Sebastián Varón

DATA1030-Fall2022

Oct 21, 2022

Machine Learning report

https://github.com/SebastianVaronCrews/Early-Warning-System

Introduction (10 points)

Our target variable is the student's grade in the final academic period- our main focus is casting this as a regression problem on the continuous final grade value.

However, we acknowledge past approaches to a multi-class classification problem on this dataset by using a regression as a basis for the different classes. [Cortez and Silva 2008]. We also acknowledge that this problem is commonly casted as a binary classification problem (academic early warning system) that flags the student (as being in good standing or in bad standing).

The dropout rate is a metric commonly observed by different countries' governments' education departments. It can be helpful to intervene with a student early on to prevent dropouts and encourage good academic performance.

In this dataset, there are 643 data points and 33 features.

The features include demographic data about the student, some information about the student's school, and grade information from different periods across the semester. There are categorical features, such as the mother and father's job titles; ordinal features such as the mother and father's education level; and continuous features such as final grade. Other potentially informative features to build an "academic early warning system" (and that are also anti-correlated to final student grade) include absences, failures, and weekend alcohol consumption

The data was collected via questionnaires and school report cards from one particular highschool in Portugal. There is data for a mathematics classroom, as well as a Portuguese language classroom.

The dataset is well documented in UCI machine learning repositories. https://archive.ics.uci.edu/ml/datasets/student+performance

There you will find a detailed description for each feature.

On UCI machine learning repositories there is a citation request to a paper by Cortez and Silva (2008). In this paper, the authors' approach was threefold: they created a binary

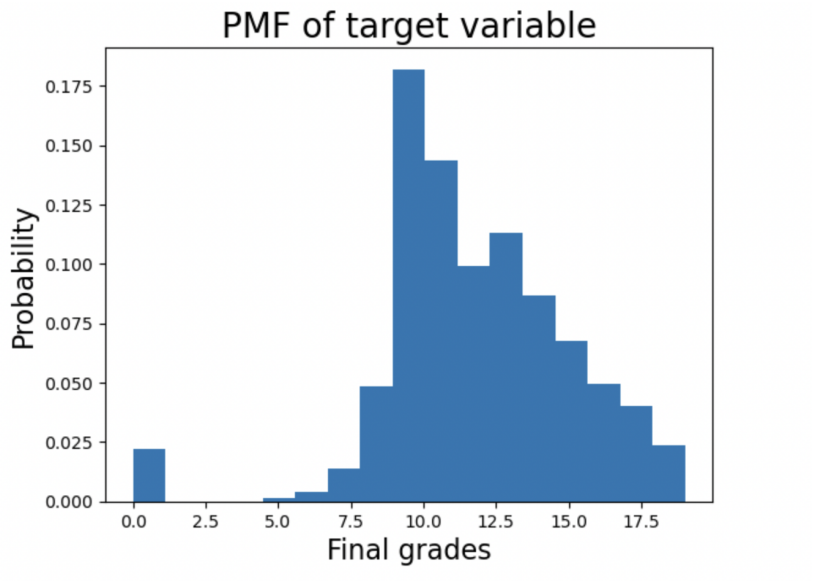
(pass/fail) predictive model, a five- class classifier, and a regression on the continuous final grade. Below is a table of their reported results for their binary classifier:

Measure	Value
Sensitivity	0.9756
Specificity	0.5605
Precision	0.8743
Negative Predictive Value	0.8800
False Positive Rate	0.4395
False Discovery Rate	0.1257
False Negative Rate	0.0244
Accuracy	0.8752
F1 Score	0.9222
Matthews Correlation Coefficient	0.6359

