Private and interpretable clinical prediction with quantum-inspired tensor train models

Anonymous authors
Paper under double-blind review

Abstract

Publicly available clinical machine learning models pose an underappreciated privacy risk: their parameters or outputs can be exploited to identify patients whose data were used during training. Moreover, this risk is exacerbated by models such as logistic regression (LR), which are typically preferred in clinical settings for their transparency. To assess this empirically, we attack LORIS, a publicly available LR model for immunotherapy response prediction hosted on a U.S. government website. From evaluations through its public interface, we recover the model parameters and identify the training cohort with certainty. More broadly, we design cohort-level membership inference attacks under three levels of adversarial access—binary black-box, continuous black-box, and white-box—and apply them to both LR models and shallow neural networks (NNs) trained on the same task. Our results reveal that even a cohort of 35 patients can be reliably identified within training sets of hundreds to thousands, and that common practices such as cross-validation amplify rather than mitigate this risk. To address these vulnerabilities, we propose a quantum-inspired defense based on tensorizing discretized models into tensor trains (TTs). This representation fully obfuscates model parameters and preserves accuracy, while offering black-box privacy comparably to Differential Privacy. Additionally, the TT representations retain LR interpretability and extend it through efficient computation of marginal and conditional distributions, enabling this richer analysis also for black-box models such as NNs. Our results establish tensorization as a practical, post-hoc foundation for private, interpretable, and effective clinical prediction.

1 Introduction

Machine learning (ML) is increasingly used for clinical prediction but poses critical privacy risks, as models trained on sensitive medical data can inadvertently leak individual information (Fredrikson et al., 2014; Sweeney, 2015). In domains where interpretability is essential, such as clinical prediction, intuitive models like logistic regression (LR) are often preferred, yet they are particularly vulnerable to such attacks. More complex models like neural networks (NNs) are harder to attack, but their complexity also makes it challenging to design strong, accuracy-preserving defenses, leaving them vulnerable.

In this work, we study these vulnerabilities in a relevant real-world setting. We design a cohort-level membership inference attack in which an adversary attempts to determine which patient cohorts were included in the training of a given model. Although this task is easier than individual membership inference—i.e., identifying single patients from the training dataset—in medical settings each public cohort may correspond to a small group of patients collected from a specific hospital or study, whose identification may still reveal sensitive information.

To defend against such attacks while preserving the key benefits of clinical prediction models, we propose a defense based on quantum-inspired tensor network (TN) models, focusing on tensor trains (TTs). Specifically, we learn TT models via TT-RSS (Pareja Monturiol et al., 2025) from model outputs discretized into b bins, thereby compressing the output information available to a black-box adversary while removing parameter-level information not present at the black-box level. The use of discretized outputs is motivated by prior

