Complex-valued Federated Learning with Differential Privacy and MRI Applications

Anneliese Riess¹,², Alexander Ziller², Stefan Kolek³, Daniel Rueckert²,⁴, Julia Schnabel¹,²,⁵, and Georgios Kaissis¹,²,⁴

¹Helmholtz Munich, Germany  
²TUM, Germany  
³LMU Munich, Germany  
⁴Imperial College London, UK  
⁵King’s College London, UK

Abstract

Federated learning enhanced with Differential Privacy (DP) is a powerful privacy-preserving strategy to protect individuals sharing their sensitive data for processing in fields such as medicine and healthcare. Many medical applications, for example magnetic resonance imaging (MRI), rely on complex-valued signal processing techniques for data acquisition and analysis. However, the appropriate application of DP to complex-valued data is still underexplored. To address this issue, from the theoretical side, we introduce the complex-valued Gaussian mechanism, whose behaviour we characterise in terms of $f$-DP, $(\varepsilon, \delta)$-DP and Rényi-DP. Moreover, we generalise the fundamental algorithm DP stochastic gradient descent to complex-valued neural networks and present novel complex-valued neural network primitives compatible with DP. Experimentally, we showcase a proof-of-concept by training federated complex-valued neural networks with DP on a real-world task (MRI pulse sequence classification in $k$-space), yielding excellent utility and privacy. Our results highlight the relevance of combining federated learning with robust privacy-preserving techniques in the MRI context.

Keywords: Federated learning · Differential Privacy · Complex Numbers

1 Introduction

Complex-valued (CV) signal processing is becoming increasingly important in medicine and medical imaging tasks. For example, frequency-domain magnetic resonance imaging (MRI) data is acquired in the CV $k$-space; audio signals from speech or a patient’s heartbeat can be represented as CV spectrograms and wearable or implanted biological sensors produce measurements which can be efficiently represented in the complex field. Moreover, many real-valued problems can be represented and solved in the complex domain, e.g. differential equations, which can be solved more efficiently by first taking their Fourier/Laplace transforms. In many of these examples, privacy preservation is paramount to protect patients and to furnish objective security guarantees, and is often mandated by ethical considerations and legal regulations [JIV19].

Federated learning (FL) has been proposed to enable privacy-preserving data processing in medical imaging. Here, users contribute to training a joint model without sharing their private data, but rather only model updates (e.g. gradients) with a central server that coordinates the training. However, this decentralized approach alone does not suffice to prevent privacy violations, as prior works have shown that FL models are vulnerable to attacks which disclose sensitive information, such as data
reconstruction attacks [BDS+23, FT24, FGC+22]. At the same time, FL does not provide a formal privacy guarantee that can objectively quantify the protection provided by this approach. These remarks underscore that, to be formally privacy-preserving, FL must be complemented by additional privacy technologies. Differential Privacy (DP) [DR+14], a formal framework and set of techniques for deriving insights from sensitive databases while protecting the privacy of individuals who contributed their data, has established itself as the tool of choice in this regard. DP can be thought of as a “contract” between a data owner and a data processor that guarantees that the influence of any individual’s data on the outcome of a computation and –by extension– any harm originating from the release of its results, is limited. However, while DP has been studied extensively in many sub-fields of AI, to our knowledge, it has yet not been sufficiently investigated in the context of CV data processing.

Our Contributions We propose key theoretical and methodological innovations to enable the application of DP in federated CV neural networks (CVNNs). Concretely, we (1) introduce the complex-valued Gaussian mechanism and characterise its privacy properties in Section 3; (2) we generalise the fundamental algorithm of DP deep learning, DP stochastic gradient descent (DP-SGD) to CVNNs in Section 4; (3) we propose novel CVNN primitives (complex GroupNorm and ConjMish, a new activation function) and investigate their properties in Section 4; (4) finally, in a proof-of-concept medical imaging CV FL application in Section 5, we find that that applying our methods yields excellent accuracy.

Related Work CV signal processing workflows have witnessed increasing interest over the past few years. Arguably, biosensing [Pek16] and magnetic resonance imaging analysis [CCPV20, KFH+20, VYL17] are among the most relevant for privacy preservation, and have also seen increasing usage of AI tools. CVNNs have only recently gained significant traction, as automatic differentiation systems have –until recently– not natively supported CV gradients and due to the increased computational expense of CV operations. This has changed with the near-universal adoption of the Wirtinger calculus [KD09, Wir27] in deep learning frameworks, and with the introduction of native (i.e. hardware-optimised) primitives for e.g. convolutions. So far, only a single other study has demonstrated the use of CVNNs in FL [YLC24], and no studies before ours have addressed the biomedical domain or DP applications therein.