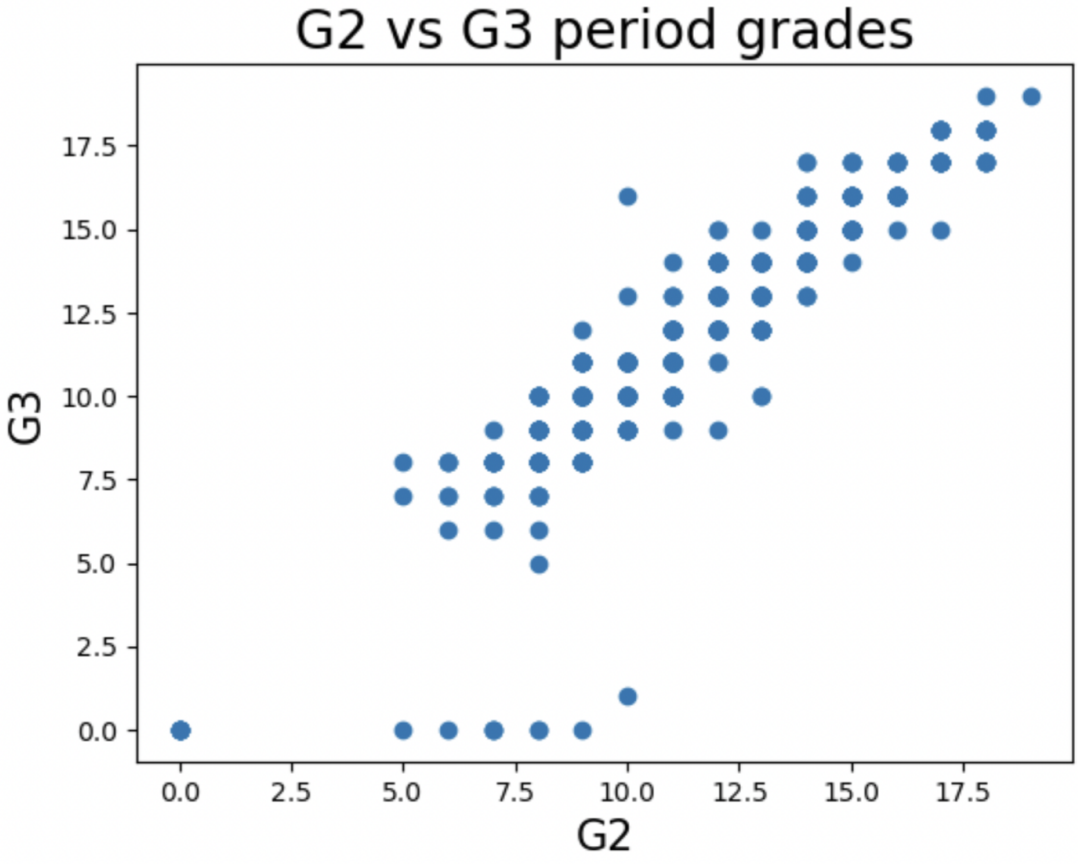
Higher sensitivity and lower specificity means their classifier's strength is in covering the fail class, even at the expense of higher false alarm rate. The paper also cites previous work by Pardos et al. (2006) which achieved a regression on final student grades with a mean squared error of 15%.

Exploratory Data Analysis

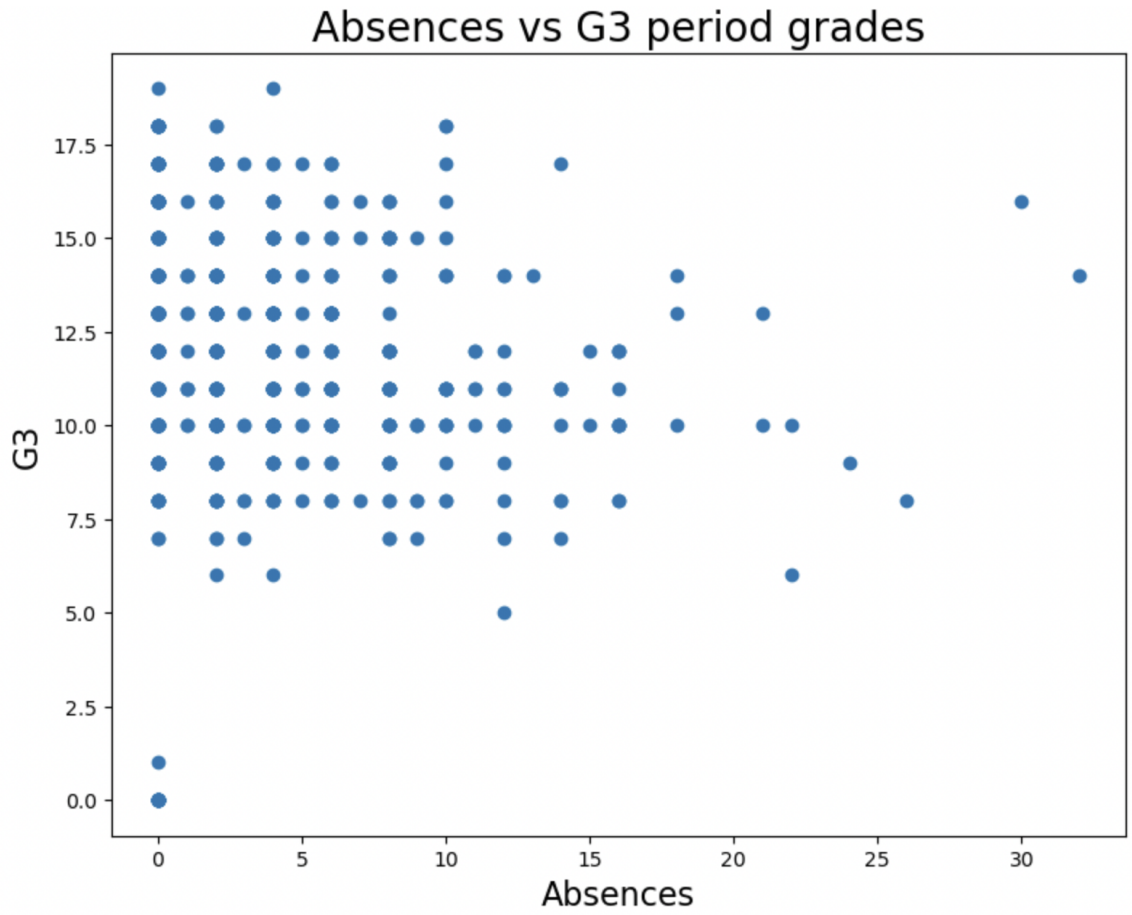
As part of the EDA, we created a normalized histogram (i.e, probability mass function) for our target variable, final grades. The result was a somewhat normally distributed distribution that varies within one order of magnitude. Notice the outliers with a zero final grade (this may or may not be a fault in the data).



