

# Addressing Heterogeneity in Federated Learning: Challenges and Solutions for a Shared Production Environment

Tatjana Legler<sup>1,2</sup>  
tatjana.legler@rptu.de

Vinit Hegiste<sup>1</sup>  
vinit.hegiste@rptu.de

Ahmed Anwar<sup>2</sup>  
ahmed.anwar@dfki.de

Martin Ruskowski<sup>1,2</sup>  
martin.ruskowski@dfki.de

<sup>1</sup>University of Kaiserslautern-Landau (RPTU), Germany

<sup>2</sup>German Research Center for Artificial Intelligence (DFKI), Germany

## Abstract

Federated learning (FL) has emerged as a promising approach to training machine learning models across decentralized data sources while preserving data privacy, particularly in manufacturing and shared production environments. However, the presence of data heterogeneity variations in data distribution, quality, and volume across different or clients and production sites, poses significant challenges to the effectiveness and efficiency of FL. This paper provides a comprehensive overview of heterogeneity in FL within the context of manufacturing, detailing the types and sources of heterogeneity, including non-independent and identically distributed (non-IID) data, unbalanced data, variable data quality, and statistical heterogeneity. We discuss the impact of these types of heterogeneity on model training and review current methodologies for mitigating their adverse effects. These methodologies include personalized and customized models, robust aggregation techniques, and client selection techniques. By synthesizing existing research and proposing new strategies, this paper aims to provide insight for effectively managing data heterogeneity in FL, enhancing model robustness, and ensuring fair and efficient training across diverse environments. Future research directions are also identified, highlighting the need for adaptive and scalable solutions to further improve the FL paradigm in the context of Industry 4.0.

## 1 Introduction

Federated Learning (FL) is a collaborative learning approach that enables the training of models across multiple decentralized devices or servers holding local data samples, without exchanging their data [1]. This is achieved by training multiple clients on their local data, computing model updates, and then aggregating (e.g. averaging) them on a central server [2]. By keeping data on local devices and only sharing model updates, FL minimizes the risk of data breaches and preserves user privacy. Techniques such as differential privacy and secure multi-party computation can be applied to further enhance security and privacy [3]. In addition to the aspect of data sovereignty and privacy, employing an FL model circumvents the need for training a new model from scratch at each location, thereby enhancing energy efficiency [4]. Due to its inherent distributed approach, FL is also better able to scale and respond to the failures of individual participants, making it more robust [5]. Moreover, FL models are adept at generalizing across diverse scenarios, further underscoring their practical utility in distributed learning environments. Since the global server lacks information about the clients, heterogeneity such as varying data distributions must be addressed differently than in the centralized case, leading to new challenges and therefore new approaches to solving them. Section 2 initially explores the various types of heterogeneity relevant in the context of FL. Subsequently, Section 3 identifies

those that are particularly significant in the production environment and presents preliminary methods for addressing them.

## 2 State of the Art

First, some different types of heterogeneity are outlined briefly before exploring potential strategies to mitigate them, as these strategies frequently address multiple issues simultaneously. The literature lists various types that are not consistently defined or clearly differentiated, but they generally encompass the following categories [6, 7]:

*Device heterogeneity* in FL describes variations in computational power, network connectivity, and energy constraints or other hardware restrictions among participating clients [8]. This type of heterogeneity challenges the uniform application of methods as clients range from powerful servers to resource-limited mobile and IoT devices. Clients with limited computational capabilities and energy constraints may not perform complex computations or frequent communications, impacting the overall learning process. Additionally, disparities in network connectivity can result in uneven data transmission rates, further complicating model synchronization and convergence [9]. System heterogeneity sometimes also includes model heterogeneity.

*Model heterogeneity* occurs when clients use different model architectures, making it difficult to collaborate on a common model [10]. Traditional FL methods are limited to training models with the same structures, as the simple aggregation of weights is only feasible when each weight has a counterpart in other clients. This hinders applicability in scenarios with different hardware and communication networks, where otherwise the models could be selected according to the available computing power, e.g. smaller and efficient models on edge device and more powerful ones on high performance computers. To overcome this challenge, much more sophisticated approaches than simple averaging are required and often are based on knowledge distillation [11, 12].

The term system heterogeneity often combines device and model heterogeneity. Both types of heterogeneity are crucial when considering mobile devices like smartphones, which operate across diverse hardware configurations supported by a single operating system. It is essential that none of these configurations are excluded in the optimization of a joint model. Similarly, in applications such as data collection from vehicles in preparation for autonomous driving, the hardware may vary significantly [13]. However, the diversity of this data becomes is even more critical to achieve a well generalized model [14].