DP [DMNS06] has become a standard technique for privacy preservation in AI. Due to space constraints, for a detailed introduction to DP we refer to [DR+14, DRS19]. DP-SGD for real-valued NNs was introduced by [ACG+16]. Only a limited number of studies have examined DP in conjunction with CV data [CSH+22, FX13, FMVH19] or introduced techniques for privacy accounting using CV functional representations [KJH20], however, to our knowledge, none have formalised a general framework to handle DP for CV tasks.

2 Preliminaries

Throughout the paper, we assume a standard FL setup with a central server and several computation nodes, but all introduced techniques apply equally to peer-to-peer FL topologies, swarm learning, etc.. Moreover, we assume all parties to be honest but curious, such that computation nodes perform a local privatisation of their updates before submitting them to the central server; this is not a limitation as our techniques can be readily adapted to all other threat models. As is customary in DP literature, each node holds a set of sensitive records from a universe $\mathcal{X}$, called a database $\mathcal{D}$. From this, an adjacent database $\mathcal{D}'$ can be constructed by adding, removing or replacing the data of a single individual. We assume without loss of generality that individuals are unique throughout the federation. DP is typically realised by first executing a deterministic query function $q$ over the database and then randomising its output by the addition of noise through a DP mechanism $\mathcal{M}$. The noise is calibrated to the query function’s (global) $(\ell_p)$-sensitivity induced by the $p$-norm ($p \in [1, \infty]$), which we denote by $\Delta_p(q)$. As the $p$-norm is also defined for CV vectors, we employ the same strategy to randomise the output of a
CV query. In turn, we introduce the complex Gaussian mechanism (cGM) in Section 3, which serves as the CV counterpart to the Gaussian mechanism (GM), one of the most employed mechanism to achieve DP in real-valued settings.

Every CV function \( q : \mathcal{X} \rightarrow \mathbb{C}^n \) can be written as:

\[
q(D) = q_R(D) + i \cdot q_3(D),
\]

where \( q_R : \mathcal{X} \rightarrow \mathbb{R}^n \) and \( q_3 : \mathcal{X} \rightarrow \mathbb{R}^n \) denote the real and imaginary parts of \( q(D) \), respectively. Representation 1 is useful as many complex functions can be thought of as operators acting on the real and imaginary parts of a complex number separately and then “assembling” the result. However, we caution against equating CV networks to real-valued networks with two “channels” if the appropriate CV operations are not used, as this discards the information within the relationship between the real and imaginary part. Other differences between \( \mathbb{C} \) and \( \mathbb{R}^2 \) call for distinctive treatment when handling tasks in \( \mathbb{C} \). For instance, since the complex plane does not admit a natural ordering, the minimisation of CV functions is not defined. Hence, CVNNs use complex-to-real loss functions. Moreover, to obtain correct gradients for optimisation, we employ the Wirtinger calculus [KD09], which provides a CV gradient operator for real-valued loss functions in CVNNs. In particular, this serves as a base for our proposed CV counterpart to DP-SGD.

3 Theoretical Results

To characterise the privacy properties of CV mechanisms, we utilise the \( f \)-DP framework [DRS19]. Relying on statistical hypothesis testing, \( f \)-DP interprets DP through a trade-off function \( T \) between the Type I and Type II statistical errors faced by a membership inference adversary trying to determine whether one of the adjacent databases contains the individual or not. A mechanism \( \mathcal{M} \) satisfies \( f \)-DP if, for all pairs of adjacent databases \( D \) and \( D' \), \( T(q(D), q(D'))(\alpha) \geq f(\alpha) \) holds \( \forall \alpha \in [0,1] \) for some trade-off function \( f \). Gaussian DP (GDP) is a specialisation of \( f \)-DP when the trade-off-function has the form \( G_\mu := T(\mathcal{N}(0,1),\mathcal{N}(\mu, 1)) \). In particular, \( \mathcal{M} \) preserves \( \mu \)-Gaussian DP (\( \mu \)-GDP) if it preserves \( f \)-DP, for \( f(\alpha) = G_\mu(\alpha) = \Phi(\Phi^{-1}(1 - \alpha) - \mu) \), where \( \alpha \) is the Type-I statistical error and \( \Phi \) is the cumulative distribution function of the standard, real-valued normal distribution. In this light, we introduce our key CV additive noise mechanism:

