ML4SCIENCE Project Report:
Synthetic data generation with VAE for fairness mitigation

Valentine Delevaux, Julie Le Tallec, Axel Croonenbroek

***Abstract*—Online simulations offer students a self-directed and interactive learning experience. They encourage experimentation, critical thinking, and problem-solving and can be adapted to different learning styles, while providing immediate feedback. Machine Learning can help assessing students' interaction with simulations and therefore identify areas for improvement, ensuring these tools effectively enhance the learning experience and contribute to a more profound understanding of the subjects.**

While classifiers are powerful tools, training them on biased data can lead to unfair predictions. This project focuses on mitigating bias in a classifier by generating synthetic data using a Variational Autoencoder (VAE), to allow a fair prediction of whether a student will succeed in a task or not, based on its interactions with the simulation.

This project was supervised by the ML for Education lab, and specifically by Jade Maï Cock and Richard Lee Davis.

I. Introduction

A. Context of the project

The ML for education lab is currently exploring student's behaviour on a chemistry experiment, with the objective of helping students understanding the Bert-Lambert law of absorbance. The test was performed on 254 French- or German-speaking Swiss students. While experimenting with the simulation, the students could play with different parameters such as the color of a laser or the concentration of a solution, in order to understand the effect of each variable on the absorbance. Subsequently, the students were asked to answer a specific question, based on their understanding of the simulation.

During the experimentation phase, the lab collected all the clicks made by the students, to assess their interactions with the simulation. Students who failed the test task were labeled as 0, whereas the students who succeeded were labeled as 1. The final objective of the lab is to analyse the correlation between the way a student uses the simulation, and their ability to pass the test or not.

After collecting the data the ML for education lab trained a classifier which predicts whether a student will fail the test task or not, based on their clicks during the experimentation phase. Fairness analysis of the predictors across the different demographics (French- and German-speaking) showed that the model was being unfair.[1]

B. Objectives of the project

The main objective of our project was to generate synthetic data from parts of the original data in order to over-sample some demographic categories to increase the fairness of the model. To achieve this goal, we chose to use a Variational Autoencoder (VAE) model and tested and compared several different over-sampling strategies.

A Variational Autoencoder (VAE) is a generative probabilistic model that aims to encode and decode data efficiently while learning a continuous, probabilistic representation of the input data. These models are particularly known for their ability to generate new data samples by sampling from the learned probability distribution in the latent space. They consist of an encoder network that maps input data to a probability distribution in a latent space and a decoder network that reconstructs the input data from samples drawn from this distribution.[2]

II. Dataset

The data set is composed of 254 student profiles. Among other things, each profile contains the demographics of the student (gender and language), their label (0 or 1 for fail or success) and a sequence of clicks, represented by vectors of size 10 (transformed by the lab with previous data preprocessing). One of the first four digits of the vector is set to 1, corresponding to one of four different states. Similarly, one of the six last digits is set to be the time component (time at which the click has occurred), corresponding to a specific type of action. All the other digits are set to 0. Thus, the positions of the two non-zero digits determine the "type" of each click. The number of clicks varies between 20 and 800, with 70% of students under 100.

III. Methods

A. Data preprocessing

Due to the limited number of samples (254 samples), we chose to retain the entire data set, to avoid losing information. Our focus was on data format adjustments to ensure compatibility with our model.

We simplified the data set by converting all non-null values in the click sequences to 1, thereby eliminating the temporal dimension. Hence, the resultant data set consisted of lists of vectors of length 10, characterized by two instances of 1 and eight instances of 0.

Subsequently, we transformed these vectors into one-hot vectors, each corresponding to one out of the 22 possible types of clicks observed in the data set.

Lastly, we changed the structure of the input data to a Numpy array to make it suitable for our VAE model. The

variability in list lengths, as depicted in Figure 1, necessitated the supplementation of certain lists with vectors containing zeros. We tried adding a masking technique to focus the model's attention on relevant data, with the objective to mitigate the risk of information loss associated with excluding longer lists of clicks. Subsequent analysis of the model with masking revealed that this approach was unsuitable for the project, resulting in the rejection of the idea. (see Section III-C1)

As the model was constructed such that both an input data set and a decoder input data set were used in the encoding and decoding processes, we generated the decoder input data set by shifting the sequences of clicks by one position for each sample.

