Practical Application of a WGAN-GP Model to Loan Data in R

Seminar Paper

submitted on 30th July 2023

European-University Viadrina
Faculty of Business Administration and Economics
Chair of Finance and Capital Market Theory

Name:	Recep Göktuğ Şengün	128272
	Miguel León Cornejo	128253
	Ovidio Herrera	133123

Course of Studies: Neural Networks in Finance

Academic Supervisor: Dr. Rick Steinert

1. Introduction

In recent years, generative AI models have gained tremendous importance due to their potential applications across various domains. The ability of these models to generate synthetic data that closely resembling real-world data has opened up new possibilities in fields such as image generation, natural language processing, and finance. One prominent development in the field of generative models is the Generative Adversarial Network (GAN) and its subsequent enhancement, the Wasserstein GAN with Gradient Penalty (WGAN-GP). These models have revolutionized the field of artificial intelligence by introducing a novel training mechanism that encourages the generation of close to real-world images.

The focus of this seminar paper lies in the practical application of the WGAN-GP model to loan data. While GANs have traditionally been employed in generating images, this research aims to explore the generative ability of the model outside the realm of images by using loan data. The main objective is to uncover effective ways of tuning the hyperparameters, evaluate the quality of the synthetic data, and ultimately generate high-quality loan data that can be used for various financial analyses and decision-making processes.

The application of a WGAN-GP model to loan data holds great significance as it offers promising insights into the financial sector. By generating synthetic loan data, financial institutions can gain a deeper understanding of customer behavior, risk analysis, and investment strategies. Additionally, the ability to generate high-quality loan data can also have implications for privacy and data security by reducing the need to share sensitive real-world data.

This seminar paper is structured as follows. The second chapter presents the theoretical framework through a literature review, providing an overview of the structure, behavior and evolution of the GAN and its variants WGAN and WGAN-GP. In the third chapter, we apply the WGAN-GP model to loan data, covering descriptive statistics of the loan data, model setup, and analysis of the generated synthetic loan data. The fourth chapter

concludes the research, discussing XXX and potential areas for future improvements. This paper contributes to the knowledge on generative AI models' practical use in the financial sector.

2. Theoretical Framework

2.1. Basic Structure of GAN models

The Generative Adversarial Network (GAN) is a combination of two feed forward neural networks that are continuously learning from each other. Their adversarial nature comes from the fact that both networks have opposing objectives. The first Neural Network, called the discriminator (D), identifies which data is real or fake and the Generator (G) tries to produce synthetic (fake) data as similar to the real data as possible. Their constant interaction allows them to increasingly become better until the synthetic data generated is close to undistinguishable from the real data.

According to (Gulrajani et al., 2017) this interaction can be formally expressed as the minimax objective function:

$$\min_G \max_D \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}\left[\log\big(D(x)\big)\right] + \mathop{\mathbb{E}}_{\tilde{x}\sim\mathbb{P}_g}[\log(1-D(\tilde{x}))]$$

The first part of this function maximizes the probability that the discriminator classifies the data properly, and the second minimizes the probability that the synthetic data produced is classified as fake.

At the beginning of the training process, the generator is fed a noise distribution in order to produce two sets of synthetic data, one labeled as real and the other one as fake. The correctly labeled one is concatenated with the real data to form the training set used to optimize the discriminator. After this has been done, the trained discriminator is fed the other set of synthetic data (labeled as real), subsequently the prediction is used to compute the generator loss and adjust its weight in a matter to maximize the probability of the synthetic data being classifying as real by the discriminator. This process is repeated multiple times until the desired quality of synthetic data is obtained.

