Student Academic Performance Prediction with Recurrent Neural Network

Feier Xiao

College of Computer Science and Technology

Australian National University

u660937@anu.edu.au

Abstract. The research for the artificial neural network has been increasingly popular in recent years, as it performs fairly well in a variety of areas, such as computer vision, speech recognition, etc. This paper evaluates the effectiveness of Recurrent Neural Network (RNN), a deep learning neural network, specialize in time-series analysis. This paper builds an RNN and a normal back-propagation neural network, perform a prediction with a student marking dataset. It turns out that the RNN perform better than the normal neural network, with an accuracy of 72.04% compared with the baseline accuracy of 65.36%.

Keywords: artificial neural network, recurrent neural network, student academic performance prediction

1 Introduction

1.1 Background

Artificial neural network (ANN) or neural network, a computational model inspired by human brains structure, is trained to recognize underlying features of the data [1]. Back-propagation is a widely used algorithm for supervised learning with the neural network. The model adjusts the weights of the neurons to minimizes the loss, which is the calculated difference between prediction and actual output [2].

1.2 Dataset

The dataset used in this paper is the same one as the research [3][4]. It contains 5 columns of student's information, 10 columns of marks for several assessments (lab, assignment, tutorial, etc.), and 1 column of the final mark for 153 students at The University of New South Wales. It should be noticed that 352 (19.3%) of the marking data are missing, imputed by a dot, and there are also 24 zero marks. Besides, around 49% of entries contain missing values, including 8 entries have no marking data at all. Since this data set has a fairly large proportion of missing data, the performance of the training model might be limited. For more statistics about the data set, please refer to the Appendix. The task is to predict a final grade (D, CR, P, F) based on the mark of other assessments (Lab, Assignment, etc.). The motivation behind is to allow the students to be aware of the final grades that might have, according to the assessment they have already known during the semester, so the students could have better management of their performance [5].

1.3 Previous Work: BDR

Researchers have found that the non-parametric method, including back-propagation algorithm, has a tradeoff between variance and bias, where variance, in this case, is sensitive to the data and the bias is failing to recognize features [6]. The dataset requires to achieve both low variance and low bias is massive, but in reality, the training normally doesn't have enough data. There are a couple of techniques to reduce variance by increasing bias, such as pruning [7]. Outlier removal is one of the effective methods to reduce variance. Outlier is a set of statistically inconsistent data compared with other data in the dataset [8]. It could cause large weight changes with back-propagation, and the time model spent in learning will increase, so as the variance [3]. If the outlier can be detected and removed, the variance should be decreased. Therefore, outlier detection is a challenging problem. The BDR is invented since the researchers found the frequency of the errors in the training set could be approximately distributed in a bimodal shape (Fig. 1.), where the low error area contains features already learned by the model, and the high error area is most likely to contains the outliers [3]. The BDR algorithm removes a subset of the patterns that fall in the high error area, and continue the training with the pruned dataset, until the variance of the errors is fairly low, which means the high error area is almost out of patterns [3]. However, the author has conducted an experiment on BDR algorithm and found that the neural network with BDR doesn't perform better than the one without BDR, with an accuracy of 60.72% compared with the baseline accuracy of 65.36%, which is accordance with the research of [3][4]. It is proved that the BDR could detect outlier, but its unsatisfied performance on the bias might indicate the usefulness of the outlier data in this dataset. Therefore, this paper will explore other techniques to improve prediction accuracy.

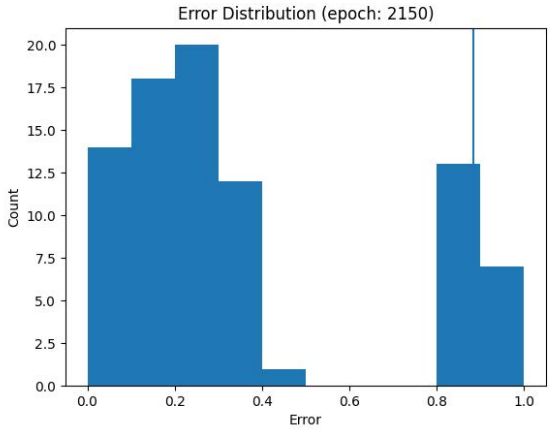

Fig. 1. Bimodal distribution of errors found during the training with BDR algorithm.