**Definition 1** (Complex Gaussian mechanism). Let \( q : \mathcal{X} \rightarrow \mathbb{C}^n, D \in \mathcal{X}, \) and \( \psi \sim \mathcal{N}_\mathbb{C}(0, \Gamma, C) \) denote the complex Gaussian distribution with location parameter \( \mu = 0 \in \mathbb{C}^n \), covariance matrix \( \Gamma \in \mathbb{C}^{n \times n} \) and relation matrix \( C \in \mathbb{C}^{n \times n} \). Then, the complex Gaussian mechanism (cGM) is defined as:

\[
\mathcal{M}(D) = q(D) + \psi.
\]

We will consider the cGM when \( \psi \sim \mathcal{N}_\mathbb{C}(0, 2\sigma^2 \mathbf{I}_n, 2i\gamma \mathbf{I}_n) \). The cGM has variance \( \sigma^2 \) in the real and imaginary part of each coordinate and total variance \( 2\sigma^2 \) per coordinate, as \( \text{Var}(z) = \text{Var}(\Re(z)) + \text{Var}(\Im(z)) \) holds for any random variable in \( \mathbb{C} \). Moreover, the cGM marginals are non-circular complex Gaussian distributions whose real and imaginary components are correlated with correlation coefficient \( \rho = \frac{\Delta_2}{\sqrt{\sigma^2(1 - \rho^2)}}. \) Whenever \( \rho = 0 \), we observe a special case: the circular cGM, whose marginals are circular complex Gaussian distributions. In turn, its real and imaginary components are i.i.d. scalar real-valued Gaussian distributions \( \mathcal{N}(0, \sigma^2) \). We next characterize the privacy properties of the cGM:

**Theorem 1.** Let \( \mathcal{M} \) be the cGM with correlation coefficient \( \rho \neq 1 \) acting on a query function \( q \). Then, \( \mathcal{M} \) satisfies \( \mu \)-GDP with:

\[
\mu = \sqrt{\frac{\Delta_2(q)^2}{\sigma^2(1 - \rho^2)} + \frac{2|\rho|}{\sigma^2(1 - \rho^2)} \cdot \Delta_2(q_R) \cdot \Delta_2(q_3)}.
\]

**Proof.** The proof of Theorem 1 can be found in Appendix A. \( \square \)
Note that $3$ is monotonically increasing in $|\rho|$. Specifically, for $|\rho| \to 1$, $\mu \to \infty$ and the cGM becomes \textit{blatantly non-private} We next turn to the circular case:

\textbf{Corollary 1.} The circular cGM acting on $q$ satisfies $\mu$-GDP with $\mu = \Delta_2(q)/\sigma$.

\textit{Proof.} Follows from Theorem 1 by setting $\rho = 0$. \hfill $\Box$

Since the cGM (including the circular special case) satisfies $\mu$-GDP, it inherits all of the properties of GDP, i.e. resilience to post-processing, group privacy, subsampling and composition. Additionally, one can provide $(\epsilon, \delta)$ and Rényi-DP guarantees using the techniques presented in [DRS19, Mir17]. These results allow one to leverage available privacy accounting tools for real-valued DP to design DP workflows for CV tasks as demonstrated in Section 5.

Interestingly, Corollary 1 shows that choosing $\rho \neq 0$ can never improve the privacy guarantee of the cGM. Moreover, observe that the mean squared error (MSE) between $z \in \mathbb{C}$ and its perturbed version $z + \psi$, where $\psi$ is a CV random variable drawn from a zero-centered distribution, satisfies:

$$\text{MSE} (z, z + \psi) = \mathbb{E} \left( ||z - (z + \psi)||_2^2 \right) = \text{Var}(\psi) = \text{Var}(\Re(\psi)) + \text{Var}(\Im(\psi)).$$

Hence, the MSE is independent of the correlation between the real and imaginary components of the CV noise, and consequently, there is no benefit from using correlated complex noise in terms of the introduced distortion. The circular cGM is thus in this sense \textit{optimal} in terms of its privacy-utility trade-off.

\section{4 Training CVNNs with DP}

\textbf{ζ-DP-SGD} Real valued DP-SGD [ACG+16] is a key technique to train deep NNs with DP. Recall that the key steps of DP-SGD are (1) clipping the $\ell_2$-norm of the \textit{per-sample} gradients to a pre-defined threshold, and (2) adding (real-valued) Gaussian noise calibrated to this threshold. Then, each training step leads to the release of a privatised gradient which is used to update the local node’s weights, or is shared with the central server, e.g. for aggregation. To generalize DP-SGD to CVNNs and enable their federated training, we next introduce ζ-DP-SGD, presented in Algorithm 1. Recall from Section 2 that a complex-to-real loss function $\mathcal{L} : \mathbb{C}^n \mapsto \mathbb{R}^1$ is minimised in CVNNs. Using the Wirtinger calculus, we clip the $\ell_2$-norm of the \textit{per-sample conjugate gradient} [KD09], which represents the direction of steepest ascent in this setting:

$$\nabla \mathcal{L} := 2 \left( \frac{\partial \mathcal{L}}{\partial \theta_1}, \ldots, \frac{\partial \mathcal{L}}{\partial \theta_n} \right),$$

where $\overline{\theta} = (\overline{\theta}_1, \ldots, \overline{\theta}_n)$ is the conjugate weight vector. The conjugate gradient is twice the conjugate Wirtinger derivative with respect to the weights [KD09], which results in parity with the real-valued case in terms of the effective learning rate. In particular, using the theoretical results in the previous section, each step of ζ-DP-SGD satisfies GDP, enabling us to utilise the composition and subsampling theorems of [DRS19] to account for the total privacy cost of training a CVNN.

\textbf{CVNN Primitives for ζ-DP-SGD Training} Many CVNN components such as complex convolutional and linear layers as well as split (e.g. CReLU [TBZ+17]) and fully complex (e.g. Cardioid [VYL17]) activation functions are compatible with ζ-DP-SGD. However, Batch Normalisation (BN) [IS15] and its CV implementation [TBZ+17] are prohibited in DP as they “contaminate” the activations with information from other samples in the batch, leading to undefined \textit{per-sample} (conjugate) gradients, which are required for a correct implementation.

To address this issue, BN is typically replaced with Group Normalisation (GN) [WH18] in DP NNs. Since a CV implementation of the GN layer is missing, we next introduce a novel CV GN layer. Recall that, while real vectors are normalised by subtracting the mean and dividing by the variance, in complex vectors the covariance between the real and imaginary components must also be
Algorithm 1 ζ-DP-SGD

Require: Database with samples \( \{z_1, \ldots, z_N\} \in \mathbb{C}^n \), neural network with loss function \( \mathcal{L} \) and weight vector \( \theta \in \mathbb{C}^n \). Hyperparameters: learning rate \( \eta_i \), noise variance \( \sigma^2 \), sampling probability \( p = \frac{R}{N} \), gradient norm bound \( B \), total steps \( T \).

Initialize \( \theta_0 \) randomly

for \( t \in [T] \) do

Draw a batch \( L_t \) with sampling probability \( p \) (e.g. using Poisson sampling)

Compute per-sample conjugate gradient

For each \( i \in L_t \), compute \( \tilde{g}_t(z_i) \leftarrow \nabla \mathcal{L}(\theta_t, z_i) \)

Clip conjugate gradient

\( \bar{g}_t(z_i) \leftarrow \frac{\tilde{g}_t(z_i)}{\max\left(1, \frac{\|\tilde{g}_t(z_i)\|}{B}\right)} \)

Apply the circular complex Gaussian Mechanism and average

\( \hat{g}_t \leftarrow \frac{1}{|L_t|} \left( \sum_{i \in L_t} \bar{g}_t(z_i) + \mathcal{N}_C(0, 2B^2\sigma^2I_n, 0) \right) \)

Descend

\( \theta_{t+1} \leftarrow \theta_t - \eta_t \hat{g}_t \)

end for

Output updated neural network weight vector \( \theta_T \) and compute the privacy cost.

---

considered. We address this by grouping the activations, and then whitening them group-wise. Similar to [TBZ+17], we initialise the affine parameters of the GN layer to \( \gamma = \frac{1}{\sqrt{d}} + \frac{1}{\sqrt{d}} i \) and to \( \beta = 0 \). An implementation of the whitening algorithm and of the GN layer can be found in Listing 1. Of note, the same approach can be used for Layer, Instance or weight normalisation [SK16], as our implementation is differentiable.

Listing 1 PyTorch implementations of complex GN.

def whiten_single(vec):
    flat_vector = vec.flatten()
    centered = flat_vector - flat_vector.mean()  # subtract mean to center the tensor
    stacked = torch.stack((centered.real, centered.imag))
    cov = torch.cov(stacked)  # compute covariance between real and imaginary.
    w_mat, lmbda, _ = torch.svd(cov)  # Compute 1/sqrt. of covariance matrix.
    x = (torch.matmul(w_mat, stacked))
    result = torch.matmul(w_mat.T, x)
    return result

whiten_group = vmap(vmap(whiten_single))  # vmap over batch and group axis

class ComplexGroupNorm2d(nn.Module):
    def __init__(self, num_groups, num_channels):
        ...  # initialise gamma and beta

    def forward(self, x):
        group_shape = (-1, self.num_groups, self.num_channels // self.num_groups) + x.shape[2:]
        x = x.reshape(group_shape)  # split into groups
        x = x.reshape((-1, self.num_channels, ) + group_shape[3:])  # reshape to original shape
        x = x * gamma + beta  # affine operation
        return x