In dependence on the aforementioned types, *participation heterogeneity* can occur, for example, an unstable internet connection can lead to some clients frequently joining and leaving the FL system, leading to irregular participation [15]. Depending on the volume of new data generated and the selection criteria for participation, some clients may not produce sufficient new data to qualify for participation in a communication round, resulting in a fluctuating frequency, that can affect the consistency and convergence of the global model.

*Data Heterogeneity* refers to variability in data distribution across clients that can lead to biases and affect model performance [16]. Independent and Identically Distributed (IID) variables refer to a sequence of random variables that are statistically independent and follow the same underlying probability distribution [17]. The IID assumption is fundamental in probability theory and statistics, facilitating the modeling of numerous real-world phenomena, such as repeated trials of an experiment or the behavior of a system over time. Although these assumptions are crucial for constructing and validating statistical models, real-world problems seldom exhibit true uniform distribution [18]. However, the definition of non-IID is more varied as there are different ways of deviating from the uniform distribution [1].

There are different types of shifting the relationship between the input (features) and output data (labels) [19]:

- *Covariate shift*: Feature distribution skew

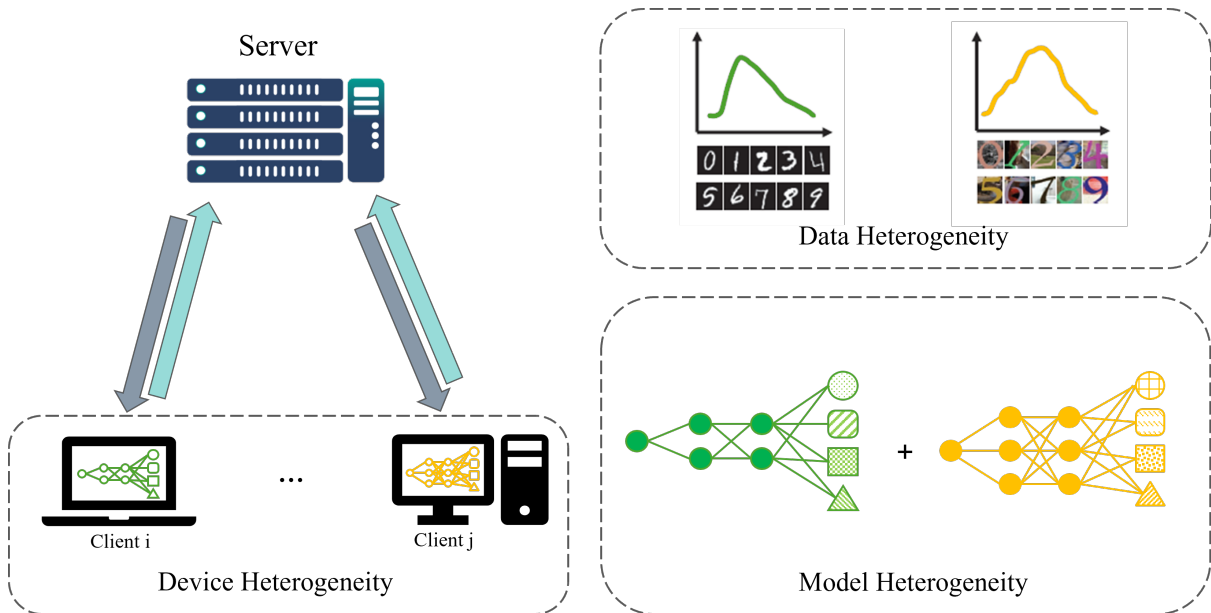


Figure 1: Overview of heterogeneity in Federated Learning systems. The figure illustrates three types of heterogeneity that can affect FL systems: device heterogeneity (differences in computational resources among clients), data heterogeneity (variations in data distributions across clients), and model heterogeneity (differences in model architectures or parameters used by different clients). In parts based on [11]

- *Prior probability shift*: Label distribution skew
- *Concept shift*: Same label, different features or vice versa.

Systems that are actively deployed may encounter temporal shifts in newly generated data over time, diverging from the original training dataset, a phenomenon known as *dataset shift* [20]. Therefore, considerations for continual learning become essential, such as incorporating strategies to learn new information while retaining previously acquired knowledge (i.e., preventing catastrophic forgetting) [21, 11].

