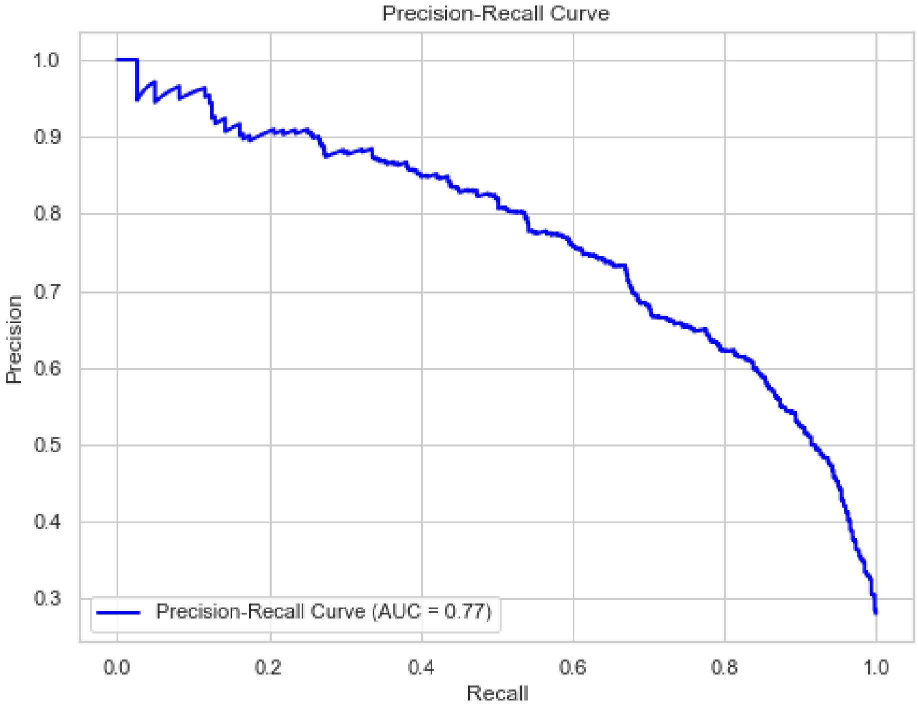

Figure 8: Precision-Recall Curve

Recall	Threshold
0.1	30.0
0.2	62.0
0.3	92.0
0.4	123.0
0.5	154.0
0.6	192.0
0.7	235.0
0.8	278.0
0.9	340.0

Table 9: Threshold as a Function of the Recall Value

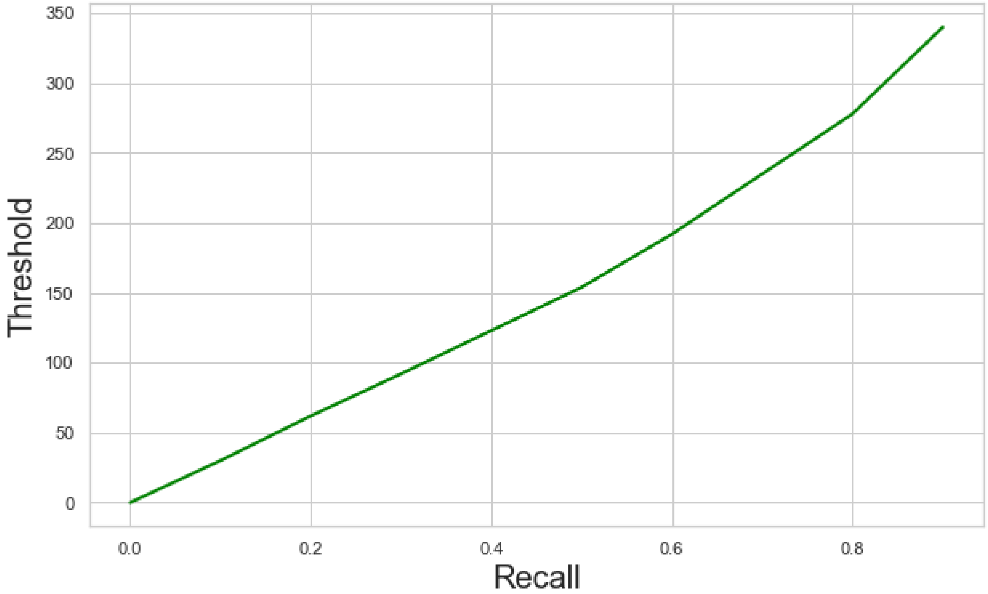

Figure 10: Threshold as a Function of Recall