MLBD Milestone 4 Report

Ahmed Ezzo, Julian Blackwell, Thomas Berkane

1 Introduction

We make predictions for student reflection responses to the "How do you feel about your learning progress" question. We base our predictions on both the students' session interactions (response time, response correctness) and the characteristics of the session (number of questions, feedback mode, etc.).

Two approaches are attempted: an approach using aggregated features and machine learning models, and an approach using time series data as input to neural network models. All the code can be found at https://github.com/ML4BD/milestone-4-team-dino.

2 Data processing

We build a dataframe from the raw data with columns participant_id, answer_time, mode, feedback_mode, force_reflection, timer, is_solo, video, image, correctness, nth_answer and response. This dataframe can be used to aggregate participant answers, as well as be used for time series analysis.

We impute missing data with a different strategy depending on the nature of the data. For categorical features, we replace missing data by the most frequent class. For numerical features, we simply replace with the mean of defined values.

In order for all features to be on the same scale, we also normalize our data.

To perform predictions on the time series data, we use a fixed number of time steps (10 in our case). Since not all participants have enough answers, we add some padding to our data.

3 Models

3.1 Aggregated data

For the classification using the aggregated data approach, we try three different models: multi-class Logistic Regression, Support Vector Machines, and Random Forest. Each model is trained on 80% of the data and then tested on the remaining 20%. To tune the hyperparameters, we apply 5-fold Grid Search Cross Validation.

We first attempt multi-class classification (classes = {happy, neutral, upset}), and then binary prediction (classes = {happy, not happy}).

3.2 Time series data

For time series data as input, we train different recurrent neural network architectures for the binary prediction problem (happy or not). We consider a simple RNN, a LSTM, and gated recurrent units (GRU). We split the data such that 10 percent is for testing, and the remaining

90 percent are split 80-20 for training and validation data. The neural networks are trained with binary cross entropy loss.

4 Results

To compare the models, we use accuracy (normal and balanced) as a metric. The baseline accuracy is computed by considering a model which always predicts the most frequent class ("happy" in our case), which gives an accuracy of about 0.560. Finally, we also compare the recurrent neural network architectures against a simple neural network with a single hidden fully connected layer.

Table 1 shows the **best** results for each model. We observe that for the multiclass classification problem, models only slightly improve in normal accuracy, but better improve the balanced accuracy (better predictive accuracy for each class), most notably random forests. For the binary prediction task, our models now perform a lot better than the most-frequent baseline predictor (with both LR and RF performing very closely). Finally, for the time series binary prediction task, we observe that recurrent neural network architectures do not necessarily perform better than the single fully connected hidden layer network on the binary prediction task. The best recurrent architecture seems to be the gated recurrent units which has equal balanced accuracy but slightly higher ROC compared to the normal neural network.

Data and Task	Model	Accuracy	Balanced Accuracy	ROC AUC
Aggregated multiclass	Baseline	0.560	0.333	-
——	Random Forest	0.571	0.398	-
——	Logistic Regression	0.572	0.381	-
——	SVM	0.572	0.376	-
Aggregated binary	Baseline	0.560	0.500	0.500
——	Random Forest	0.645	0.629	0.686
——	Logistic Regression	0.648	0.627	0.687
Time series binary	Fully-connected	-	0.616	0.673
——	Simple RNN	-	0.612	0.669
——	LSTM	-	0.615	0.674
——	GRU	-	0.616	0.674

Table 1: Model performance with tuned hyperparameters

Below are bar plots of the mean accuracies of each model with min-max errorbars (see 1, 2):