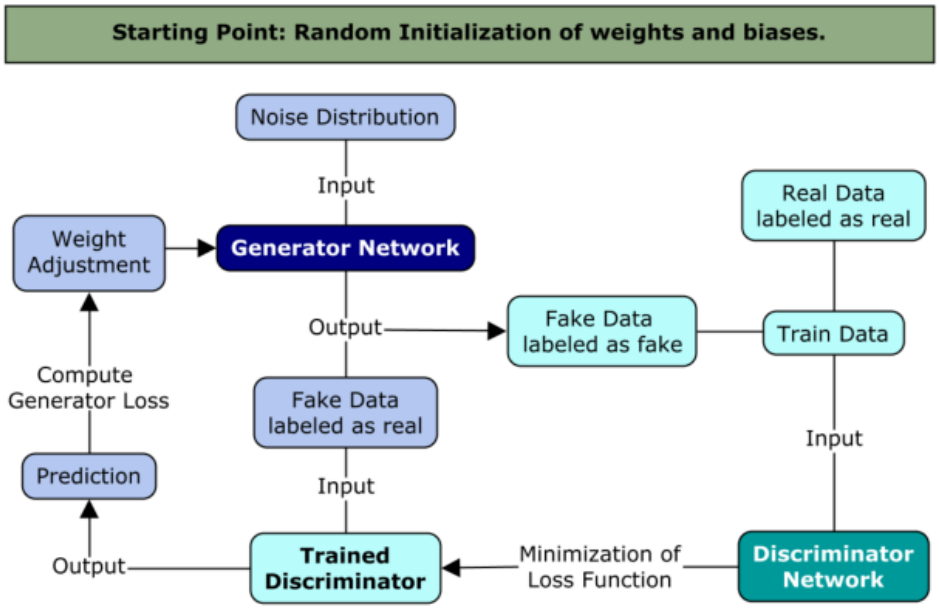

Figure 1: Adversarial Process in the GAN training.

Source: Own elaboration

To train the GAN successfully, the Discriminator and Generator need to have a balanced learning rate. A failure to do so can lead to the following problems:

Mode Collapse: This happens when the GAN generates only the same type of data and it's the result of the Generator being too good at tricking the discriminator, meaning it has learned a specific instance that the discriminator is not able to classify correctly and therefore only reproduces that instance.

Vanishing Gradients: If the discriminator overpowers the generator, there will not be sufficient feedback (Gradient near zero) to properly train the generator, causing its learning process to stop.

In addition, the adversarial nature of the GAN can cause instability in the training. This develops when the discriminator and generator fail to improve together, instead their refinement oscillates. Such a case arises when improvement in either the generator or discriminator causes the deterioration of the other one.

2.2. Development of WGAN-GP

The Wasserstein-GAN was proposed by (Arjovsky et al., 2017) to address the instability of learning in GANs. This is done by replacing the Jensen-Shannon divergence with the Wasserstein-1 distance, which quantifies the minimum cost to transform one distribution into another one. This distance function has the advantage that it is continuous and differentiable almost everywhere. (Arjovsky et al., 2017) argues that the better the discriminator is trained, the more reliable the gradient of the Wasserstein distance will be. Consequently, indicates the value function of the WGAN to be:

$$\max_{\omega \in \mathcal{W}} \mathbb{E}_{x \sim \mathbb{P}_r}[f_\omega(x)] - \mathbb{E}_{z \sim p(z)}[f_\omega(g_\theta(z)]$$

Where W is a set of 1-Lipschitz functions.

Gulrajani(2017) explains that this implies that the Wasserstein distance is also minimized when the value function is minimized under an optimal discriminator, but also that the weight clipping used by (Arjovsky et al., 2017) to enforce the Lipschitz constraint not only fails to avoid exploding or vanishing gradients, but also the discriminator can fail to converge in very deep WGAN networks as well as making the discriminator biased toward much simpler functions.

To avoid all the optimization difficulties caused by weight clipping, (Gulrajani et a al., 2017) proposes an alternative way to enforce the 1-Lipschitz constraint. Here the discriminators gradient is directly constrained by a gradient penalty. The reason for this is that a differentiable function is 1-Lipschitz if and only if it has gradients with norm at most 1 everywhere. Therefore, the value function for the WGAN-GP is:

$$L = \mathop{\mathbb{E}}_{\tilde{x} \sim \mathbb{P}_g}[D(\tilde{x})] - \mathop{\mathbb{E}}_{x \sim \mathbb{P}_r}[D(x)] + \lambda \mathop{\mathbb{E}}_{\hat{x} \sim \mathbb{P}_{\hat{x}}}[(\|\nabla_{\hat{x}} D(\hat{x})\|_2 - 1)^2]$$

It is evident that this is the original discriminator's loss with the addition of a gradient penalty. The further the norm of the gradient deviates from 1, the larger the value added to the loss. This penalty is scaled by the parameter λ.