As an additional contribution, we introduce a novel CV Mish activation function. Recall that the real-valued Mish [Mis19] is defined as: \( \text{Mish}(x) := x \tanh(\log(e^x + 1)/x) \). For use with CVNNs, we define a conjugate version:

\[
\text{ConjMish}(z) := (1 + i) \text{Mish}(\Re(z)) - (1 - i) \text{Mish}(\Im(z)).
\]

We empirically found ConjMish to drastically improve accuracy by up to 5% over the best previous alternatives (Cardioid [VYL17], ModReLU or CReLU [TBZ+17]). In contrast to Cardioid and CReLU,
ConjMish has both a *magnitude thresholding* effect and “phase non-linearity” effect instead of merely “passing through” the phase. The latter seems to improve NN convergence, and could be of independent interest. We leave a detailed investigation to future work.

5 Experiments

We next demonstrate the experimental evaluation of our framework in the context of training federated CVNNs on a real-life medical dataset, where both, stringent privacy guarantees and high accuracy are desired. We selected the task of automated MRI pulse sequence classification, which is relevant for both, the automated curation of medical images for AI applications and for image retrieval tasks in clinical routine. Recent works have tackled this challenge using both supervised [dMPB+21] and unsupervised [KBGR21] deep learning techniques. Contrary to the aforementioned works, we directly classify the MRI pulse sequence in *k*-space, that is, to *directly* classify the CV frequency-domain MRI data.

We utilised data from the *brain* sub-challenge of the Medical Segmentation Decathlon [ARB+21], consisting of 484 training records and 266 test records, which are partitioned such that one patient is only present in a single dataset. We instantiated an FL simulation using the Flower framework [BTM+20] which we augmented with a customised version of Opacus [YSS+21], and distributed the training records uniformly at random among 11 computation nodes to obtain an i.i.d. FL setting. Moreover, we uniformly distributed 110 randomly selected test records to each computation node to serve as a validation set. The rest of the test set remained at the central server and was used only to compute the final accuracy. Additionally, for comparison, we also trained the CVNN under centralised conditions. From each record, we extracted 20 centre slices for each of the four available pulse sequences: Fluid Attenuation Inversion Recovery (FLAIR), T1-weighted (T1w), T1-weighted with contrast agent (T1wGD) and T2-weighted (T2w). This resulted in a total dataset size of 38720 training and 21280 testing images, which we resized to 32 × 32 pixels, Fast Fourier transformed to simulate *k*-space (where we retained duplicated frequency components to obtain a representation with the same dimensions as the input), and normalised by whitening.

We used the model architecture from [DFAP21] consisting of three convolutional blocks with 32, 64 and 128 filters, CV GN and the ConjMish activation function (see Section 4) as well as average pooling layers. The classification layer of the network consisted of a single linear layer with 128 units. We trained the model using Adaptive Federated Averaging [RCZ+21] for 500 epochs using the NAdam optimiser with a learning rate of 0.0002, which we decayed by $10^\times$ after 300 epochs, a fixed $\ell_2$-norm bound of 1, 16 GN groups, an expected batch size of 24, and a target $\varepsilon \in [1, 3, 5, 10]$ for $\delta = 0.001$. All stated privacy guarantees are “per-patient”, and all nodes participated in every round with one round per step. Table 1 summarises these results across 10 random seeds as well as the results from centralised learning as a reference. We note for completeness that the $\varepsilon$-parameter represents the *privacy budget* in DP, and higher values correspond to worse privacy guarantees for the individuals. Interestingly, the CVNN achieved an accuracy of nearly 90% at an $\varepsilon$-value of 3, with (at most) 1 to 2% of additional performance gained by diminishing the privacy guarantee. This indicates that, in the task we consider, relatively stringent (local) DP guarantees can be achieved in FL practically without any accuracy penalty, even compared to centralised learning.

| $\varepsilon$ | 1     | 3     | 5     | 8     | 10    | $\infty$ |
|--------------|-------|-------|-------|-------|-------|----------|
| FL           | 81.65 ± 1.03 | 87.98 ± 1.46 | 88.46 ± 1.37 | 89.08 ± 1.11 | 89.99 ± 0.70 | 90.12 ± 1.26 |
| CL           | 82.85 ± 1.49 | 89.53 ± 1.89 | 89.31 ± 1.49 | 89.62 ± 1.06 | 90.33 ± 0.59 | 90.89 ± 1.41 |
6 Discussion

In this work, we investigated the application of DP techniques to CVNNs. We theoretically showed that the cGM naturally extends its real-valued counterparts to the complex domain, allowing for efficient privacy accounting. Moreover, we experimentally demonstrated a proof-of-concept for FL with DP in CVNNs and found that DP CVNN training is possible with strong privacy guarantees and excellent utility, a crucial combination in sensitive fields like healthcare.