Additionally, there might be significant disparities in the volume of data that each client contributes to the global model, often referred to as *quantity skew* or "unbalancedness". Data availability can vary greatly among clients, especially when initializing the system; some may have a large amount of historical data, while others may just have started with data acquisition. Factors previously mentioned concerning participation also influence data contribution. Such imbalances can result in a bias in model training, as clients with larger datasets or multiple clients sharing a common dataset can disproportionately affect the global model. Additionally, such imbalances can cause convergence problems, as training on uneven data distributions may slow convergence or produce suboptimal outcomes, with the global model potentially overfitting to the more data-rich clients. Figure 1 illustrates different types of heterogeneity.

### 3 Addressing Heterogeneity in a Production Environment

As described in preliminary work, we are looking at a shared production scenario in which companies can offer and request services [22]. These services can include hardware services such as milling or drilling of parts, as well as software services like quality inspection solutions as shown in [23]. While we assume that security aspects are covered through the use of a common platform, and none of the participants have malicious ulterior motives (e.g. planning data

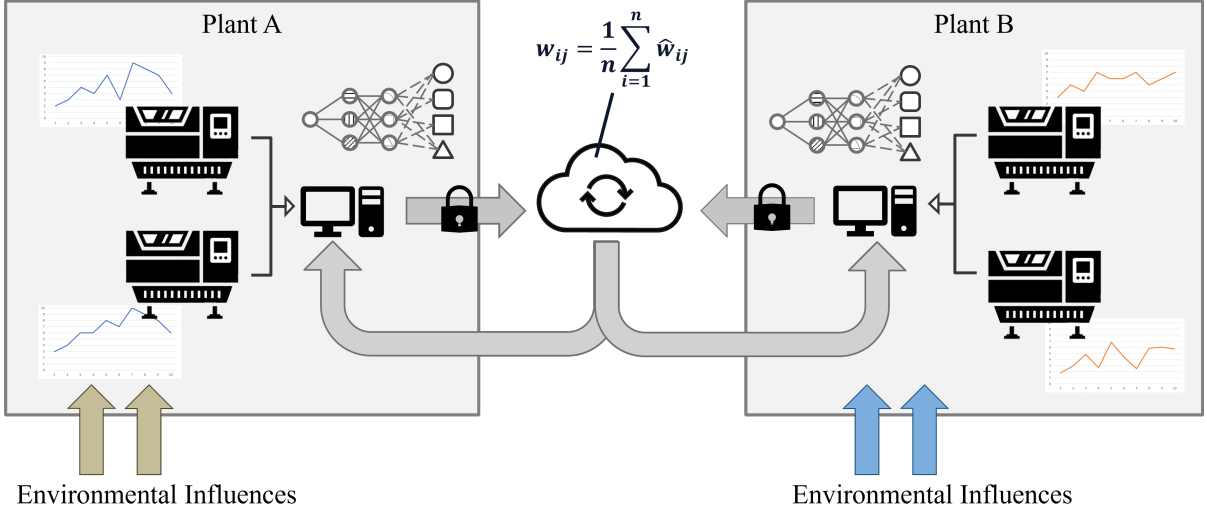


Figure 2: A federated learning system spanning multiple production plants. Only an aggregated model is shared for each plant, therefore enhancing data privacy.

poisoning attacks), all the usual requirements for FL, such as not sharing any production data, still apply [24]. Our assumptions correspond to a cross-silo setting, in which a small number of clients, e.g. from different organizations, tend to participate [25]. In contrast to cross-device, where system and model heterogeneity is also important, organizations or companies are likely to have sufficient computing resources and stable network connections. We therefore focus on data heterogeneity, e.g., handling highly non-i.i.d. data across different clients. The quality and statistical properties of data can significantly vary across clients due to diverse data collection methods, environments, or noise levels. Issues such as noisy data, which may contain errors, irrelevant information, or noise, can degrade model performance. Incomplete data, characterized by missing values or incomplete records, impacts the training process and the resultant model quality. Outliers or extreme values in the data can skew model training, particularly if they are not adequately addressed. Additionally, statistical heterogeneity manifests through variations in the mean and variance of features across clients, posing challenges in data standardization. Differences in correlation structures among different clients can further complicate the model’s ability to generalize effectively across diverse data landscapes.