We also observed a strong correlation between final grades and past period grades. This relationship was to be expected and also emphasized by [Cortez and Silva 2008].



As well as an anti-correlation between final grades and absences:



We see the highest final grades (in this particular Portuguese language classroom, at this particular school) are achieved by individuals with lower total absences. As absences increase, final grades generally decrease.

Methods: Data preprocessing and splitting

The dataset had no missing values.

This dataset is IID, as the rows are all independent and identically distributed.

In real-life application settings, an academic early warning system would have a limited amount of data about a student at any given time, and this data will be updated as more data is received. Although the frequency of arrival of new data is not at the level of having to implement a time-series approach with autocorrelation; it is still useful to keep the real-life application in mind when creating a model.

To mimic future use of the model, it will be useful (in the modeling step of the pipeline), to create one model with all demographic features, but no grade/school-report features; then create a second model with all demographic features and only grades from the first period; and finally create a third model with all demographic features and first as well as second period grades. All three models will be used to predict the final period grades.

Thus, a different amount of features must be used in each feature matrix. After selecting the features for each of the three models, each of these sets of features can be preprocessed accordingly. Because it does not contain group structure, or time-series structure, we use sklearn's basic train_test_split function to split the dataset into other (80%) and test (20%) sets. Shuffling is important with IID data, and it is useful to apply sklearn's k-fold for cross-validation, especially because of our low number of data points .

Some features were categorical, such as father or mother's job label. For these, we used sklearn's OneHotEncoder to create a one-hot vector for each job label in the dataset.In the case of continuous values, such as different period grades, number of absences, and number of failures, we applied sklearn's StandardScaler and MinMax scalar. After preprocessing, we have 79 features.

Methods: pipeline and parameter tuning

The approach for creating the models was based on the particular application setting. Because we want to create an early academic warning system, it would be advantageous to set-up the model in a way such that we can generate predictions at different points of the semester. At the very beginning of the term, there is no grade information. However, if the available features can be used to generate a prediction, it could serve as an early intervention for that student.

As soon as grades arrive (a third of the way through the semester), the model can be re-trained to produce new, actualized predictions.

This happens again (two thirds of the way through the semester)- new grades arrive and the models must be re-trained and predictions remade- in order to predict final period performance.

For this reason we trained a set of models in period 0, a set of models in period 1, and a set of models in period 2. Only available features were used in each step.

Each set of models (a set for each period) consists of 6 regressors: random forest, support vector machines, lasso, a ridge, elastic-net, and k-nearest neighbors. Additionally, a seventh model was trained using the XGBoost framework.

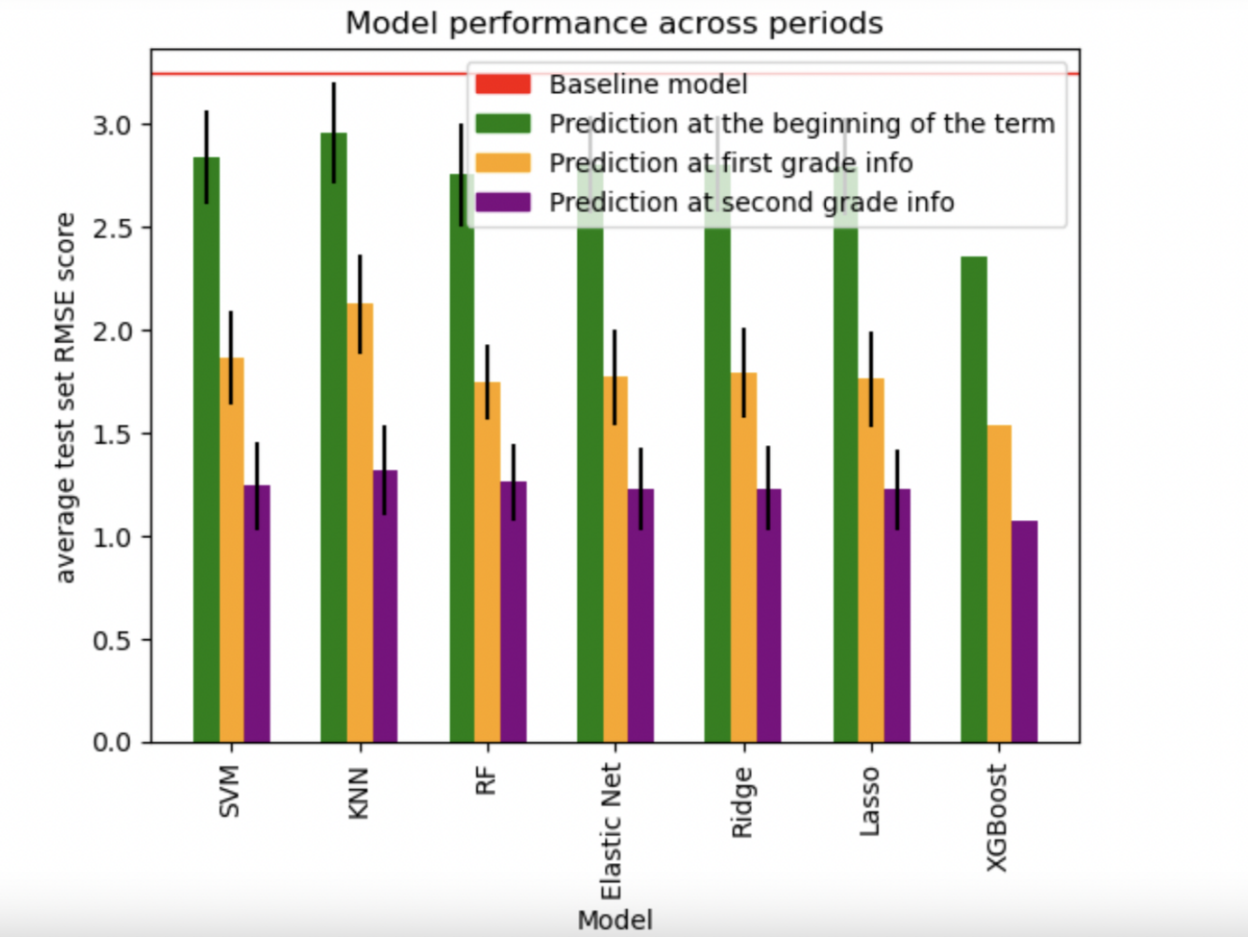
The following is a table with the parameters tested and tuned for each model:

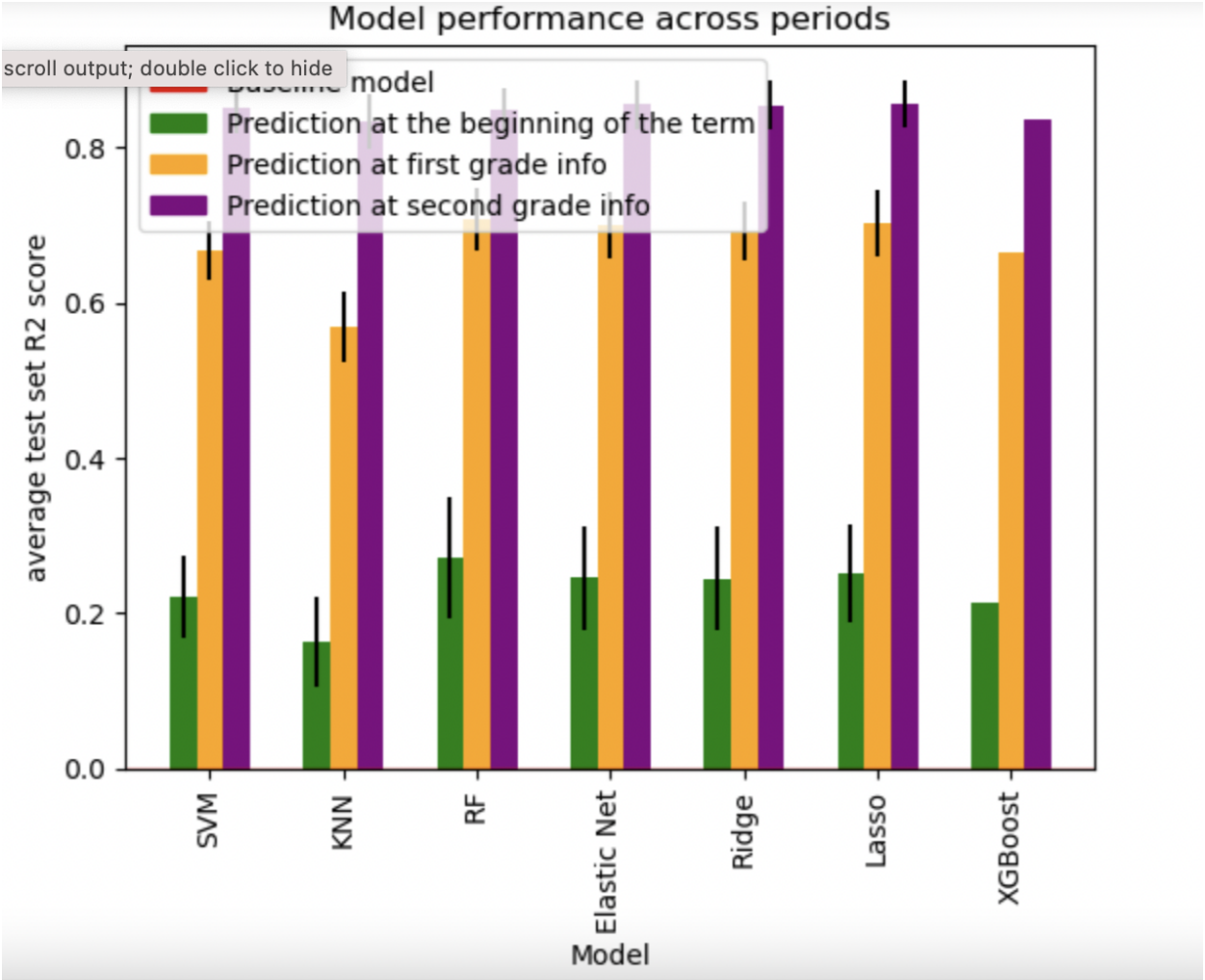
Random Forest	Max depth': [1, 3, 10, 30, 100],	'maxfeatures: [0.5,0.75,1.0]				
SVM	gamma/ bandwidth	c/ regularization				
KNN	N (# of neighbors)					
Lasso	Alpha (regularization penalty)					
Ridge	Alpha (regularization penalty)					
Elastic net	'alpha':[1e-4, 1e-3, 1e-2, 1e-1, 1e0, 1e1, 1e2],	'elasticnet__l1_ratio': [0.1, 0.25, 0.5, 0.75, 0.9],	'elasticnet__max_iter': [100000] }			
XGboost	"learning_rate": [0.03] ,	"n_estimators": [10000]	"seed": [0]	"missing": [np.nan]	"colsample_bytree": [0.9]	"subsample": [0.66]