We foresee several interesting avenues for future work: For one, the communication efficiency of CVNNs in FL is reduced to their nominally higher number of parameters (two real-valued floating point numbers per parameter). Thus, optimising e.g. mixed-precision techniques for such applications could reduce communication overhead. Moreover, network quantisation strategies tailored to CVNNs could further increase efficiency while maintaining high accuracy. Furthermore, we intend to explore additional CV mechanisms, such as the CV Laplace mechanism, in future studies.

In conclusion, we anticipate the adoption of CVNNs to increase in a variety of machine learning tasks through the broader availability of software tools and improved hardware support. In particular, since, in collaborative and federated learning, many such tasks concern sensitive data, we contend that integrating rigorous privacy techniques such as DP is essential for increasing trust by providing formal guarantees of model behaviour.

References

[ACG+16] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In Proceedings of the 2016 ACM SIGSAC conference on computer and communications security, pages 308–318, 2016.

[ARB+21] Michela Antonelli, Annika Reinke, Spyridon Bakas, Keyvan Farahani, Bennett A Landman, Geert Litjens, Bjoern Menze, Olaf Ronneberger, Ronald M Summers, Bram van Ginneken, et al. The medical segmentation decathlon. arXiv preprint arXiv:2106.05735, 2021.

[BDS+23] Franziska Boenisch, Adam Dziedzic, Roei Schuster, Ali Shahin Shamsabadi, Ilia Shumailov, and Nicolas Papernot. When the curious abandon honesty: Federated learning is not private, 2023.

[BTM+20] Daniel J Beutel, Taner Topal, Akhil Mathur, Xinchi Qiu, Javier Fernandez-Marques, Yan Gao, Lorenzo Sani, Hei Li Kwing, Titouan Parcollet, Pedro PB de Gusmão, and Nicholas D Lane. Flower: A friendly federated learning research framework. arXiv preprint arXiv:2007.14390, 2020.

[CCPV20] Elizabeth K Cole, Joseph Y Cheng, John M Pauly, and Shreya S Vasanaawala. Analysis of deep complex-valued convolutional neural networks for MRI reconstruction. arXiv preprint arXiv:2004.01738, 2020.

[CSH+22] Antoine Chatelain, Vincent Schellekens, Florimond Houssiau, Yves-Alexandre De Montjoye, Laurent Jacques, and Rémi Gribonval. Compressive learning with privacy guarantees. Information and Inference: A Journal of the IMA, 11(1):251–305, 2022.

[DFAP21] Friedrich Dörmann, Osvald Frisk, Lars Nørvang Andersen, and Christian Fischer Pedersen. Not all noise is accounted equally: How differentially private learning benefits from large sampling rates. In 2021 IEEE 31st International Workshop on Machine Learning for Signal Processing (MLSP), pages 1–6. IEEE, 2021.

[DMNS06] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In Theory of cryptography conference, pages 265–284. Springer, 2006.
[dMPB+21] Jean Pablo Vieira de Mello, Thiago M. Paixao, Rodrigo Berriel, Mauricio Reyes, Claudine Badue, Alberto F. De Souza, and Thiago Oliveira-Santos. Deep learning-based type identification of volumetric MRI sequences. In 2020 25th International Conference on Pattern Recognition (ICPR). IEEE, January 2021.

[DR+14] Cynthia Dwork, Aaron Roth, et al. The algorithmic foundations of differential privacy. Found. Trends Theor. Comput. Sci., 9(3-4):211–407, 2014.

[DRS19] Jinshuo Dong, Aaron Roth, and Weijie J Su. Gaussian differential privacy. arXiv preprint arXiv:1905.02383, 2019.

[FGC+22] Liam Fowl, Jonas Geiping, Wojtek Czaja, Micah Goldblum, and Tom Goldstein. Robbing the fed: Directly obtaining private data in federated learning with modified models, 2022.

[FMVH19] Ferdinando Fioretto, Terrence WK Mak, and Pascal Van Hentenryck. Differential privacy for power grid obfuscation. IEEE Transactions on Smart Grid, 11(2):1356–1366, 2019.