1.4 Previous Work: RNN

Unlike a normal neural network, a Recurrent Neural Network (RNN) is one of the deep neural networks, that is able to perform prediction on the time series data [9][10], with a structure of Figure 2. The RNN with this structure is capable of picking up the relationship between data points. This dataset contains assessments for a student at different time, and it is worth investigating the relationship between the assessments, which makes the RNN an appropriate technique for this prediction task. Also, the research in [11] applied the RNN to predict final grades of students from the other assessments, which is similar to the task in this paper. The outcomes of the research [11] are promising, and it has been proven that the RNN is effective to early prediction of final grades. Based on the result of the previous work, this paper will explore and evaluate the improvement on prediction with the RNN, compared with the normal back-propagation neural network.

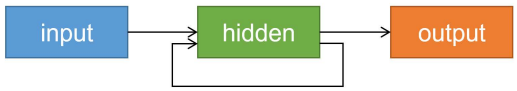

Fig. 2. Recurrent Neural Network Neuron Structure

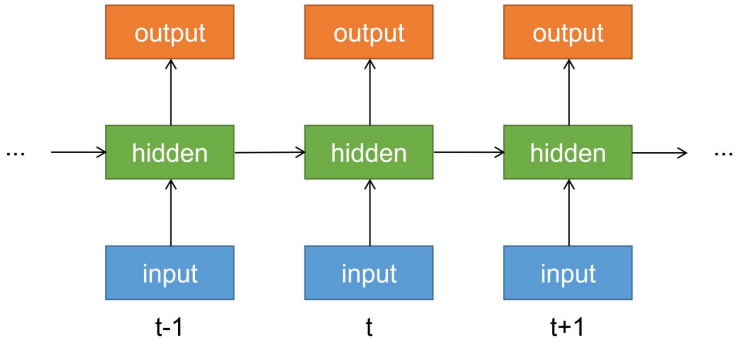

Fig. 3. Recurrent Neural Network Structure, where t is time

2 Method

2.1 Data Pre-processing

To prepare the data for the neural network training, several processes have been performed as below:

- The student's information is considered irrelevant to the prediction of the final mark, thus removed.

- The 10 columns of marks of assessment are normalized into the scale from 0 to 1, using Formula. As stated by the research [12], the normalization process is important to product accuracy output and greatly reduce training time.

$$z = \frac{x - \min(X)}{\max(X) - \min(X)} \quad . \tag{1}$$

- All other missing data is imputed to 0. The tremendous amount of missing data could have a negative effect on the prediction task [13]. There are various ways for imputation for missing values, but in this task, imputation might bring in bias. Also, the missing values in the academic assessment are more likely to be the student failing to attend the exam or submit their assignments, so imputation to zero is acceptable.

- The final mark is using the hundred-mark system. Similar to [3][4], it is converted into the category as Table 1.

Table 1. Conversion from grade to category.

Final Mark	Category
> 75	0
65 - 74	1
50 - 64	2
< 50	3

2.2 Network Topology and Techniques

- The topology (Table 2) of the normal neural network is similar to the normal neural network used in [3][4]. The researcher in [3] reckon 5 or 10 hidden neurons doesn't make too much difference but the author finds that 10 hidden neurons are slightly better in a few runs. For the RNN, the input will take a sequence and the hidden layer is set to 22 after fine-tuning.

- The split ratio for both neural networks is 7:2:1 (training: validation: test) for the cross-validation, since the dataset in this task is small, and cross-validation can be helpful in preventing the overfitting issues [14].