Methods: evaluation metrics and some results

Both the RMSE and the R2 score were used to evaluate the models. RMSE is useful because it is in the same unit as our target variable, while R2 is a dimensionless quantity- they provide different vantage points from which to evaluate. The mean target value was used to predict the baseline for both RMSE and R2 which were 3.26 and 0.0 respectively.

In terms of RMSE score, the lasso regression was our best performing linear model. XGboost was the only model capable of outperforming Lasso. The results below show the performance for all the models across the different periods- a total of 21 models were trained and evaluated against the baseline. For both RMSE and R2 metrics, all models outperformed the baseline. The error bar here represents the model's uncertainty due to randomness.





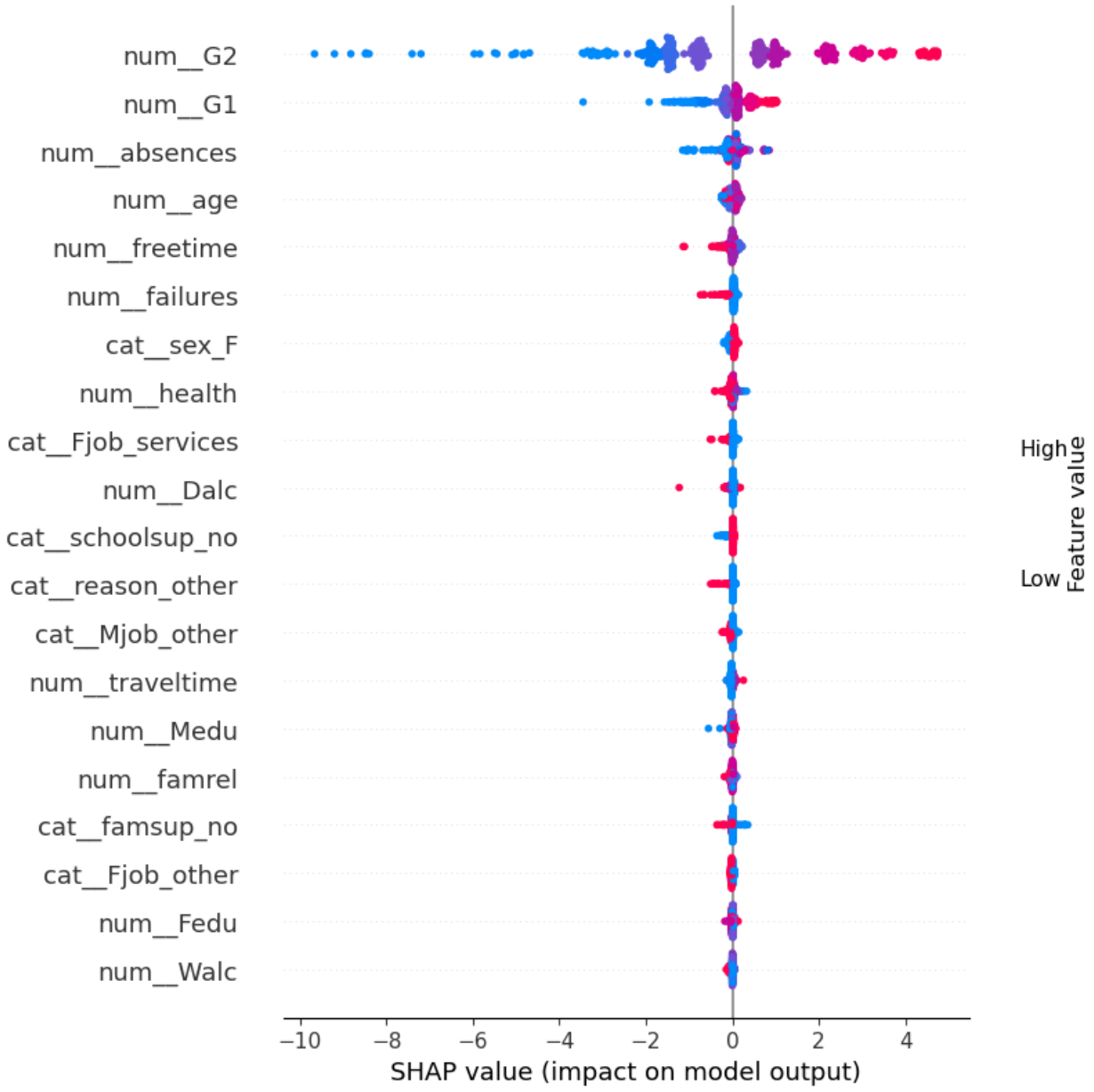
Additional results:

As mentioned above, our most predictive model in terms of RMSE is the XGboost, across all time periods. However, in terms of R2 score, only KNN achieves a slightly better performance than XGboost.