[FT24] Shanglun Feng and Florian Tramèr. Privacy backdoors: Stealing data with corrupted pretrained models, 2024.

[FX13] L. Fan and L. Xiong. Adaptively sharing real-time aggregate with differential privacy. IEEE Transactions on Knowledge and Data Engineering (TKDE), 26(9):2094–2106, 2013.

[IS15] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning, pages 448–456. PMLR, 2015.

[JIV19] Anna Jobin, Marcello Ienca, and Effy Vayena. The global landscape of ai ethics guidelines. Nature Machine Intelligence, 1(9):389–399, 2019.

[KBGR21] Turkay Kart, Wenjia Bai, Ben Glocker, and Daniel Rueckert. DeepMCAT: Large-scale deep clustering for medical image categorization. In Deep Generative Models, and Data Augmentation, Labelling, and Imperfections, pages 259–267. Springer International Publishing, 2021.

[KD09] Ken Kreutz-Delgado. The complex gradient operator and the CR-calculus. arXiv preprint arXiv:0906.4835, 2009.

[KFH+20] Thomas Küstner, Niccolo Fuin, Kerstin Hammernik, Aurelien Bustin, Haikun Qi, Reza Hajlosseiny, Pier Giorgio Masci, Radhouene Neji, Daniel Rueckert, René M Botnar, et al. CINENet: deep learning-based 3D cardiac CINE MRI reconstruction with multi-coil complex-valued 4D spatio-temporal convolutions. Scientific reports, 10(1):1–13, 2020.

[KJH20] Antti Koskela, Joonas Jälkö, and Antti Honkela. Computing tight differential privacy guarantees using fft. In International Conference on Artificial Intelligence and Statistics, pages 2560–2569. PMLR, 2020.

[Mir17] Ilya Mironov. Rényi Differential Privacy. 2017 IEEE 30th Computer Security Foundations Symposium (CSF), 2017.

[Mis19] Diganta Misra. Mish: A self regularized non-monotonic activation function. arXiv preprint arXiv:1908.08681, 2019.

[Pek16] Musa Peker. An efficient sleep scoring system based on EEG signal using complex-valued machine learning algorithms. Neurocomputing, 207:165–177, 2016.
A Proof of Theorem 1.

**Theorem 1.** Let $\mathcal{M}$ be the cGM with correlation coefficient $\rho \neq 1$ acting on a query function $q$. Then, $\mathcal{M}$ satisfies $\mu$-GDP with:

$$
\mu = \sqrt{\frac{\Delta_2(q_2)^2}{\sigma^2(1-\rho^2)}} + \frac{2|\rho|}{\sigma^2(1-\rho^2)} \cdot \Delta_2(q_R) \cdot \Delta_2(q_3). 
$$

(3)

**Proof.** Consider a membership inference adversary who observes a mechanism output $y = \mathcal{M}(q(\cdot)) \in \mathbb{C}^n$ and wants to assess whether $y$ originated under $\mathcal{D}$ or $\mathcal{D}'$ based on this single observation. Moreover, assume the adversary is able to conduct a Neyman-Pearson optimal hypothesis test to distinguish $\mathcal{M}(q(\mathcal{D}))$ from $\mathcal{M}(q(\mathcal{D}'))$. The proof is thus reduced to a CV simple vs. simple binary hypothesis testing problem for the location parameter (i.e. mean) of a complex Gaussian distribution with equal covariance and relation matrix. Choosing the likelihood ratio as our test statistic leads to the optimal test design with the hypotheses:

$$
\mathcal{H}_0 : y \sim \mathcal{N}_\mathbb{C}(q(\mathcal{D}), 2\sigma^2 I_n, 2i\gamma I_n) \quad \text{and} \quad \mathcal{H}_1 : y \sim \mathcal{N}_\mathbb{C}(q(\mathcal{D}'), 2\sigma^2 I_n, 2i\gamma I_n).
$$

(7)

