**Project Documentation: Modulation Classification using Advanced Federated Learning**

1. Overview

This project focuses on developing an advanced Federated Learning (FL) model for modulation classification using the radioml2016.10a dataset provided by [deepsig.ai](https://www.deepsig.ai/datasets). The goal is to enhance privacy, security, personalization, and model averaging in FL by incorporating advanced techniques.

1. Techniques Explored

The project explores four advanced FL techniques:

1. Differential Privacy-enabled FL:
   * Reference: [Differential Privacy in FL](https://arxiv.org/pdf/2003.00229.pdf)
   * Source Code: [FL with Differential Privacy](https://github.com/Yangfan-Jiang/Federated-Learning-with-Differential-Privacy)
2. Secure Aggregation-empowered FL:
   * Reference: [Secure Aggregation in TensorFlow](https://medium.com/dropoutlabs/federated-learning-with-secure-aggregation-in-tensorflow-95f2f96ebecd)
   * Paper: [Secure Aggregation in FL](https://eprint.iacr.org/2021/386)
   * Code: [SAFELearn](https://github.com/TRUST-TUDa/SAFELearn)
3. Personalized FL:
   * Reference: [Personalized FL with Non-IID Data](https://arxiv.org/pdf/2103.00710.pdf)
   * Codes: [PFL-Non-IID](https://github.com/TsingZ0/PFL-Non-IID), [PersonalizedFL](https://github.com/microsoft/PersonalizedFL), [Per-FedAvg](https://github.com/KarhouTam/Per-FedAvg)
4. Matched Averaging-enabled FL:
   * Reference: [Matched Averaging in FL](https://arxiv.org/abs/2002.06440)
   * Code: [FedMA](https://github.com/IBM/FedMA)
5. Important Notes

* Combination of Techniques:
  + The project must successfully run and combine at least two techniques (e.g., Differential Privacy and Personalized techniques) into a single FL algorithm.
* Simulation Settings:
  + Simulations should cover various settings, such as running the selected FL algorithm on IID and non-IID data, different numbers of clients, various learning rates, and different levels of Differential Privacy (for the first technique).
* Comparative Analysis:
  + Compare simulation results with the standard FL method.
  + Reference: [Comparative Analysis in FL](https://arxiv.org/abs/2203.09670)

1. Implementation Details
2. Dependencies

* Python
* TensorFlow
* Required libraries for each technique (specified in the respective source code repositories)

1. Running the Code
2. Clone this repository:

bashCopy code

git clone https://github.com/your-username/your-repository.git cd your-repository

1. Follow the instructions in each technique's source code repository to install dependencies and run the code.
2. Combining Techniques

* Modify the FL algorithm to incorporate multiple techniques. Consider compatibility and potential synergies between techniques.

1. Simulation Results

* Thoroughly document and explain simulation results.
* Compare performance under different settings and with the standard FL method.
* Provide insights into why the implemented model is superior.

1. Conclusion

This project demonstrates the implementation and integration of advanced FL techniques for modulation classification. Through extensive simulations and analyses, we aim to contribute insights into the effectiveness of these techniques in enhancing privacy, security, personalization, and model averaging within the FL paradigm. The combination of techniques is expected to provide a more robust and adaptable FL solution for modulation classification tasks.

Feel free to reach out for any questions or clarifications.