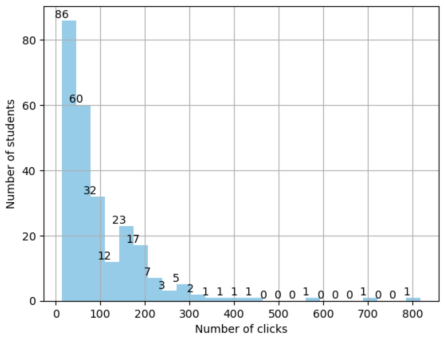


Fig. 1: Distribution of the number of clicks per student

B. Variational Autoencoder Model

To create our VAE, we adapted the code of a Keras implementation of LSTM variational autoencoder, initially designed to generate sentences from a continuous space [3]. This model comprised an autoencoder, mapping the input data to a probabilistic distribution in a lower-dimensional latent space, using mean and standard deviation parameters, and a decoder, reconstructing input data from sampled points in the latent space.

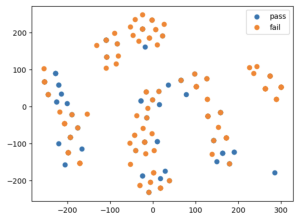
The training objective minimised a loss function, including a reconstruction loss for accuracy and a regularisation term (KL divergence) for a well-behaved latent space. The VAE learned a generative model, resulting in a generator, which is able to generate new data points by sampling from the latent space, using a seed sequence.

In our project, the objective of the generator was to generate new sequences of clicks (sequences of one-hot vectors), from the real input data.

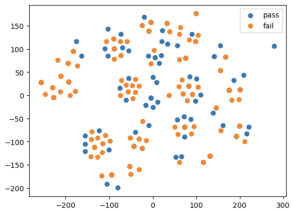
The encoder architecture consisted of an LSTM layer followed by two dense layers. Conversely, the generator architecture comprised one dense layer followed by an LSTM layer.

C. Search of best hyperparameters and model training

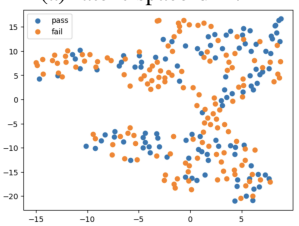
1) Latent Space Visualization: We employed latent space visualization to identify optimal dimensions that would effectively capture variations within our data set. This involved mapping of samples onto the latent space, visualizing the resulting distribution and tracking the accuracy of the VAE reconstructions for each set of parameters. Seeing distinct clusters would ascertain that the model could discern meaningful differences and representations, particularly with regard to the diverse demographics present in our data set. Our goal was to improve the model's capability to capture key features and better understand the underlying patterns in the data. Figure 2d shows the distribution of the initial data set on the latent state, based on the z-mean of each of the points. Grid search on the latent and intermediate dimensions allowed us to select respectively dimensions 2 and 64, without mask, for our final model. Indeed, from Figure 2a it is apparent that with these dimensions, we start observing some structure and clustering in the distribution of the points, according to the performance of the students. On the opposite, as depicted in Figure 2b and Figure 2c, other dimensions or the presence of a mask in the model disturb the organisation of the latent state. Testing our hyper parameters with some fake clustered data showed that the model was able to create a structured latent space as shown in Figure 2d, leading to the conclusion that part of our problem resided in the fact that the real data were not very polarized between the different demographics and labels.