- The researchers in [4] used the sigmoid as the activation function, while in the research of [15], it is found the ReLU perform better and faster than sigmoid in practice, and Leaky ReLU can overcome the "dead" neuron problem of normal ReLU. Therefore, Leaky ReLU is applied and softmax is used for the final output, which is a widely used activation for multiple class classification.

- For the loss function, the model used the Cross-Entropy, as the model is expected to produce a probability vector, and Cross-Entropy is useful in this case [16]. For the optimizer, Adam is applied mainly because it could accelerate the learning process, save computational power, and escape from local minimum [17].

- Overfitting could lead to low variance, and the performance on the testing set might be affected. To prevent overfitting, the early stop technique is applied [18], when the loss on validation is bouncing from the minimum by 5%, the model will stop earlier. But in practice, the early stop doesn't perform well on RNN, as it fluctuates in a huge range and triggers the early stop at the beginning, causing underfitting. It is also argued that Adam has difficulty with convergence. Therefore, the decay of learning rate is introduced in the RNN [19]. It could slow down the learning rate when it detects the accuracy of validation set is not improved for a period of time. With the learning rate reduced, the model can better converge to the global minimum and avoid being overfitting. That also allows the model to take a larger initial learning rate, which makes the training faster without worrying about the convergence issue.

Table 2. Network topology

	Normal NN	RNN
Layer	3	2
Input Neurons	10	1
Hidden Neurons	10	22
Output Neurons	4	4
Learning Rate	0.001	0.01

2.3 Experiment Design

In this paper, a back-propagation neural network will be built and trained for the prediction task on the marking dataset, in comparison to the RNN on the same condition. The process is repeated 100 times for one run, and both of the models will perform ten runs on training and prediction. The average of the performances is then compared.

3 Result and Discussion

3.1 Result

After proper parameters tuning, the average accuracy on the test set for the normal back-propagation can reach 65.36%, while the RNN model shows a promising improvement on the accuracy, with an accuracy of 72.04%. However, it is found that the epochs needed for training RNN is much more than the normal neural network.

Since the number of epochs needed for each training could be different with the early stop technique, the average trend cannot be aggregated and plotted. Instead, a typical trend of accuracy and loss of the normal back-propagation neural network is shown in Figure 1. It can be seen that the model stops the training right before the rising of the validation loss due to the early stop, which proves that the early stop technique is helpful to properly train the model without being underfitting or overfitting.

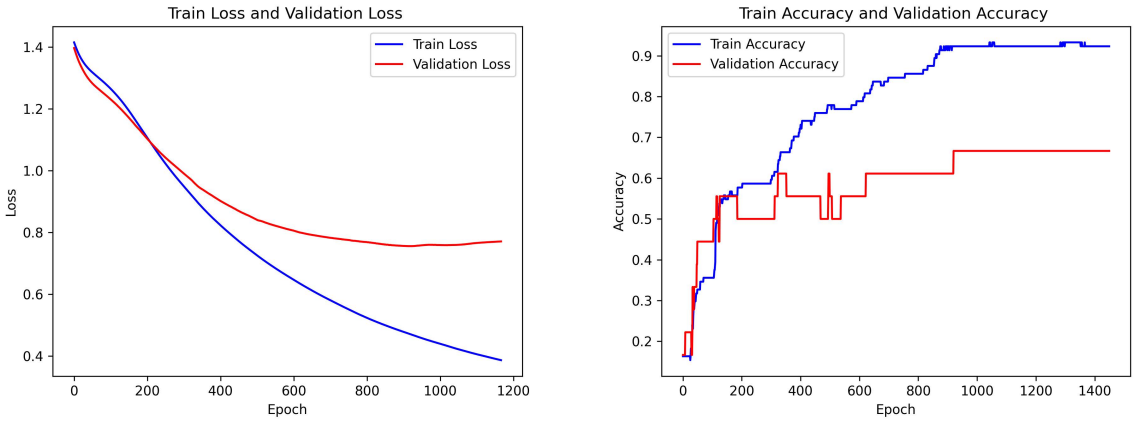

Fig. 4. A typical accuracy and loss tendency in the normal back-propagation neural network training.