A tiered (also: hierarchical or grouped) training approach, where clients with similar system capabilities will be grouped and perform a model aggregation within this group before a common model is sent to the next tier or global server [26, 27]. This can make particular sense if a company has many of its own machines or devices that already benefit from the participation of a (local) FL system. Let’s assume, for example, that a machine manufacturer supplies a customer with 100 machines, 20 of which are located in a hall. Although the machines are identical in construction and possibly produce the same part, there are always small deviations due to tolerances. This group of 20 machines will have similar environmental conditions and will probably also be equipped and maintained with the same or similar resources. Creating an FL system on their data alone will lead to a somewhat more stable and better-generalized model, but integrating the other groups will bring in more data diversity and thus unlock more of the potential. Even more of the benefits of FL can be leveraged when the next step is to go beyond corporate boundaries. Many companies keep their production data heavily protected, including how many units of each product have been produced in a given time. The mere possibility of disclosing the number of machines to a competitor can be an exclusion criterion for participation in a system. In such a case, however, the company can implement FL at least within its own locations. Given that data protection regulations across national borders also

affect information flow within a company, this allows various sites to be interconnected (see Figure 2). In a tier-based approach, the number of machines does not need to correlate with the number of FL clients, since even a single model update can be sent to the next higher tier. This has the advantage of disclosing even less information than traditional approaches when participating in the FL system. However, a disadvantage is that the central servers are unaware of the weighting of these model updates and may need to rely on alternative client-selection techniques (e.g., performance-/loss-based).

A more robust aggregation can be achieved by modifying the standard FedAvg algorithm to account for data heterogeneity, such as by weighting updates based on data size or quality. For example, [28] propose Centered Kernel Alignment (CKA) to compute the similarity of feature maps in the output layer and enables fast model aggregation and improves global model accuracy in non-IID scenario .

The initial approach to client selection proposed in [2] was to sample a random fraction of clients for the next communication round. Current client selection techniques investigate which parameters can be used to determine the next participants. Rather than clustering clients into tiers as previously described, clustering approaches can also be employed to temporarily identify clusters and use them to select a client from each. This method ensures a more comprehensive coverage of the entire spectrum. Other approaches utilize training metrics such as local accuracy and training loss to identify and select the worst-performing clients, as these clients have the greatest potential for improvement and can therefore add the most to the global model. The concept of Contribution-Based Selection refers to methods that utilize the impact of a client on the global model as a criterion for selecting clients in subsequent communication rounds. In their work, [29] employ the Shapley value, a concept from cooperative game theory, to estimate each client’s contribution to the global model. Additionally, [30] propose an advancement in contribution estimation by considering both gradient space and data space for individual clients. Generally speaking, the less information that needs to be shared in manufacturing, the better. Therefore, processes that operate solely based on weights or weight changes are preferred.

As the production of defects is highly costly, every company strives to eliminate them as effectively as possible. This naturally results in a significant discrepancy in the dataset, as data with good parts is significantly more likely than data with errors. To counteract this class imbalance, various techniques can be used, especially locally at the client. These include artificially enlarging the data set and changing the weighting of the classes. Synthetic data generation can be used particularly in use cases where there is a high imbalance or where it would be very costly to collect new real data [31, 32]. It can be used to either fill in gaps in the real dataset, by creating new data points [33] or augment the existing dataset [34] to improve generalization. Furthermore, a local resampling of sparse data points can also mitigate the imbalance [35].

## 4 Conclusion

In conclusion, the exploration of heterogeneity in FL within a shared production environment has unveiled a complex landscape of challenges and potential solutions that could significantly advance the field of decentralized machine learning. Our review highlights the crucial need for robust, adaptive methods that can accommodate the unique constraints and characteristics of each client in the federated network. Strategies such as personalized modeling, advanced aggregation techniques, and thoughtful client selection have shown promise in mitigating the adverse effects of heterogeneity, thereby enhancing model performance and fairness across various settings.

Furthermore, the discussion emphasizes the importance of continued research into scalable, flexible solutions that can handle the dynamic and evolving nature of data and system architectures in real-world applications. By fostering a deeper understanding of these issues and

continuously innovating on the solutions, the full potential of FL in industrial and commercial applications can be realized.

In overcoming these barriers, FL will not only improve model accuracy and training efficiency but will also pave the way for more secure, privacy-preserving, and collaborative machine learning endeavors in globally distributed networks. This study sets the stage for future research directions, urging a sustained commitment to addressing these challenges within the landscape of Industrie4.0 and beyond.

## Acknowledgements

This work was funded by the Carl Zeiss Stiftung, Germany under the Sustainable Embedded AI project (P2021-02-009).