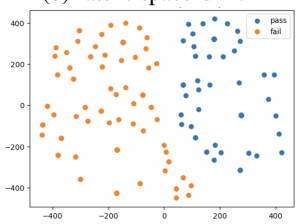
(a) latent space dim. 2



(b) latent space dim. 4



(c) latent space dim. 2, with mask



(d) latent space dim. 2, with fake clustered data

Fig. 2: t-SNE Visualisation of the Latent Space for different hyperparameters and situations

2) Number of epochs: Computing the training and validation loss at every step during model fitting allowed us to choose the best number of epochs. Hence, we optimized the model to increase its ability of reconstructing input data. The results shown in Figure 3 correspond to a VAE model with latent and intermediate dimensions of respectively 2 and 64. They demonstrate an increase in validation loss, illustrating over-fitting in steps after 120 training epochs. This led us to opt for 120 training epochs, corresponding to a reconstruction accuracy of 56%.

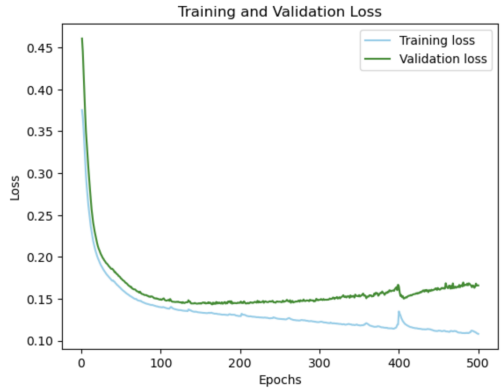


Fig. 3: Evolution of training and validation loss as a function of the number of epochs for the training of the VAE model

D. Generation of synthetic data

In order to generate synthetic data, we used the previously trained generator to create new click sequences, by supplying encoded latent states and seed sequences aligned with the chosen over-sampling strategies (the latent states were implemented with the encoder). Introducing new points in the latent space as the mean of other existing points, allowed to increase the number of generated samples and further diversify the generated sequences. The probabilistic nature of the model introduced inherent randomness, which led to observable differences among the generated sequences. This stochastic element added unpredictability, enriching the synthetic data with variations in the samples inspired by the original data set.

IV. Experiments

A. Different sampling strategies

1) Over-sampling based on performance: The first option to tackle the classifier's fairness problem was to over-sample sequences of clicks, based on the performance of the students on the test task. In light of this, we tested four different sampling strategies.

1) Equal re-balancing : We generated the required number of samples in the minority category to have the same number of "success" and "fail" students. The classifier was then trained on the combination of the original data and generated sequences.
2) 100% synthetic data with balanced subsets: The classifier was trained on only synthetic data, with the same number of "success" and "fail" samples.
3) 100% synthetic data with unbalanced subsets: The classifier was trained on only synthetic data, with the number of "success" and "fail" samples reflecting the proportions in the original data set.
4) 50% synthetic, 50% original data with balanced subsets: The classifier was trained with half original, half synthetic data, with equal proportions of "success" and "fail" performances.

2) Over-sampling based on performance and language: The second option that we considered to generate synthetic data was to split the initial data set into four different subsets, segregating the students based on both their performance and their language (French-success, French-fail, German-success, German-fail). We used the same sampling strategies as above.

1) Equal re-balancing : We generated the required number of samples in the minority categories to have equal numbers of students in the four subsets. The classifier was then trained on the combination of the original data and generated sequences.
2) 100% synthetic data with balanced subsets: The classifier was trained on only synthetic data, with the same number of samples in the four subsets.
3) 100% synthetic data with unbalanced subsets: The classifier was trained on only synthetic data, with the number instances in each categories reflecting the proportions in the initial data set.
4) 50% synthetic, 50% original data with balanced subsets: The classifier was trained with half original, half synthetic data, with equal proportions of each category.

B. Complete model structure

To apply the different sampling strategies, we imported our previously trained and saved model in the existing code provided by the lab, modifying the sub-parts in charge of over-sampling the input data. In addition, we implemented functions providing information about which demographic category to over-sample, along with the required number of new samples to generate. In that way, the code allowed to expand the training input data according to chosen sampling strategy and final data set size, before training and evaluating the classifier (evaluation on real data only). For the sampling strategies that require to choose a fixed final data set size, we opted for 400. As we observed some variation in the predictions of the classifier, we conducted each experiment five times, to have better insights on the result's reproducibility for each sampling strategy.