Note that the epoch and loss calculation are slightly different from above. The RNN takes sequences and predicts on each one, therefore, the epochs might be much less than above, but in fact, the training for the RNN model is much slower than the normal neural network. Also, the loss is aggregated for each epoch, and the training set with more data points has a much higher loss, compared with the validation set. It could be seen that the model fluctuates in a fairly huge range. The dynamic learning rate is large at the beginning and it helps the model quickly find the approximate global minimum, and by slowing down the learning rate, it helps the model converge to the minimum point accurately. That explains the large fluctuation at the beginning and the small on at the end. In general, it is trained properly and it successfully converged.

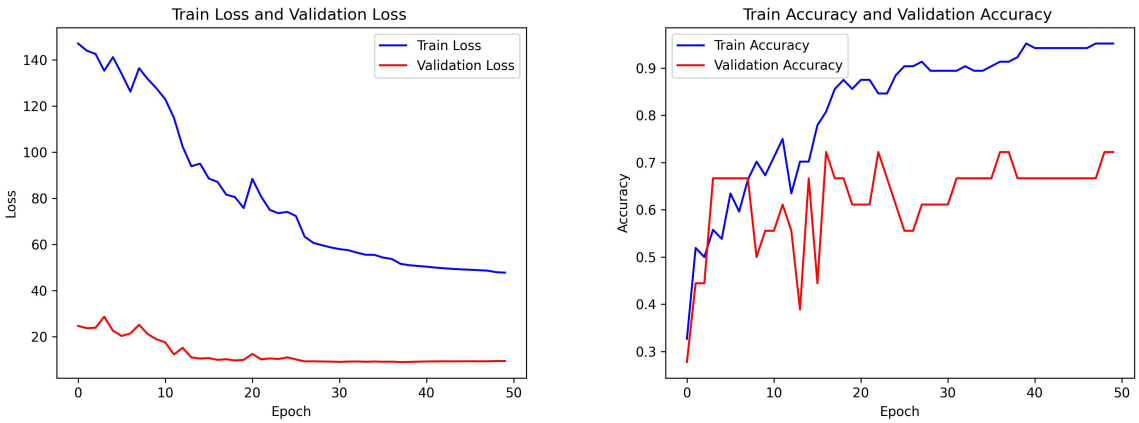

Fig. 5. A typical accuracy and loss tendency in the training with RNN.

Below is the performance comparison between normal neural network and RNN, regarding the accuracy. The training and prediction are repeated for 100 times in one run, and the average accuracy is provided. After ten runs, again the average of the results from the ten runs is calculated and presented. Based on the result, it is obvious that the normal neural network is more stable, while the RNN did have a better performance on this task, holding a 6.68% lead over the normal neural network on average, 7.34% lead in the best case, and 3.61% lead in the worst case.

Table 3. The average accuracy of the Normal NN model

	Train	Validation	Test
1	89.43%	66.39%	65.24%
2	90.70%	65.27%	65.71%
3	**90.66%**	**66.25%**	**67.11%**
4	89.43%	64.76%	64.29%
5	88.91%	66.92%	64.11%
6	91.64%	65.51%	65.53%
7	90.32%	66.14%	65.66%
8	89.64%	65.88%	65.09%
9	87.66%	68.31%	65.98%
10	90.35%	65.63%	64.91%
Average	89.87%	66.10%	**65.36%**

Table 4. The average accuracy of the RNN model

	Train	Validation	Test
1	88.01%	71.02%	73.87%
2	86.53%	74.77%	73.47%
3	84.08%	69.88%	69.28%
4	90.42%	72.68%	73.97%
5	**91.17%**	**77.02%**	**74.45%**
6	88.70%	69.91%	67.72%
7	87.05%	70.19%	72.62%
8	87.67%	72.87%	73.07%
9	88.23%	69.86%	69.96%
10	90.91%	73.72%	71.98%
Average	88.27%	72.19%	**72.04%**