## References

- [1] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. Nitin Bhagoji, e. Bonawitz, Advances and open problems in federated learning, *FNT in Machine Learning (Foundations and Trends in Machine Learning)* 14 (1–2) (2021) 1–210.
- [2] B. McMahan, E. Moore, D. Ramage, S. Hampson, B. A. y. Arcas, Communication-efficient learning of deep networks from decentralized data, in: A. Singh, J. Zhu (Eds.), *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, Vol. 54 of *Proceedings of Machine Learning Research*, PMLR, Fort Lauderdale, FL, USA, 2017, pp. 1273–1282.
- [3] V. Mothukuri, R. M. Parizi, S. Pouriye, Y. Huang, A. Dehghantanha, G. Srivastava, A survey on security and privacy of federated learning, *Future Generation Computer Systems* 115 (2021) 619–640.
- [4] L. Liu, J. Zhang, S. H. Song, K. B. Letaief, Client-edge-cloud hierarchical federated learning, in: *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, IEEE, 2020, pp. 1–6.
- [5] H. Xu, J. Li, H. Xiong, H. Lu, Fedmax: Enabling a highly-efficient federated learning framework, in: *2020 IEEE 13th International Conference on Cloud Computing*, IEEE, Piscataway, NJ, 2020, pp. 426–434.
- [6] M. Ye, X. Fang, B. Du, P. C. Yuen, D. Tao, Heterogeneous federated learning: State-of-the-art and research challenges, *ACM Computing Surveys* 56 (3) (2024) 1–44.
- [7] A. M. Abdelmoniem, C.-Y. Ho, P. Papageorgiou, M. Canini, A comprehensive empirical study of heterogeneity in federated learning, *IEEE Internet of Things Journal* 10 (16) (2023) 14071–14083.
- [8] X. Li, Z. Qu, B. Tang, Z. Lu, Fedlga: Toward system-heterogeneity of federated learning via local gradient approximation, *IEEE transactions on cybernetics* 54 (1) (2024) 401–414.
- [9] B. Luo, W. Xiao, S. Wang, J. Huang, L. Tassiulas, Tackling system and statistical heterogeneity for federated learning with adaptive client sampling, in: *IEEE INFOCOM 2022 - IEEE Conference on Computer Communications*, IEEE, 2022, pp. 1739–1748.
- [10] S. Alam, L. Liu, M. Yan, M. Zhang, Fedrolex: Model-heterogeneous federated learning with rolling sub-model extraction, in: S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, A. Oh (Eds.), *Advances in Neural Information Processing Systems*, Vol. 35, Curran Associates, Inc, 2022, pp. 29677–29690.

- [11] W. Huang, M. Ye, B. Du, Learn from others and be yourself in heterogeneous federated learning, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition, IEEE, Piscataway, NJ, 2022, pp. 10133–10143.
- [12] Z. Zhu, J. Hong, J. Zhou, Data-free knowledge distillation for heterogeneous federated learning, in: M. Meila, T. Zhang (Eds.), Proceedings of the 38th International Conference on Machine Learning, Vol. 139 of Proceedings of Machine Learning Research, PMLR, 2021, pp. 12878–12889.
- [13] L. Fantauzzo, E. Fani, D. Caldarola, A. Tavera, F. Cermelli, M. Ciccone, B. Caputo, Feddrive: Generalizing federated learning to semantic segmentation in autonomous driving, in: 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), IEEE, 2022, pp. 11504–11511.
- [14] Y. Li, X. Tao, X. Zhang, J. Liu, J. Xu, Privacy-preserved federated learning for autonomous driving, IEEE Transactions on Intelligent Transportation Systems 23 (7) (2022) 8423–8434.
- [15] P. Pene, W. Liao, W. Yu, Incentive design for heterogeneous client selection: A robust federated learning approach, IEEE Internet of Things Journal 11 (4) (2024) 5939–5950.
- [16] C. Yang, Q. Wang, M. Xu, Z. Chen, K. Bian, Y. Liu, X. Liu, Characterizing impacts of heterogeneity in federated learning upon large-scale smartphone data, in: J. Leskovec (Ed.), Proceedings of the Web Conference 2021, ACM Digital Library, Association for Computing Machinery, New York, NY, United States, 2021, pp. 935–946.
- [17] K. Jacobs, Discrete Stochastics, Springer eBook Collection Mathematics and Statistics, Birkhäuser, Basel, 1992.
- [18] K. Hsieh, A. Phanishayee, O. Mutlu, P. Gibbons, The non-iid data quagmire of decentralized machine learning, in: H. D. III, A. Singh (Eds.), Proceedings of the 37th International Conference on Machine Learning, Vol. 119 of Proceedings of Machine Learning Research, PMLR, 2020, pp. 4387–4398.
- [19] G. D. Y, N. G. Nair, P. Satpathy, J. Christopher, Covariate shift: A review and analysis on classifiers, in: 2019 Global Conference for Advancement in Technology (GCAT), IEEE, 2019, pp. 1–6.
- [20] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. Dillon, B. Lakshminarayanan, J. Snoek, Can you trust your model’s uncertainty? evaluating predictive uncertainty under dataset shift, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems, Vol. 32, Curran Associates, Inc, 2019.
- [21] G. I. Parisi, R. Kemker, J. L. Part, C. Kanan, S. Wermter, Continual lifelong learning with neural networks: A review, Neural networks : the official journal of the International Neural Network Society 113 (2019) 54–71.
- [22] M. Simon, S. Jungbluth, A. Farrukh, M. Volkmann, J. Hermann, M. Ruskowski, Implementation of asset administration shells in a shared production scenario with gaia-x, in: 2023 IEEE 28th International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, 2023, pp. 1–8.
- [23] V. Hegiste, T. Legler, M. Ruskowski, Application of federated machine learning in manufacturing, in: 2022 International Conference on Industry 4.0 Technology (I4Tech), IEEE, 2022, pp. 1–8.