Now, we introduce some notation to ease reading of the remaining of the proof. Let $\tilde{z}$ denote the augmented vector constructed from $z \in \mathbb{C}^n$ in the following way:

$$
\tilde{z} = \begin{bmatrix} z & \bar{z} \end{bmatrix} \in \mathbb{C}^{2n},
$$

(8)
where $\bar{z}$ is the element-wise complex conjugate of $z$. Moreover, let $C, \Gamma \in \mathbb{C}^{n \times n}$ be the diagonal matrices $\Gamma = 2\sigma^2 I_n$ and $C = 2i \cdot \gamma I_n$. Then, the probability density functions (PDFs) $f_0(z)$ and $f_1(z)$ under $\mathcal{H}_0$ and $\mathcal{H}_1$, respectively, are:

$$f_0(z) = \frac{1}{\pi^n \sqrt{\det(P) \cdot \det(P)}} \exp \left( -\frac{1}{2} [\bar{z} - q(D)]^H \left[ \begin{array}{cc} \Gamma & C \\ C & \Gamma \end{array} \right]^{-1} [\bar{z} - q(D)] \right),$$

$$f_1(z) = \frac{1}{\pi^n \sqrt{\det(P) \cdot \det(P)}} \exp \left( -\frac{1}{2} [\bar{z} - q(D')]^H \left[ \begin{array}{cc} \Gamma & C \\ C & \Gamma \end{array} \right]^{-1} [\bar{z} - q(D')] \right),$$

where $P = \Gamma - C^H \Gamma^{-1} C \in \mathbb{C}^{n \times n}$, and $H$ denotes the complex conjugate transpose. These probability density functions (PDFs) $f_0(z)$ and $f_1(z)$ under $\mathcal{H}_0$ and $\mathcal{H}_1$, respectively, are used to compute the log likelihood ratio $L = \log \left( \frac{f_1(z)}{f_0(z)} \right)$. It is well-known that, for this problem, the log-likelihood ratio test statistic (i.e., privacy loss random variable) $L$ is (real) Gaussian distributed [SS10]. In particular, the mean of $L$ is $\frac{d}{2}$ under $\mathcal{H}_1$ and $-\frac{d}{2}$ under $\mathcal{H}_0$, and its variance is $d$ under both hypotheses, where $d$ is given by:

$$d = [q(D') - q(D)]^H \left[ \begin{array}{cc} \Gamma & C \\ C & \Gamma \end{array} \right]^{-1} [q(D') - q(D)],$$

(9)

(we refer to Section 7 from [SS10] for more details). Moreover, it is also well-known that that the power of any such a test is monotonically increasing with $d$. Therefore, we seek to maximise $d$ for all $D, D' \in \mathcal{X}$. Due to the specific form of the matrix above (see Equation 9), after some algebraic manipulation, we can compute it explicitly:

$$\left[ \begin{array}{cc} \Gamma & C \\ C & \Gamma \end{array} \right]^{-1} = \left[ \begin{array}{cc} \frac{\sigma^2}{2(\sigma^2 - \gamma)} I_n & \frac{-i\gamma}{\sigma^2 - \gamma} I_n \\ \frac{-i\gamma}{\sigma^2 - \gamma} I_n & \frac{\sigma^2}{2(\sigma^2 - \gamma)} I_n \end{array} \right].$$

(10)

Using 10, we can rewrite $d$, and obtain:

$$d = \frac{||q(D) - q(D')||_2^2}{\sigma^2(1 - \rho^2)} \pm \frac{2|\rho|}{\sigma^2(1 - \rho^2)} (\Re(q(D) - q(D'))^T (\Im(q(D) - q(D'))).$$

(11)

Without loss of generality, the adversary can choose $\pm \rho$ to obtain the highest $d$. Moreover, using 11, we can compute an upper bound for $d$ employing the sensitivity $\Delta_2(q)$ and the Cauchy-Schwarz inequality:

$$d \leq \frac{\Delta_2(q)^2}{\sigma^2(1 - \rho^2)} + \frac{2|\rho|}{\sigma^2(1 - \rho^2)} ||\Re(q(D) - q(D'))||_2 \cdot ||\Im(q(D) - q(D'))||_2.$$  

(12)

$$d \leq \frac{\Delta_2(q)^2}{\sigma^2(1 - \rho^2)} + \frac{2|\rho|}{\sigma^2(1 - \rho^2)} \Delta_2(q_\mathcal{R}) \Delta_2(q_\mathcal{I}) := \hat{d}.$$  

(13)

No tighter bound than 13 can be computed without making assumptions on the query $q$ or the databases $\mathcal{D}, \mathcal{D}'$. Thus, we can use $d$ to compute the trade-off that bounds the worst-case scenario of the cGM:

$$f_{\sqrt{\hat{d}}}(\alpha) = \Phi \left( \Phi^{-1}(1 - \alpha) - \sqrt{\hat{d}} \right),$$

(14)

where $\alpha$ is the Type-I statistical error and $\Phi$ is the cumulative distribution of the standard, real-valued normal distribution. To conclude, we note that 14 is the trade-off function of a (real-valued) $\sqrt{d}$-GDP mechanism. \hfill \square