3.2 Discussion

- Generally, the performance of the normal neural network is in accordance with the research [3], which is around 65%. While, the RNN model outperforms the normal neural network model by around 7%, which could be considered as a noticeable improvement.
- The experiment result has proven the RNN works fairly well in early prediction for student's academic performance. One reason could be the relationship between the data points is useful in this task. If a student does not perform well at the very beginning of the semester, but he works hard and keeps improving along with the time, he is more likely to achieve a better result at the end. A simple neuron network model might be incapable of detecting the trend and trapped by the low performance in the early assessment, while RNN has the internal state to capture the dynamic behaviour.
- Combining the result from the previous work with BDR [3], the result also validates the opinion of the research that the outlier in this dataset could be useful, and removing them all is not appropriate. It is actually realistic that the students might have an unstable performance in academic assessment, and lots of human factors could be included.
- It should be noticed that dynamic learning rate is an effective technique to help the model converge, especially when Adam optimizer is used. It also allows the model to accept a larger learning rate at the beginning and accelerate the training and avoid being trapped by the local minimum, and it slows down the learning rate to converge.

3.3 Limitation

- Although this paper proves that the RNN outperforms the normal neural network in this task, the accuracy of 65~75% is still not very high, compared with the research [11]. One reason could be the dataset is not suitable for this task. The size of the dataset is fairly small and the high proportion of missing value could significantly affect the result. It would be better to confirm the research result with another better dataset in the future.
- The dataset might be of low quality. From the previous research [4] with BDR, 28% of the patterns are considered outliers by the BDR algorithm. Although the researchers in [3] reckon some of the outliers might be useful in this dataset, and removing all the outliers is not appropriate, but it could be tricky to identify the invalid pattern from the outlier without human inspection. Not removing the outliers could affect the performance of the models by introducing invalid patterns, which might make the research result slightly less reliable.

4 Conclusion and Future Work

The experiment has shown that the RNN outperforms the normal neural network, in terms of prediction accuracy on this task, while the fact that it spends more time training should also be considered, but in general, the RNN is an effective model for the student academic performance prediction tasks.

Given that the RNN has a promising beginning on this task, more experiments on evaluating other complicated deep neural network could be conducted, such as Bidirectional RNN (BRNN) or Long short-term memory (LSTM). Also, it would be interesting to conduct experiments with the Evolutionary Algorithm such as Genetic Algorithm, to see if it can further improve the performance of the models. As stated on the above analysis of the limitation, the research result might be affected by the low quality of this dataset, it is worth further confirming the result by conducting similar experiments on a better dataset.

References

1. A. Kumar, "ARTIFICIAL NEURAL NETWORK: IN DEPTH", *International Journal for Technological Research in Engineering*, vol. 4, no. 11, 2017. [Accessed 6 May 2020].

2. P. May, H.-C. Ehrlich, and T. Steinke, "*ZIB Structure Prediction Pipeline: Composing a Complex Biological Workflow Through Web Services*," Berlin, Heidelberg, 2006, pp. 1148-1158: Springer Berlin Heidelberg.

3. Slade, P and Gedeon, TD "Bimodal Distribution Removal," in Mira, J, Cabestany, J and Prieto, A, *New Trends in Neural Computation*, pp. 249- 254, Springer Verlag, Lecture Notes in Computer Science, vol. 686, 1993.

4. E. C. Y. Choi and T. D. Gedeon, "Comparison of extracted rules from multiple networks," *Proceedings of ICNN'95 - International Conference on Neural Networks*, Perth, WA, Australia, 1995, pp. 1812-1815 vol.4.