V. Results

To evaluate each sampling strategy, we compared the performances of the classifier trained with over-sampled input data with its baseline performances without synthetic data augmentation. We analysed these performances based on two characteristics: fairness and accuracy.

A. Fairness

We evaluated the model's fairness by examining the disparity in false positives (FP) between French- and German-speaking students. In the scope of this project, FP is a crucial metric to monitor, as it signifies instances of incorrect classification as "success" (which could lead to not assisting a student experiencing difficulties in understanding the simulation).

As observed in Figure 4, we achieved to reduce the difference between the FP of the French- and German-speaking students by up to 7%. Relatively important standard deviations

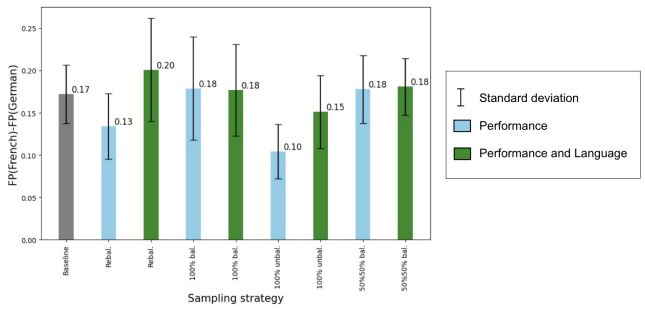


Fig. 4: Average difference in FP between French- and German-speaking students for different sampling strategies, across 5 experiments

illustrate quite variable results. However, it is important to notice that some variability is also present in the baseline, without any data augmentation. Overall, better results are observed with sampling strategies based on only performance. The sampling strategies that seem to most improve fairness are rebalancing and 100% unbalanced synthetic data, based on performance, with FP differences of respectively 13% and 10%.

B. Accuracy

Along with the classifier's fairness, we measured its accuracy using the ROC value, for each sampling strategy. ROC is a metric that enables to measure the trade-off between false positives and false negatives, with higher values (closer to 1) indicating better discrimination between positive and negative instances.

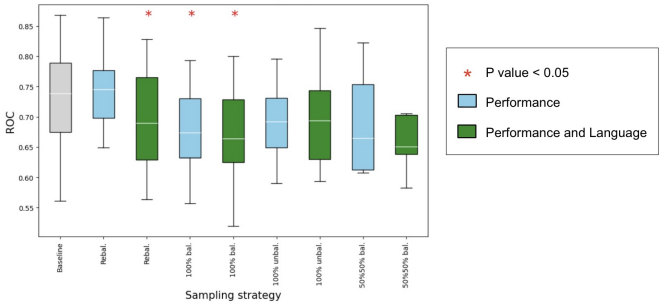


Fig. 5: Average ROC of classifier for different sampling strategies, across 5 experiments

Figure 5 illustrating the ROC values for the different sampling strategies allows to conclude that synthetic data augmentation leads to an overall loss in accuracy. However, this loss is quite small and significant only in three cases, based on a t-test experiment. Oversampling also results in reducing the variability of the performance of the classifier, as compared to the baseline. Specifically, the best performance is achieved for rebalancing based on performance, with a median ROC value of around 0.75, against 0.74 for the baseline. In addition, we achieve ROC values around 0.69 for a 100% synthetic unbalanced dataset.

VI. Discussion

According to our results, the most suited oversampling strategy seems to be rebalancing based on performance, as it shows an important reduction of the false positive difference between French- and German-speaking students. Moreover, this strategy displays good and stable accuracy performances, compared to the baseline and other strategies.

We would have expected strategies taking the language into account for oversampling to be more efficient, as the fairness problem is related to this variable.

However, as the data set is composed of a very limited number of samples, the separation of the samples into four categories taking into account the performance and the language at the same time, leads to very few representative instances of each category for the VAE to train on. This probably leads to reduced understanding of the patterns underlying each category for the model, and may explain why using four categories shows less results.