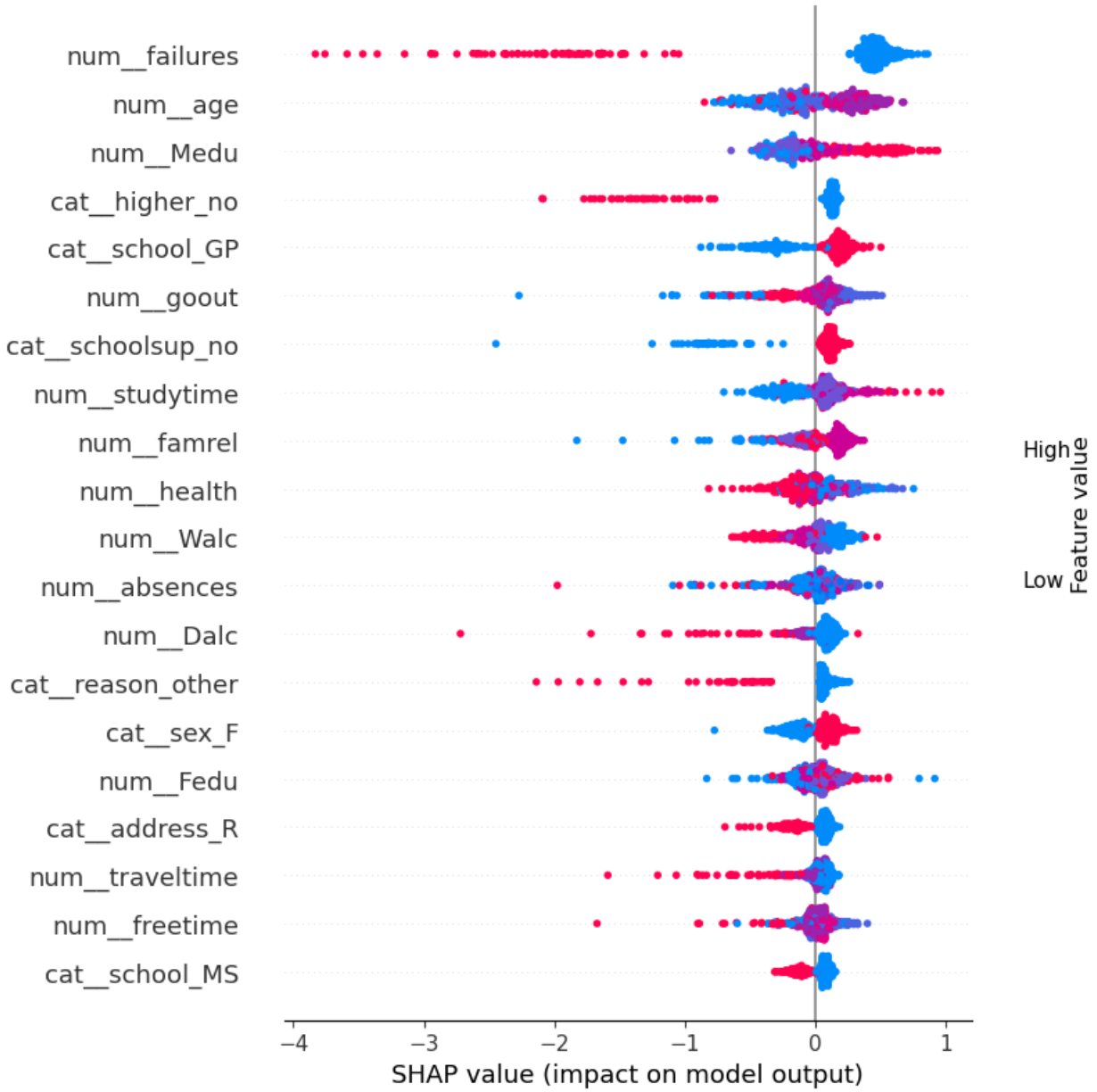
Global feature importance:

Using the XGboost framework in conjunction with the SHAP values, our most predictive model (XGboost later in the semester) reveals the most important features to be previous period