It is important to note that interpolated data between the distributions is being used to establish the gradient penalty, (Gulrajani et a al., 2017) explains that the motivation behind this is that the optimal discriminator contains straight lines connecting points from both distributions. Therefore, points in these lines can be used to enforce the constraint, given that it is intractable to enforce it everywhere.

2.3. Introduction to oversampling with neural networks for loan data

Logistic regression is often applied to loan data for default prediction, but in order to do a good prediction the logistic model needs to have a balanced dataset, meaning that each category needs to have a similar number of observations. This is not often the case with loan data, where the observations for non-defaulters far exceed the ones of defaulters.

This problem can be addressed in two ways. The first one is with undersampling, which consists in reducing the number of observations of the majority class. The other one is oversampling, where you generate enough data of the minority class to correct the imbalance.

When the degree of imbalance is too large, neither of these two approaches is without disadvantages. Undersampling can possibly result in the exclusion of a substantial amount of information and poses the challenge of deciding which observations from the majority class will be used. Furthermore, with oversampling the problem consists of generating good quality synthetic data from a small sample without consistently repeating the same observations, and this is where the WGAN-GP becomes useful.

The WGAN-GP allows to learn the distribution of the minority class and create new data with the same properties and characteristics as the original data, even under extreme cases of unbalanced data. As a result, a better prediction is obtained from the logarithmic regressions.

2.4. Other GAN applications in Finance

Even though GANs are commonly used in the field of Images, (Eckerli and Osterrieder, 2021) provide a useful summary of different ongoing developments of their use in Finance, among which are in market prediction, tuning of trading models, portfolio management and optimization, and diverse types of fraud detection.

3. Application of WGAN-GP to sample loan data

3.1. Data and methodology

The provided data set contains information about loan data from applicants at the LendingClub. LendingClub is a US based company, founded in 2006, which hosts an online agency platform for peer-to-peer loans, and was the world's largest peer-to-peer lending platform in 2013 (Schumpeter 2013). Until the end of 2015 it reportedly originated approximately $16 billion in loans through its services (Lending Club 2023).

This research utilizes the R software for analysis. The original data set comprises 37 variables, each with 254,191 observations, and for this paper, a pre-written script selects four numeric variables for the WGAN-GP model application: 1) annual_inc: the annual income of the borrower; 2) installment: the monthly payment by the borrower; 3) int_rate: the interest rate on the loan; 4) loan_amnt: the listed amount of the loan.

To gain initial insights into the data's structure, summary statistics are provided for the variables. This is essential to understand specific characteristics of the data, which can be used as stylized facts when evaluating the synthetic data. Notably, all variables exhibit positive skewness, with annual income showing the highest value. Additionally, a Kurtosis of 4.401 indicates a Leptokurtic distribution, suggesting heavy tails and a more peaked central region compared to a normal distribution. This signifies that most borrowers' annual income is concentrated in a small region, with a few outliers having extremely high incomes. The mean annual income is $72.5k, while the maximum value is around $8,706.6k, which further underlines this characteristic.

Table 1: Summary statistics for loan data set

Measure	**loan_amnt**	**int_rate**	**installment**	**annual_inc**
Minimum	500,00	5,32	15,69	3.000,00
Median	12.000,00	13,53	365,23	62.000,00
Mean	13.571,00	13,78	418,27	72.510,00
Maximum	35.000,00	28,99	1.424,57	8.706.582,00
Standard Deviation	8.133,37	4,40	244,91	58.785,37
Skewness	0,83	0,34	1,01	40,76
Kurtosis	0,10	-0,27	0,93	4.401,11

Further analysis reveals the relationships between different variables using a correlation heatmap.