In addition, the visualisation of the latent space in Section III-C1 reveals a potential area for improvement, to achieve better clustering of the data according to the demographics of each student. These results, combined with a VAE accuracy of 56% could also explain why some oversampling strategies are not as efficient as expected. Under the current conditions, it is questionable whether considering the mean of points in the latent space to generate additional instances is well suited. An alternative worth exploring could involve generating new points, proximal to the input data in the latent space, instead.

Furthermore, we could further investigate the efficiency of the 100% synthetic sampling strategies by testing different sample sizes.

To help further reducing the variability of the classifier, it would also be interesting to take into account the time dimension that was given in the initial data and that we ignored for the sake of simplicity.

Finally, we could also consider "meta-training" the VAE model and more specifically the generator inside of the classifier, to improve its performance directly based on fairness, rather than on input data reconstruction. This process would involve training the VAE and generating synthetic data, using this data to test the classifier and reproduce the experiments to find hyperparameters that most improve the classifier's fairness.

VII. Ethical risks

One of the identified ethical risks of the project is the potential bias amplification through the VAE-based data set augmentation.

Indeed, there is a risk of potential amplification of biases present in the original data set during the augmentation process. In the context of our project, we could imagine a situation in which the initial dataset comprises a majority of

German-speaking students, compared to the number of French-speaking students. Inevitably, the VAE will be trained less on French-speaking students, which could result in deficient understanding of the key features underlying patterns in the sequences of this category of students. As a consequence, the efficiency of the generator regarding French-speaking students would be decreased, perpetuating or increasing existing biases, leading to unfair or discriminatory outcomes.

Stakeholders impacted by this risk include students represented within the data set and those influenced by subsequent classifier decisions. The negative impact involves biased predictions of the classifier, that could result in improper educational decision, i.e., not providing assistance to a student who requires help in understanding the topics covered by the simulation. This may not dramatically change the life of students, but the accumulation of small biases might. The severity of this risk is considerable, as it could significantly impact students' academic life and performances. The likelihood of occurrence is moderate to high, given the inherent risk of amplifying biases within machine learning processes, particularly when learning from biased data sets.

As this project already focuses on ethics and reducing biases across different demographic groups, extensive research into bias mitigation techniques in machine learning was conducted. The most important metric that was measured in order to have an idea of the bias was the disparities in the number of false positives of the classifier.

In response to this risk, several measures were integrated into the project. Augmented data was subjected to fairness testing to ensure minimized biases in classifier outcomes.

To conclude, the ethical risk of bias amplification in data set augmentation through VAEs is significant. While proactive measures were taken to mitigate the risks in the classifier's fairness, complete elimination is challenging due to the intrinsic limitations of the technology and complexities of societal biases.

VIII. Conclusion

We achieved to successfully increase the fairness of the classifier, by augmenting the data set using a pre-trained VAE model. Testing multiple oversampling strategies led to the conclusion that generating synthetic instances to rebalance the number of students in the categories corresponding to "fail" and "success", is currently the best approach.

This augmentation method not only addressed the imbalance in the data set but also notably enhanced the model's ability to make more equitable predictions across various demographic groups, without decreasing its accuracy.

While our current approach has shown promising results, it still presents areas of improvement. Exploring other data generation techniques, further optimizing the Variational Auto-Encoder model or refining existing oversampling strategies might offer even more robust enhancements in fairness metrics.

References

[1] J. M. Cock, M. Marras, C. Giang, and T. Käser, "Generalisable methods for early prediction in interactive simulations for education," 2022. [Online]. Available: https://arxiv.org/abs/2207.01457

[2] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Jozefowicz, and S. Bengio, "Generating sentences from a continuous space," 2015. [Online]. Available: https://arxiv.org/abs/1511.06349

[3] ——, "Generating sentences from a continuous space," 2015. [Online]. Available: https://github.com/alexeyev/Keras-Generating-Sentences-from-a-Continuous-Space