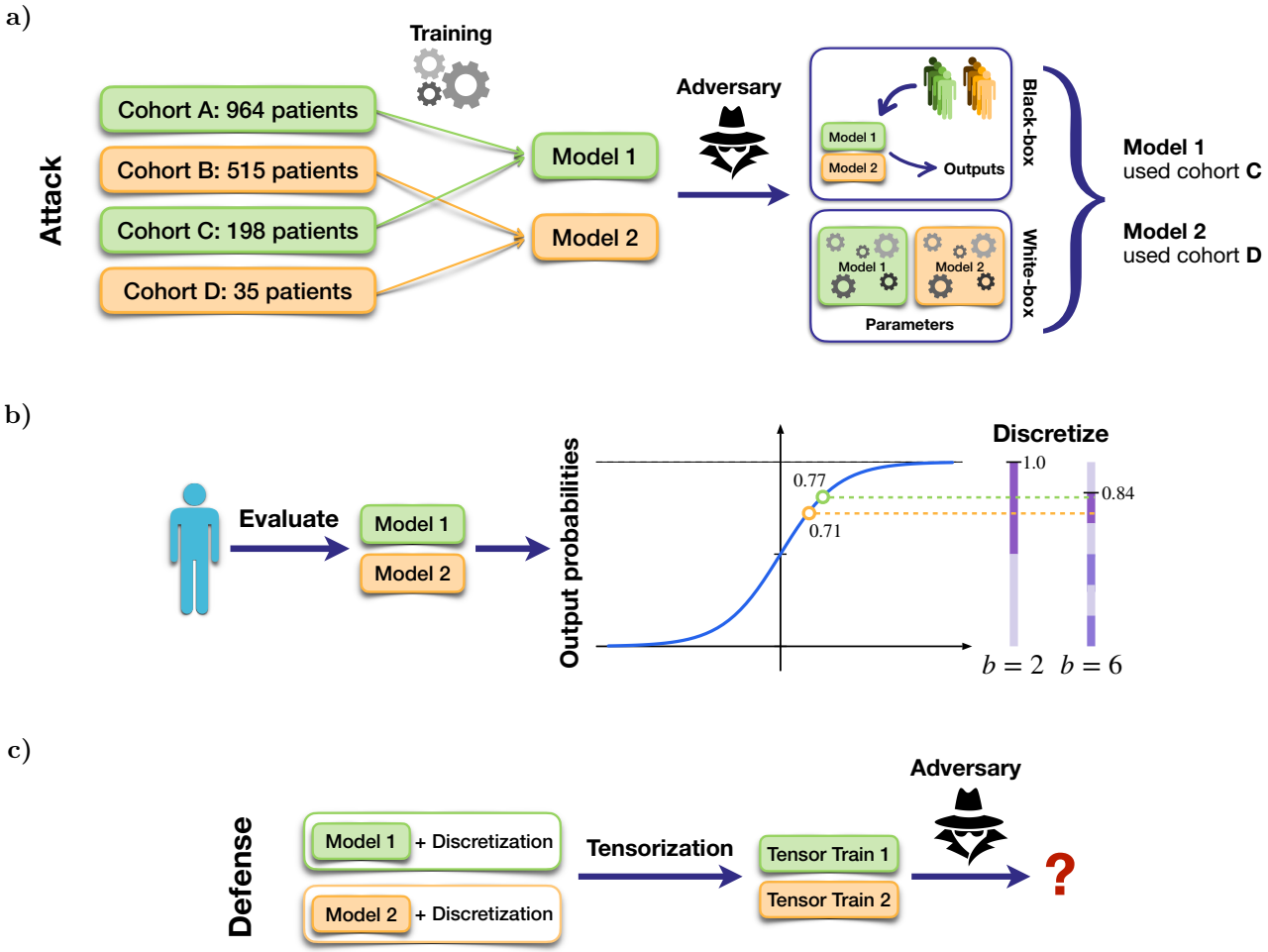

Figure 1: **Overview of the attack and defense setting. (a)** Membership inference attack. Given a set of public patient cohorts, an adversary attempts to determine which cohorts were used to train a target model, exploiting either its output probabilities (black-box access) or its parameters (white-box access). **(b)** Output discretization. Continuous model output probabilities are discretized into b bins (here $b = 2$ and $b = 6$), compressing the output space and reducing the training-data information available to a black-box adversary. **(c)** Tensorization defense. Discretized model evaluations are used to learn a tensor train representation via TT-RSS (Pareja Monturiol et al., 2025), enhancing black-box privacy and removing all parameter-level information not present at the black-box level, thus preventing also white-box attacks.

work showing that compressing a model's output space can reduce membership inference risk (Jia et al., 2019; Yang et al., 2020; 2023; Ye et al., 2022), while the obfuscation of parameter-level information builds on formal white-box privacy guarantees for TNs (Pozas-Kerstjens et al., 2024). Our solution therefore combines output-space compression with parameter-level obfuscation, while retaining predictive accuracy and interpretability. Figure 1 illustrates the full pipeline.

As a case study, we attack LORIS (Chang et al., 2024), a publicly available LR model for prediction of response to immune checkpoint blockade (ICB) immunotherapy. As shown by Chang et al. (2024), LORIS outperforms all other studied models—including NNs—in terms of accuracy and generalization, while also providing direct access to feature importance and producing output scores that grow monotonically with true population level response probability. For these reasons, LORIS was deployed through a public web interface hosted on a U.S. government website for informational purposes[1]. Although our study leverages the

[1]LORIS is available at: `https://loris.ccr.cancer.gov/`.

availability of open-source code and data for LORIS to validate results, we introduce uncertainty regarding the information available to an adversary, allowing the framework to generalize to more restricted settings.

Under this threat model, we design a membership inference attack under binary black-box (bBB), continuous black-box (cBB), and white-box (WB) access, using a shadow model approach that trains multiple models with varied hyperparameters and datasets, followed by an adversarial meta-classifier to predict which public cohorts were included in the training set. To further demonstrate the generality of our tensorization approach, we perform analogous experiments on shallow NNs trained with the same data and objectives. Additionally, we compare the results with Differential Privacy (DP) defenses for both LR and NN models.