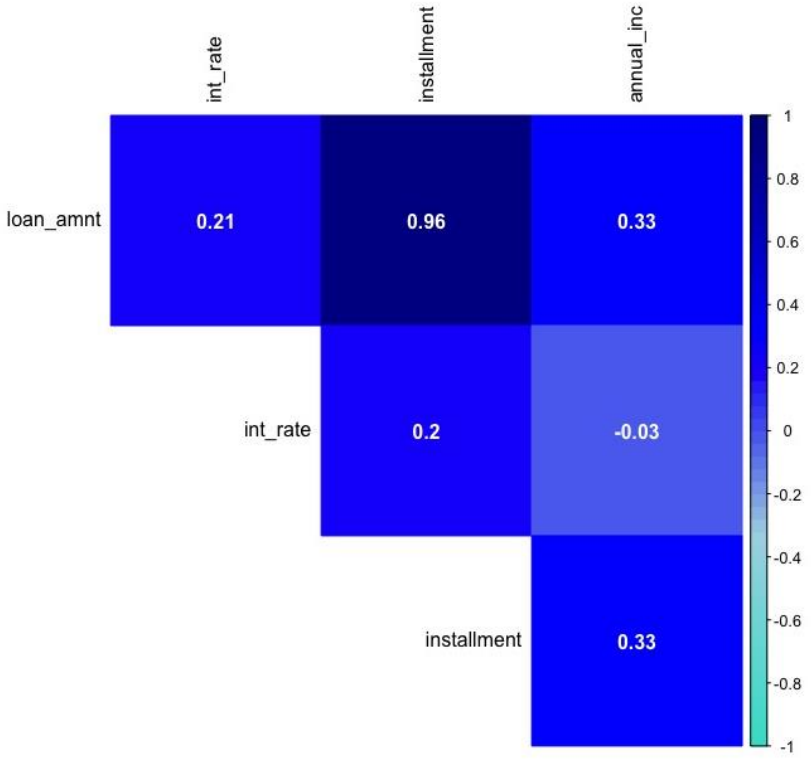

Figure 2: Correlation heat map of the training data set

Source: Own elaboration

The strongest relationship is observed between the loan amount and the installment, displaying an almost perfect positive correlation. This is expected, as a higher loan amount typically requires larger monthly installments for repayment. Moreover, the annual income exhibits a moderate positive correlation with both the loan amount and the installment. This aligns with the intuition that a higher annual income may eliminate the

need for smaller loans and allow for larger installment payments to repay the loan faster. Lastly, the annual income and the interest rate display a weak negative correlation of -0.03, indicating a slight tendency for the variables to move in opposite directions, though the effect is almost negligible. Therefore, as evaluation metrics for the synthetic data the following stylized facts and benchmarks are chosen for this research:

1. All variables are right-skewed, with annual income showing the strongest skewness combined with a high Kurtosis, 40 and 4.401 respectively.
2. The loan amount and installment show an almost perfect positive correlation of 0,96.
3. The annual income shows a moderate positive correlation with the loan amount and installment, of 0,33 each, while having almost no correlation with the interest rate (-0,03).

Prior to model application, variables are standardized to achieve a distribution with a mean of 0 and a standard deviation of 1, eliminating scale differences and enabling distance-based calculations in machine learning.

The model used for this paper is, as the title might suggest, a WGAN-GP which is implemented using the Keras and TensorFlow frameworks. First, hyperparameters such as batch size, number of epochs, discriminator updates per generator update, noise vector dimension, and gradient penalty weight are defined. Next, the generator and discriminator networks are designed as feed-forward neural networks with varying layers and units and leaky ReLu activation functions. Optimizers and loss functions are specified, including the Wasserstein distance for the discriminator and the generator's loss based on the discriminator’s predictions for fake data. A gradient penalty function enforces the Lipschitz constraint. The main training function updates the discriminator and generator models based on real and fake data samples, displaying their losses and progress during training.

To achieve the objectives of this paper, several challenges must be overcome. First, the hyperparameters must be tuned for the model to approach convergence and create

meaningful data. Second, a useful metric must be selected and implemented to evaluate the performance of the WGAN-GP model in creating synthetic loan data. To tackle the first challenge Grid search and Genetic Optimization are selected as approaches for hyperparameter tuning. As of now the only way to evaluate performance of the model is to monitor the total loss of each network and analyze their respective development over time. Although, that measure does not give any insight into the quality of the data relative to the original data set it clearly helps to understand how both networks perform during training with respect to each other.

3.2. Analysis and results

In this section, we analyze the results of applying the WGAN-GP model to sample loan data. Our main goal is to explore techniques for tuning hyperparameters and evaluating the model's performance to generate synthetic loan data resembling the original data. This synthetic data can be utilized for data augmentation and addressing imbalanced classes within the loan data.