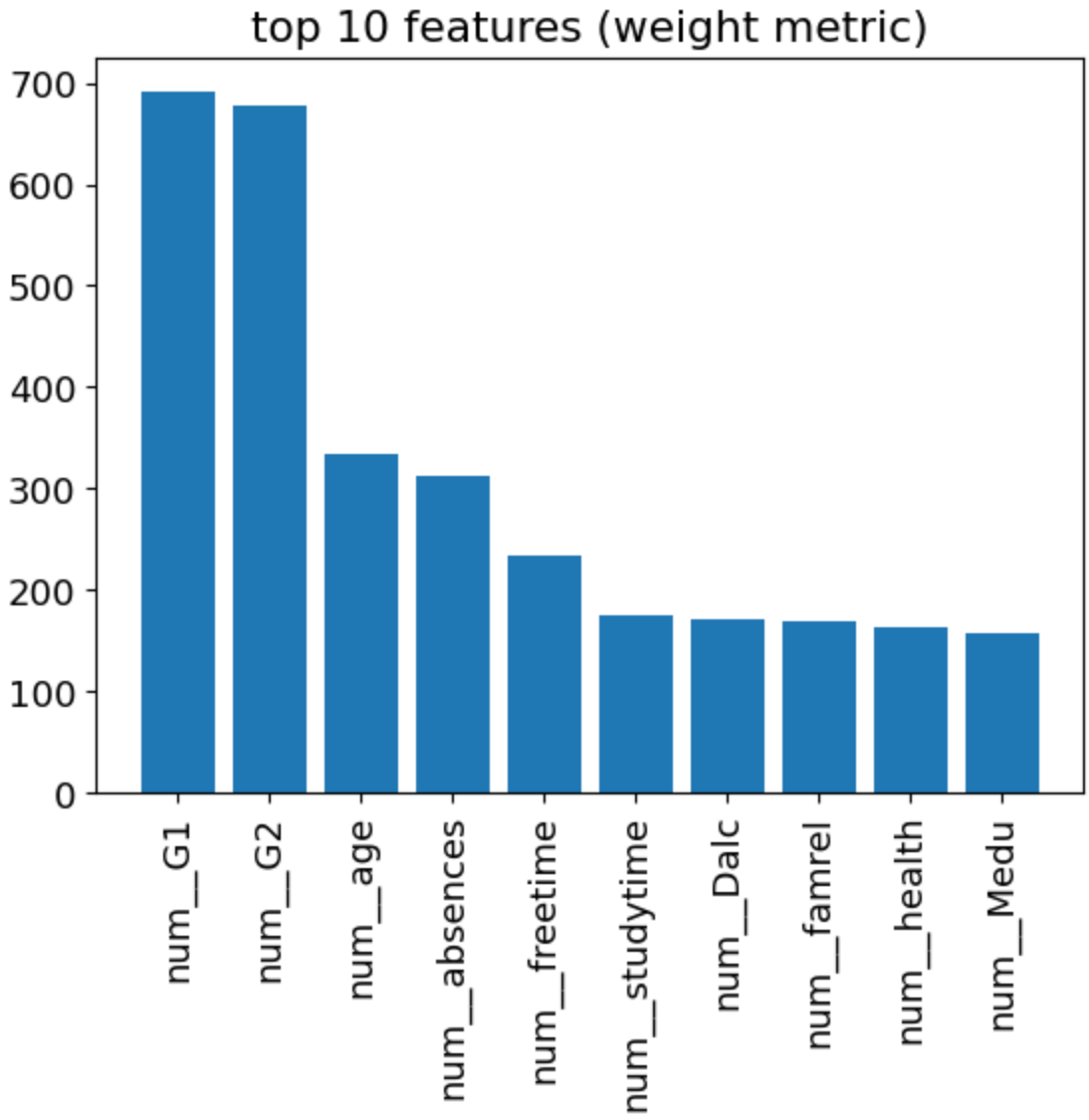
grades G1 and G2.