Our results show that unprotected models leak substantial training-cohort information, that averaged LR models obtained by repeated cross-validation—a common practice used to obtain the final LORIS model—are more vulnerable than vanilla LRs despite similar predictive performance, and that even the inclusion of a 35-patient cohort can be detected under sufficiently informative access. Regarding our proposed defense, we show that tensorizing discretized models degrades attack performance across all access levels, reducing WB attacks to random guessing while providing BB protection comparable to DP and maintaining predictive accuracy close to the unprotected models. We also find that the discretization parameter b provides control over privacy protection, in a manner analogous to how DP is tuned by adding calibrated noise (Dwork, 2006a). Additionally, TT approximations preserve key properties of LORIS, such as response monotonicity, while enhancing interpretability through efficient computation of marginals and conditionals. This supports feature-sensitivity analysis and enables the construction of cancer-type-specific models without retraining. Importantly, the same techniques can be directly applied to tensorized NNs, providing interpretability for these black-box models.

The remainder of this paper is structured as follows. Section 2 reviews related work and preliminaries on privacy attacks, defenses, and TT models. Section 3 presents the attack setting, target models, tensorization defense, and privacy results. Section 4 analyzes the interpretability of TT models. Finally, Section 5 discusses conclusions, limitations, and future directions.

2 Related work and preliminaries

The widespread adoption of ML systems increases the risk of leaking sensitive personal data. Prior work has extensively examined these vulnerabilities and proposed various defenses.

2.1 Privacy attacks

A wide range of attacks exploit privacy vulnerabilities in ML, leveraging either black-box or white-box access. Key examples include model inversion (Fredrikson et al., 2014), model classification (Ateniese et al., 2015), and membership inference (Shokri et al., 2017), which vary in scope from extracting individual training samples to uncovering global data patterns. Importantly, the level of adversarial access to a model has a substantial impact on the amount of information that can be recovered, with recent work showing that access to model parameters can enable full reconstruction of training samples (Balle et al., 2022; Haim et al., 2022; Oz et al., 2024).

In this work, we adopt the membership inference approach to identify groups of samples present in the training set. More specifically, our attack identifies which public patient cohorts were used for training. This differs from standard individual membership inference, but remains privacy-relevant in clinical settings, where a cohort may correspond to a small study, a rare cancer subtype, or a specific institution.

LR models are particularly exposed in this setting. For LR, reconstruction attacks can yield closed-form solutions (Balle et al., 2022), underscoring how widely used, transparent models can be the most exposed. This motivates our focus on LORIS and on LR-based clinical predictors, while also evaluating whether similar risks extend to shallow NNs trained on the same task.

2.2 Defense mechanisms

Given the diversity of privacy-related attacks, various defense mechanisms have been proposed. Among these, Differential Privacy (DP) (Dwork, 2006b) stands out for its rigorous framework. DP quantifies the likelihood that an attacker can infer whether a specific user's data was included in a statistical process. A randomized algorithm $\mathcal{A}$ is ε-DP if, for any set of outcomes $\mathcal{S}$ in the range of $\mathcal{A}$, it satisfies

$$\log\left(\frac{\mathrm{P}[\mathcal{A}(D)\in\mathcal{S}]}{\mathrm{P}[\mathcal{A}(D')\in\mathcal{S}]}\right)\leq\varepsilon, \tag{1}$$

where D and D' differ by a single element. This metric guides the addition of calibrated noise to achieve a target ε, based on the sensitivity of the function being protected (Dwork, 2006a; Dwork & Roth, 2014). For LR models, common defenses add noise either to the objective or to the final parameters (Chaudhuri et al., 2011); for NNs, the standard approach is DP-SGD (Abadi et al., 2016), which adds noise to gradients at each training step. However, the noise required for meaningful privacy guarantees ($\varepsilon \ll 1$) often degrades performance and may exacerbate group disparities (Bagdasaryan et al., 2019; Hansen et al., 2024). Hence, there is no consensus on how to set ε in a practically meaningful way (Garfinkel et al., 2018): while small values are theoretically ideal, larger values may still prevent attacks in practice without significantly harming accuracy (Ziller et al., 2024).

Beyond DP, recent work has explored whether standard ML practices can improve privacy. Pruning introduces small errors that resemble DP-like protection (Huang et al., 2020), while knowledge transfer reduces dependence on specific training data (Shejwalkar & Houmansadr, 2020). Several works show that non-private models produce overly spread output scores, and that compressing this space, e.g., by injecting crafted noise to prevent membership inference, improves privacy (Jia et al., 2019; Yang et al., 2020; 2023). With a similar goal, other approaches add noise to the output scores, providing DP guarantees with minimal utility loss (Ye et al., 2022; Papernot et al., 2017).