In an initial test run the model was trained on the default settings of the hyperparameters to receive a first impression of the model's performance. However, due to the generator's underperformance and the discriminator's dominance, this run was excluded from the analysis. Since this run was conducted before the specification of the evaluation metrics it is not included in the analysis. After the specification of the evaluation metrics as stylized facts 1 to 3 a grid search was conducted using the faculty's computer, with significantly higher computational power. This allowed for an extensive search through all possible hyperparameter combinations. Because of the underperformance of the generator in the initial test run, we decided to reduce the discriminator updates per generator update from 5 to 3 as well as the dimension of the random noise vector for the generator from 100 to 50. Additionally, we reduced the weight of the gradient penalty in the discriminator loss function from 10 to 5. Although, that might seem counterintuitive compared to the other adjustments, computational power combined with the time constraint for this paper, this decision served to facilitate and speed up the training process.

Eight different models were derived from the grid search, along with two more models from additional tests. Table 2 presents the tested models and their respective hyperparameter settings.

Table 2: Hyperparameter settings for training models[1]

Model	Learning Rate	Batch Size	Epochs	Discriminator Steps	Noise Dimension	Gradient Penalty
2	0,001	32	30	3	50	5
4	0,001	64	30	3	50	5
5	0,0001	32	20	3	50	5
6	0,0001	32	30	3	50	5
7	0,0001	64	20	3	50	5
8	0,0001	64	30	3	50	5
9	0,0001	32	30	3	100	5
10	0,0001	128	100	3	50	5

Before evaluating the synthetic loan data table 3 provides additional insight into the training process by showing the accumulated losses of both networks over time. For every model, but especially the models included in the grid search, the discriminator keeps on outperforming the generator by far, except for models 5 and 6.

Table 3: Accumulated losses for training models

Model	2	4	5	6	7	8	9	10
***G* Loss**	41.309	93.469	3.829	9.945	29.951	49.856	-6.931	135.229
***D* Loss**	5.355	4.569	-1.482	-1.949	-5.642	-7.006	-2.294	4.588

Table 4 presents the overall results of the tests and their respective performance compared to the benchmarks from the training data set. Out of the models included in the grid search models 6 and 8 returned the most promising results. Therefore, those two were selected for further hyperparameter tuning. Starting from model 6 the noise dimension was then increased back to 100, resulting in model 9. For model 8 two adjustments were made to explore the effect of larger batch sizes together with more epochs resulting in model 10.

[1] Models 1 and 3 returned #N/A values for the generator and discriminator loss due to unknown reasons and were therefore excluded from this list.

Model 9 clearly performed better than model 10, yet there was not one single model outperforming the others for all of the evaluation metrics. Judging from the weak performance of model 2 and 4 we can conclude that an increased learning rate leads to worse model performance.

Table 4: Stylized facts of the training models[2]

Model	***S* l_a**	***S* i_r**	1 ***S* inst**	***S* a_i**	***K* a_i**	2 ***C* l_a /inst**	3 ***C* a_i /l_a**	***C* a_i /inst**	***C* a_i /i_r**
Benchmark	0,83	0,34	1,01	40,76	4401,11	0,96	0,33	0,33	-0,03
2	0,22	0,11	0,24	1,65	3,45	0,96	0,44	0,27	-0,14
4	0,02	-2,62	-1,74	-0,80	0,24	0,92	0,19	0,05	-0,24
5	0,95	0,25	0,68	8,58	8,58	0,96	0,54	0,57	0,01
6	0,71	0,34	0,63	1,69	4,24	0,95	0,66	0,57	0,04
7	0,73	0,05	0,95	2,06	20,29	0,96	0,57	0,63	0,04
8	0,78	-0,21	0,66	6,16	39,27	0,96	0,36	0,36	-0,22
9	0,82	0,65	1,00	3,53	18,46	0,96	0,40	0,43	-0,10
10	0,35	-0,39	0,60	1,16	1,55	0,97	0,82	0,72	0,02

Additionally, we provide a visual representation of the data distribution through images 3 to 5 comparing model 2, 8, and 9. The difference between the two better performing models 8 and 9 is also visually clearly distinguishable from the weaker performing model 2. The data distribution of the synthetic data does resemble the original data more closely for models 8 and 9 than model 2.

[2] With *S* = Skewness, *K* = Kurtosis, *C* = Correlation coefficient, l_a = loan_amount, i_r = interest_rate, inst = installment and a_i = annual income.