5. T. D. Gedeon and H. S. Turner, "Explaining student grades predicted by a neural network," *Proceedings of 1993 International Conference on Neural Networks (IJCNN-93-Nagoya, Japan)*, Nagoya, Japan, 1993, pp. 609-612 vol.1.

6. S. Geman, E. Bienenstock and R. Doursat, "Neural Networks and the Bias/Variance Dilemma," in *Neural Computation*, vol. 4, no. 1, pp. 1-58, Jan. 1992, doi: 10.1162/neco.1992.4.1.1.

7. J. A. Joines and M. W. White, "Improved generalization using robust cost functions," *[Proceedings 1992] IJCNN International Joint Conference on Neural Networks*, Baltimore, MD, USA, 1992, pp. 911-918 vol.3, doi: 10.1109/IJCNN.1992.227083.

8. R. K. Pearson, "Outliers in process modeling and identification," in *IEEE Transactions on Control Systems Technology*, vol. 10, no. 1, pp. 55-63, Jan. 2002, doi: 10.1109/87.974338.

9. A. Sperduti and A. Starita, "Supervised neural networks for the classification of structures," *IEEE Transactions on Neural Networks*, vol. 8, (3), pp. 714-735, 1997.

10. R. Chandra and M. Zhang, "Cooperative coevolution of Elman recurrent neural networks for chaotic time series prediction," *Neurocomputing*, vol. 86, pp. 116-123, 2012.

11. F. Okubo et al, "A neural network approach for students' performance prediction," in 2017, doi: 10.1145/3027385.3029479.

12. J. Sola and J. Sevilla, "Importance of input data normalization for the application of neural networks to complex industrial problems," *IEEE Transactions on Nuclear Science*, vol. 44, (3), pp. 1464-1468, 1997.

13. S. Wahl et al, "Assessment of predictive performance in incomplete data by combining internal validation and multiple imputation," *BMC Medical Research Methodology*, vol. 16, (1), pp. 144-18, 2016.

14. L. Milne, T. Gedeon, and A. Skidmore, "Classifying Dry Sclerophyll Forest from Augmented Satellite Data: Comparing Neural Network, Decision Tree & Maximum Likelihood". *Proceedings Australian Conference on Neural Networks, 1995.*

15. A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier nonlinearities improve neural network acoustic models," in *Proc. icml*, 2013, vol. 30, no. 1, p. 3.

16.K. Plunkett and J. Elman, *Exercises in rethinking innateness*. MIT Press, 1997.

17.D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization", in *3rd International Conference for Learning Representations*, San Diego, 2015.

18. L. Prechelt, "Early Stopping - But When?," *Neural Networks: Tricks of the trade*. Springer, Berlin, Heidelberg, 1998.

19. M. Zaheer, S. J. Reddi, D. Sachan, S. Kale, and S. Kumar, "Adaptive Methods for Nonconvex Optimization." in *Proceedings of the 32nd International Conference on Neural Information Processing Systems (NIPS'18)*, New York, 2018.