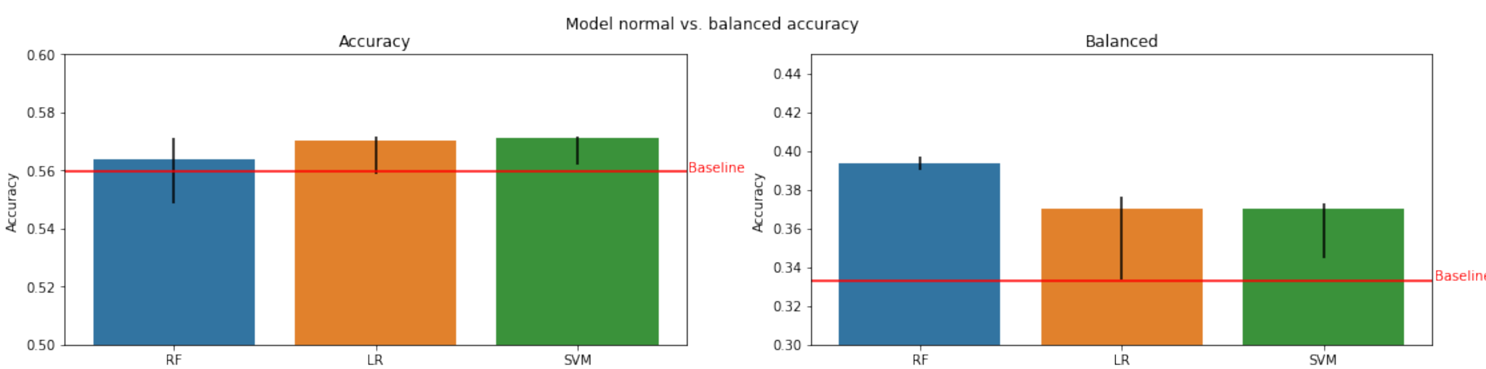

Figure 1: **Aggregated**: Multi-class classification

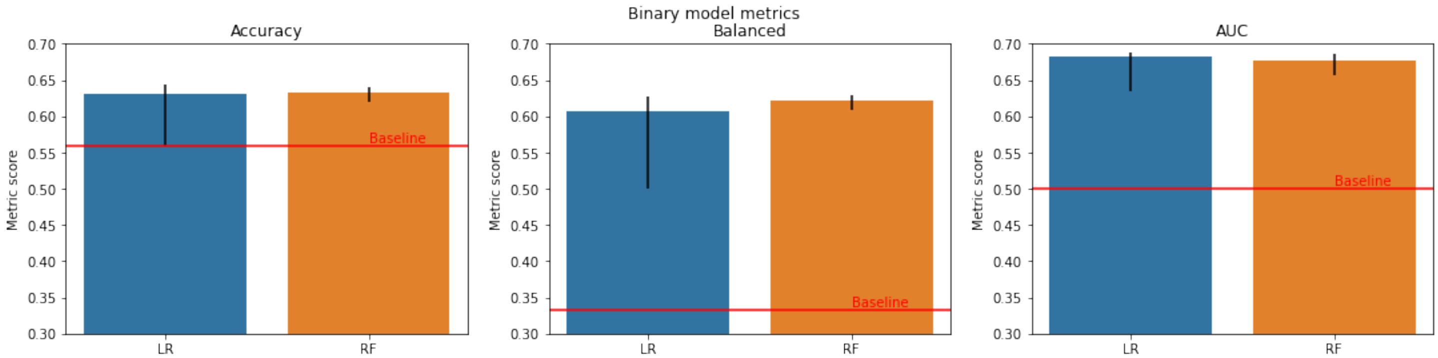

Figure 2: **Aggregated**: Binary classification

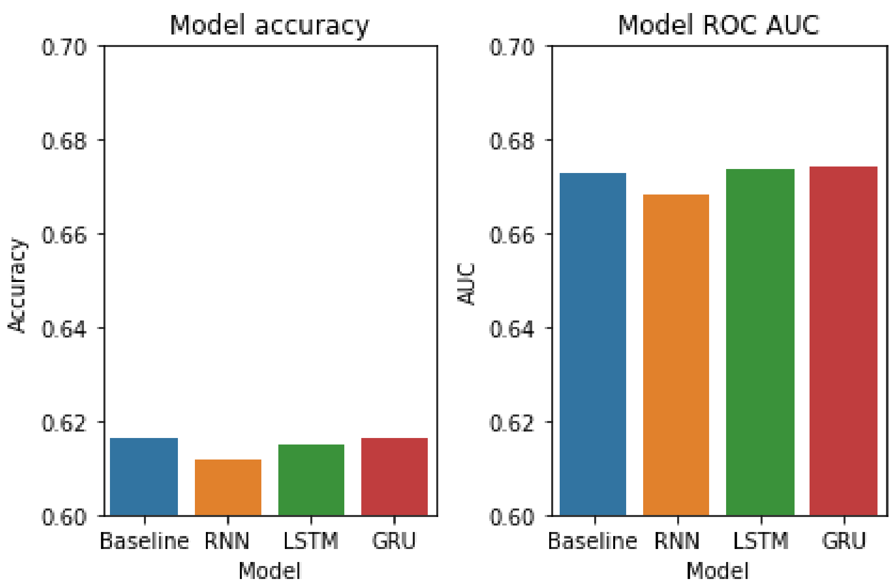

Figure 3: **Time Series**: Binary classification

The aggregated data models perform better than feeding neural networks time series data with the participants' first ten answers.