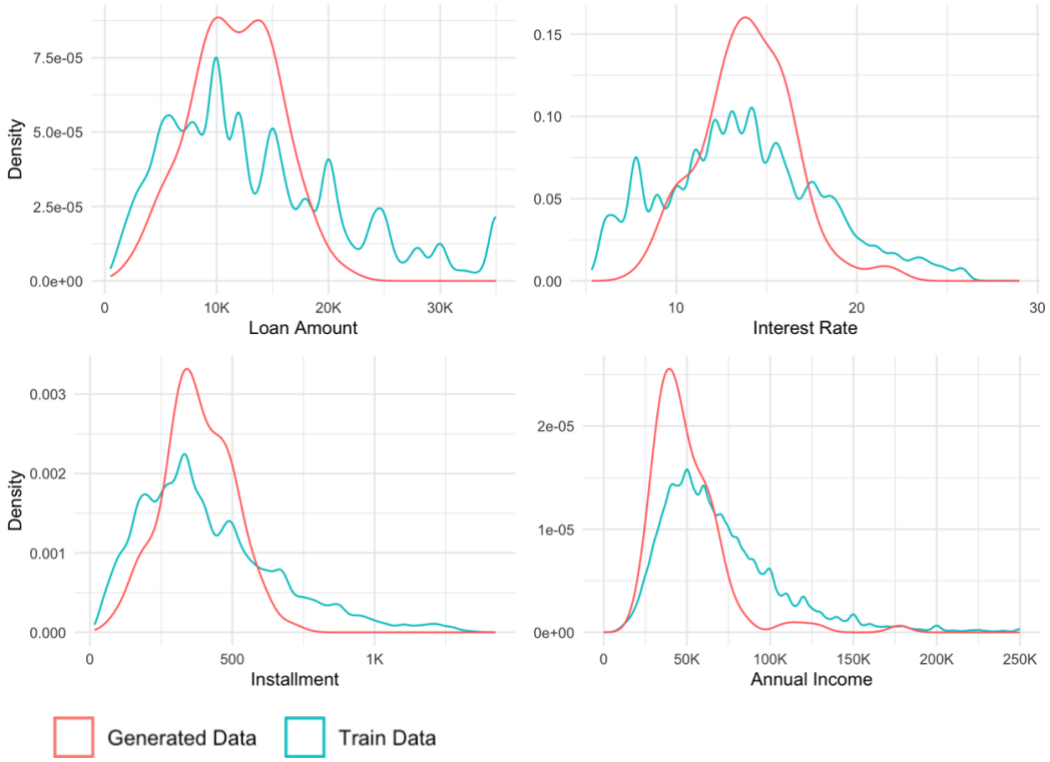

Figure 3: Synthetic data distribution from model 2 vs. training data

Source: Own Elaboration

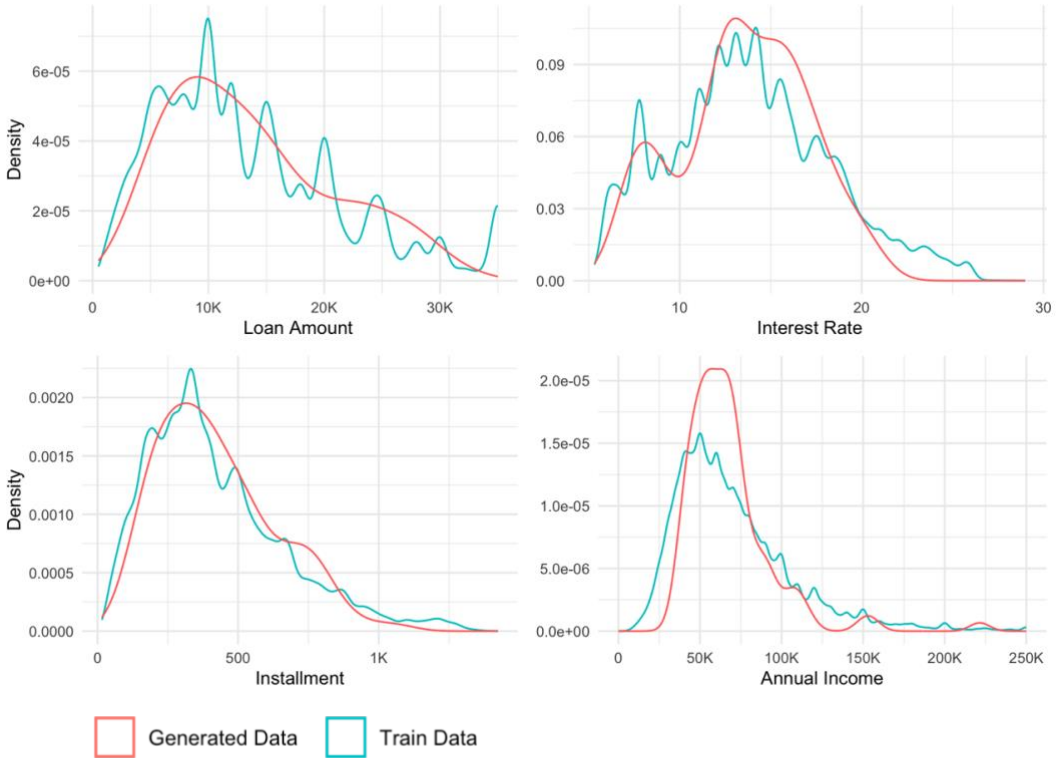

Figure 4: Synthetic data distribution from model 8 vs. training data

Source: Own Elaboration

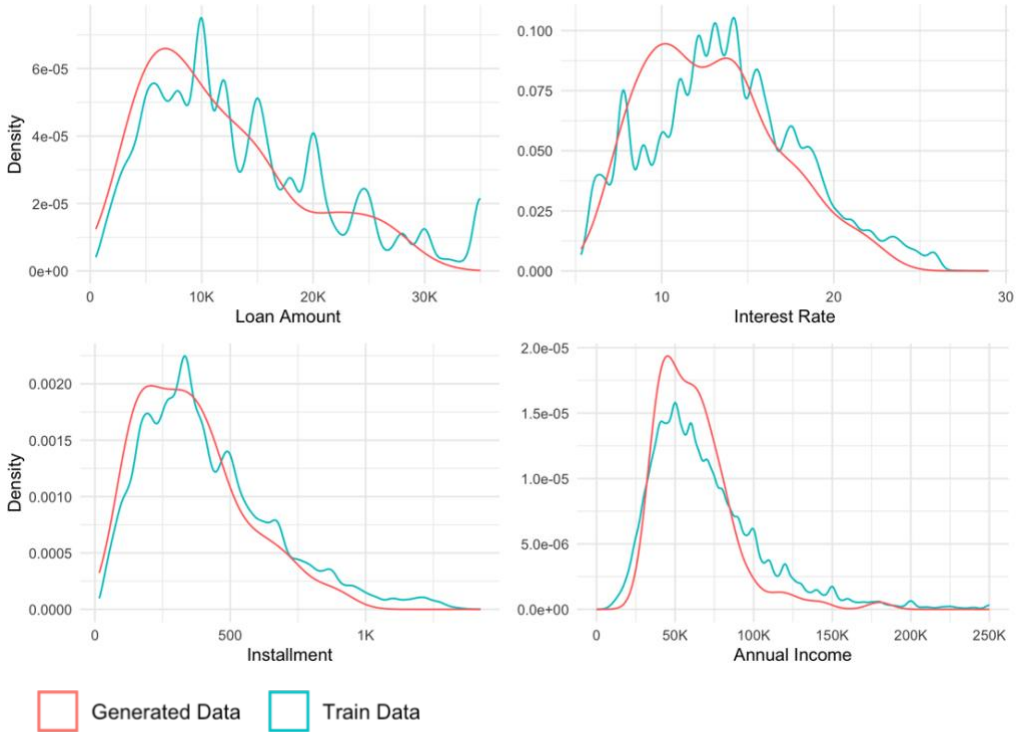

Figure 5: Synthetic data distribution from model 9 vs. training data

Source: Own Elaboration

Concluding the analysis of our results the best performing models, number 8 and 9 share the same settings on all hyperparameters except batch size and noise dimension leading to the conclusion that those hyperparameters leave more possibilities for adjustments without impacting the result too much, if certain boundaries are kept. Model 10 shows that a higher batch number combined with more epochs reduces the performance. The learning rate on the other hand seems to leave only little room for tuning as the performance of model 2 and 4 showed. Overall, epochs and batch size seem to have the highest influence on the models' performance. It is important to mention at this point that the results are to be interpreted very carefully as several issues limited the scope and possibilities of our research. Those will be addressed in the next section.