- [24] T. Legler, V. Hegiste, M. Ruskowski, Mapping of newcomer clients in federated learning based on activation strength, 32nd International Conference Flexible Automation and Intelligent Manufacturing (2023).
- [25] C. Huang, J. Huang, X. Liu, Cross-silo federated learning: Challenges and opportunities (2022).
- [26] Z. Chai, A. Ali, S. Zawad, S. Truex, A. Anwar, N. Baracaldo, Y. Zhou, H. Ludwig, F. Yan, Y. Cheng, Tifl: A tier-based federated learning system, in: M. Parashar (Ed.), Proceedings of the 29th International Symposium on High-Performance Parallel and Distributed Computing, ACM Digital Library, Association for Computing Machinery, New York, NY, United States, 2020, pp. 125–136.
- [27] T. Chang, L. Li, M. Wu, W. Yu, X. Wang, C. Xu, Pagroup: Privacy-aware grouping framework for high-performance federated learning, *Journal of Parallel and Distributed Computing* 175 (2023) 37–50.
- [28] Z. Li, T. Ohtsuki, G. Gui, Communication efficient heterogeneous federated learning based on model similarity, in: 2023 IEEE Wireless Communications and Networking Conference (WCNC), IEEE, 2023, pp. 1–5.
- [29] Y. Qiao, M. S. Munir, A. Adhikary, A. D. Raha, C. S. Hong, Cdfed: Contribution-based dynamic federated learning for managing system and statistical heterogeneity, in: NOMS 2023-2023 IEEE/IFIP Network Operations and Management Symposium, IEEE, 2023, pp. 1–5.
- [30] W. Lin, Y. Xu, B. Liu, D. Li, T. Huang, F. Shi, Contribution-based federated learning client selection, *International Journal of Intelligent Systems* 37 (10) (2022) 7235–7260.
- [31] A. Figueira, B. Vaz, Survey on synthetic data generation, evaluation methods and gans, *Mathematics* 10 (15) (2022).
- [32] V. Hegiste, S. Walunj, J. Antony, T. Legler, M. Ruskowski, Enhancing object detection with hybrid dataset in manufacturing environments: Comparing federated learning to conventional techniques, 1st International Conference on Innovative Engineering Sciences and Technological Research (ICIESTR-2024) (2024).
- [33] N. Jaipuria, X. Zhang, R. Bhasin, M. Arafa, P. Chakravarty, S. Shrivastava, S. Manglani, V. N. Murali, Deflating dataset bias using synthetic data augmentation, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), IEEE, 2020, pp. 3344–3353.
- [34] A. Langevin, T. Cody, S. Adams, P. Beling, Generative adversarial networks for data augmentation and transfer in credit card fraud detection, *Journal of the Operational Research Society* 73 (1) (2022) 153–180.
- [35] C. Khosla, B. S. Saini, Enhancing performance of deep learning models with different data augmentation techniques: A survey, in: 2020 International Conference on Intelligent Engineering and Management (ICIEEM), IEEE, 2020, pp. 79–85.