Appendix:

```
      lab2            tutass            lab4             h1
 Min.   :0.000   Min.   :0.000   Min.   :0.500   Min.   : 0.00
 1st Qu.:2.000   1st Qu.:2.000   1st Qu.:2.000   1st Qu.:17.00
 Median :2.500   Median :3.000   Median :2.500   Median :18.00
 Mean   :2.465   Mean   :3.083   Mean   :2.346   Mean   :16.90
 3rd Qu.:3.000   3rd Qu.:4.000   3rd Qu.:3.000   3rd Qu.:19.12
 Max.   :3.000   Max.   :5.000   Max.   :3.000   Max.   :20.00
 NA's   :24      NA's   :8       NA's   :23      NA's   :21
       h2              lab7             p1              f1              mid
 Min.   : 0.50   Min.   :0.000   Min.   : 0.00   Min.   : 0.00   Min.   : 0.6
 1st Qu.: 8.75   1st Qu.:2.000   1st Qu.: 8.50   1st Qu.: 9.00   1st Qu.:13.0
 Median :16.00   Median :2.500   Median :13.00   Median :11.25   Median :19.5
 Mean   :13.67   Mean   :2.325   Mean   :12.66   Mean   :12.14   Mean   :20.3
 3rd Qu.:18.00   3rd Qu.:3.000   3rd Qu.:16.50   3rd Qu.:15.38   3rd Qu.:27.0
 Max.   :20.00   Max.   :3.000   Max.   :20.00   Max.   :20.00   Max.   :43.5
 NA's   :54      NA's   :36      NA's   :40      NA's   :59      NA's   :10
     lab10            final
 Min.   :1.000   Min.   : 2.00
 1st Qu.:2.400   1st Qu.:50.00
 Median :2.400   Median :58.00
 Mean   :2.385   Mean   :56.29
 3rd Qu.:2.400   3rd Qu.:69.00
 Max.   :3.000   Max.   :95.00
 NA's   :42      NA's   :8
```

Fig. 6. More statistics about the dataset.

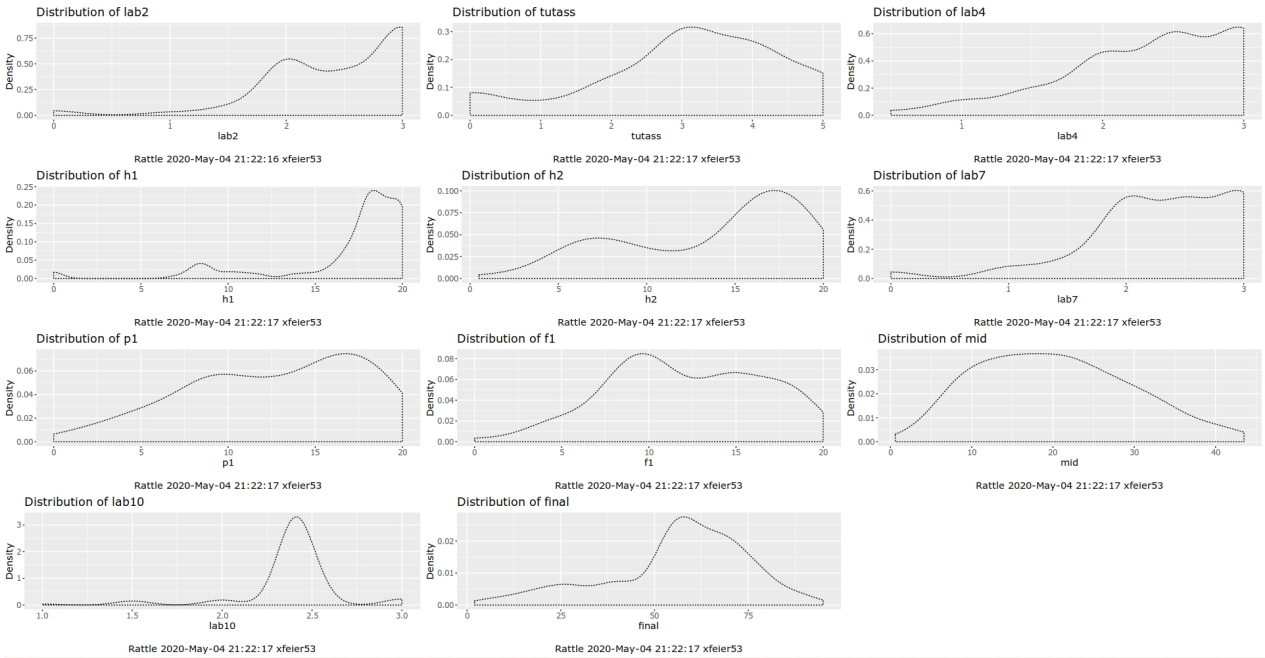

Fig. 7. Distribution of the numeric data