3.3. Discussion of issues and choices

This chapter highlights the critical issues and important choices encountered during the research, ultimately shaping the outcomes and methodology.

One of the major challenges we faced in this study was the limitation of computational power. The WGAN-GP model demands significant computational resources, and despite efforts to optimize the code, long model run times persisted. Consequently, the research findings were constrained by the computational limitations, restricting hyperparameter tuning in its full potential. Additionally, time constraints substantially impacted the research process of this paper, limiting the exploration of hyperparameters and constraining the study's scope due to extensive WGAN-GP model training and evaluation time. Furthermore, the genetic optimization approach was considered for tuning hyperparameters to enhance the model's performance. However, due to time constraints and the complexity of implementation, we were not able to realize this desired approach.

To handle the mentioned issues on the one hand and still be able to present meaningful results on the other hand the following decisions were made. Due to the prolonged model run times and computational constraints, it was necessary to limit the extent of the research findings. Although various hyperparameters were investigated, not all of them could be explored in the depth desired. Nonetheless, the obtained results provide valuable insights into the model's performance with the selected hyperparameters. To address time limitations as well as the issue regarding the implementation of the genetic optimization approach, manual adjustments were made to certain hyperparameters. Although this approach might not have fully optimized the model, it allowed for reasonable comparisons and analysis within the available timeframe.

In conclusion, the critical challenges of computational limitations and time constraints significantly influenced the research process. Despite these obstacles, we successfully obtained valuable insights into the WGAN-GP model's performance with the selected hyperparameters, allowing for meaningful results and analysis within the available timeframe.

4. Conclusion

In conclusion, this paper aimed to explore the practical application of a WGAN-GP model to loan data and address the challenges associated with tuning hyperparameters and evaluating model performance. The structure of the paper was organized into three key chapters: the theoretical framework and literature review, the application of the model with the analysis of results, and the concluding discussion.

The family of Generative Adversarial Networks can provide good quality synthetic data that allows to do better logistic regression on highly unbalanced data, such as the loan data for default prediction, but it is still open to some training problems. One its most stable training version, the WGAN-GP, has the difficulty of fine-tuning the hyperparameters and requires a lot of computational power to properly train the Network. In addition, the lack of a proven methodology makes the training process more complicated and time intensive.

Through our application of a grid search and additional adjustments we were able to assess the effect of changes in different hyperparameters, finally concluding that epochs and batch size have the most influence on the models performance. Although those results may be not fully representative due to to numerous challenges faced throughout the research.

Despite the constraints, our research sheds light on the potential of WGAN-GP models in generating synthetic loan data. The findings presented here contribute to the broader understanding of generative neural networks and their practical applications in the financial sector. While challenges remain, future research may further explore hyperparameter tuning and optimization techniques to enhance the model's performance and extend its application in the domain of loan data analysis. Overall, this study provides a valuable foundation for advancing the utilization of generative neural networks in the financial industry.

Literature

Arjovsky, Martin / Chintala, Soumith / Bottou, Léon: Wasserstein Generative Adversarial Networks, Proceedings of the International conference on machine learning, PMLR, p. 214 – 223 (2017)

Cohen, Gilad / Giryes, Raja: Generative Adversarial Networks, arXiv: 2203.00667 (2022)

Eckerli, Florian / Osterrieder, Joerg: Generative Adversarial Networks in Finance: An Overview, arXiv: 2106.06364 (2021)

Gulrajani, Ishaan / Ahmed, Faruk / Arjovsky, Martin / Dumoulin, Vincent / Courville, Aaron: Improved Training of Wasserstein GANs, arXiv: 1704.00028v3 (2017)

Lending Club: Lending Club Statistics, https://www.lendingclub.com/info/statistics.action , California (2023), accessed on 18.07.2023.

Schumpeter: Peer Review, The Economist London (2013), accessed on 18.07.2023Schumpeter: Peer Review, The Economist, https://www.economist.com/schumpeter/2013/01/05/peer-review, London (2013), accessed on