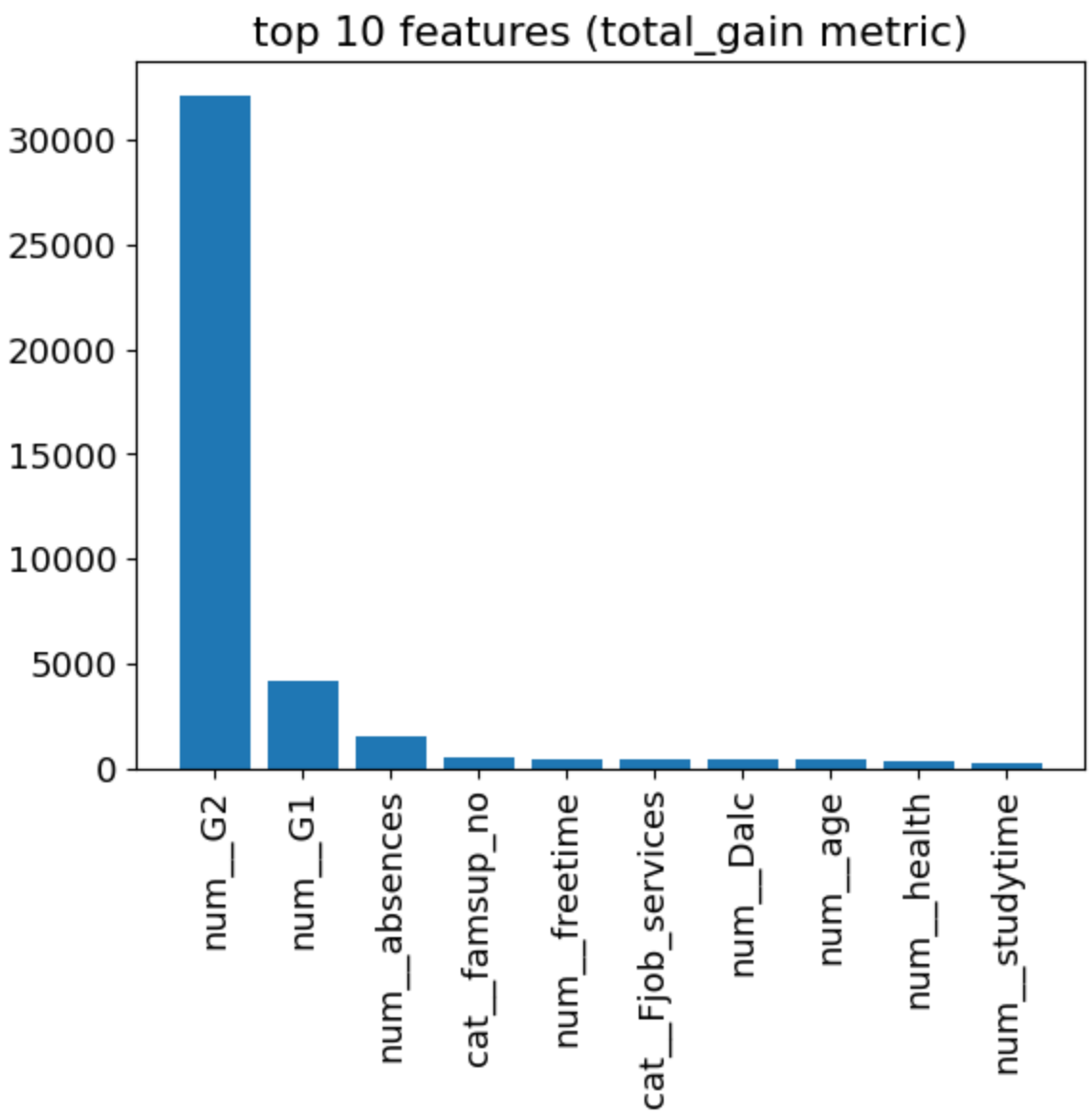


Earlier in the semester however, we do not have the previous grade performance features, and the most important feature according to SHAP global importance is now the number of failures:



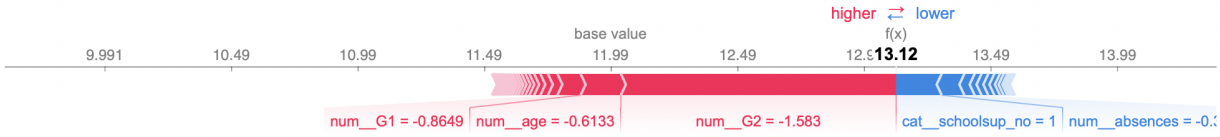
We can also consider the weight and total gain metrics, both of which reveal previous grade performance to be the most important features. Other important features include number of absences and study time.


top 10 features (weight metric)
700
600
500
400
300
200
100
0
num__G1
num__G2
num__age
num__absences
num__freetime
num__studytime
num__Dalc
num__famrel
num__health
num__Medu

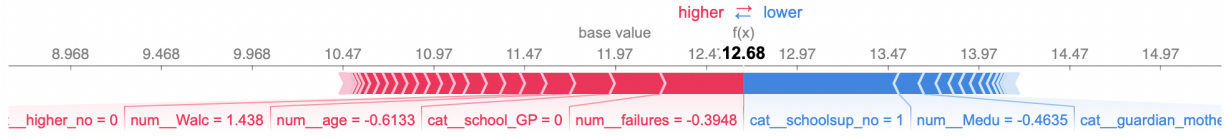


The SHAP package can also be used to evaluate local feature importance. Using the instance at index 0 as an example, we can see that for our most predictive model, previous performance is the most impactful feature for that individual's prediction.

XGBoost Local Shap value towards the end of the semester:



XGBoost Local Shap value towards the beginning of the semester:



Interestingly, at the beginning of the semester, the model relies on other features such as number of failures and the category of the school, which are most predictive in the absence of grade information.

Unimportant features across the different models include sex category and traveltime.

Outlook

Models, especially in the case of the XGBoost model, may be marginally improved by further improving the hyperparameters. With the XGBoost model, the set of hyperparameters to choose from was small, so that it can be expanded- although it will take longer to train and cross- validate.

Interpretability can also be improved by studying the feature importances of the various models in the pipeline that were imported from sklearn. In this paper, we reported the global importance metrics (see above) from our best performing model- our XGboost trained with previous grade information. However, these global importance metrics can be supplemented by those of the other models (e.g, SVM, RF, et al.)

References

P. Cortez and A. Silva. Using Data Mining to Predict Secondary School Student Performance. In A. Brito and J. Teixeira Eds., Proceedings of 5th FUture BUsiness TEChnology Conference (FUBUTEC 2008) pp. 5-12, Porto, Portugal, April, 2008, EUROSIS, ISBN 978-9077381-39-7.