Our approach builds on these ideas: tensorization acts as a knowledge-distillation mechanism that converts a model into an efficient, interpretable representation that preserves WB privacy (Pozas-Kerstjens et al., 2024). To further enhance BB privacy, we apply tensorization to discretized scores that collapse the model's output space into a smaller subdomain. The tensorization mechanism itself is independent of the obfuscation method, so alternative output-obfuscation approaches could similarly be applied before tensorizing to enforce DP. Thus, our approach may resemble work on private low-rank approximation, such as Kapralov & Talwar (2013), but at the level of full-model decomposition, in contrast with techniques that use low-rankness for private fine-tuning (Liu et al., 2025).

2.3 Tensor train models

Tensor networks are low-rank decompositions of high-dimensional tensors with roots in quantum many-body physics. They offer compact, interpretable representations of quantum states (Pérez-García et al., 2007; Orús, 2014; Cirac et al., 2021) and have recently been adapted to machine learning (Stoudenmire & Schwab, 2016; Novikov et al., 2018). TNs have been applied to model compression (Novikov et al., 2015; Tomut et al., 2024), explainable AI (Tangpanitanon et al., 2022; Aizpurua et al., 2024), anomaly detection (Wang et al., 2020), and robustness (Mossi et al., 2025). Importantly, TNs offer formal WB privacy guarantees: multiple parameterizations can represent the same model, effectively obfuscating all but its BB behavior (Pozas-Kerstjens et al., 2024).

Throughout this work, we focus on one-dimensional TNs known as tensor trains (Oseledets, 2011). An order-N tensor $T\in\mathbb{R}^{d^N}$ admits a TT representation with *ranks* r_n if it can be written as

$$T(i_1,\ldots,i_N)=G_1(i_1)\cdots G_N(i_N), \tag{2}$$

where the *cores* G_n are $r_{n-1}\times r_n$ matrices and $r_0=r_N=1$. This structure also supports continuous functions of the form

$$f(x_1, \ldots, x_N) = \sum_{i_1,\ldots,i_N} \mathrm{W}(i_1, \ldots, i_N)\, \phi_1(i_1, x_1) \cdots \phi_N(i_N, x_N), \tag{3}$$

where W is a TT-format coefficient tensor and $\phi_n(i_n, x_n)$ are vector-valued embedding functions indexed by i_n. To ensure non-negative probability scores, it is standard to define distributions via the Born rule: $p(x) = |f(x)|^2$. Further details on TTs, including efficient marginalization and conditioning, are provided in Appendix B.

TTs can be trained using SGD or physics-inspired variants (Stoudenmire & Schwab, 2016). Alternatively, TT representations can be constructed via low-rank decompositions, bypassing high-dimensional optimization. Recent techniques based on sketching (Hur et al., 2023) and cross interpolation (Fernández et al., 2025) achieve this using only function evaluations—i.e., BB access—to approximate continuous functions in TT form. A recent method, TT-RSS, extends this idea to tensorize pre-trained NNs using a small evaluation dataset (Pareja Monturiol et al., 2025). This results in an efficient procedure, requiring $O(|D|^2 Nd)$ model evaluations on a dataset of *pivots* D and an additional $O(|D|^3 Nd)$ to assemble the TT. In this work, we adopt TT-RSS to tensorize models.

3 Privacy analysis

To evaluate the privacy risks of clinical prediction models and compare defense strategies, we design a membership inference attack based on shadow-model training. Assuming an adversary with access to multiple public datasets, the attack determines which of them were used to train a model under varying levels of access. In this section, we first describe the experimental setting and methodology, following the protocol used to evaluate LORIS and the proposed defenses, and then present the privacy results.

3.1 Setting and overview

The attack considered in this study aims to identify which data cohorts, from a set of publicly accessible cohorts, are included in the training set of a given clinical prediction model. To this end, we follow a shadow-model-based approach, in which multiple models are trained under different configurations, with training sets constructed from possible combinations of cohorts. We then train a multi-label meta-classifier to predict the presence or absence of each cohort in the training set, leveraging either the model outputs (bBB or cBB access) or information contained in the model parameters (WB access). We next specify the clinical datasets, target and protected models, attack construction, evaluation metrics, and implementation details used in this analysis.

3.1.1 Clinical datasets

In our study, we evaluate privacy vulnerabilities within the setting of Chang et al. (2024). The underlying task in that work is the prediction of treatment response in cancer patients receiving immune checkpoint blockade (ICB) immunotherapy, using tumor mutational burden (TMB) together with additional clinical, genomic, and pathological variables. Although TMB has been proposed as a biomarker of ICB efficacy, it is not universally predictive, motivating the development of multivariate models that combine TMB with other patient characteristics. Importantly, all patient cohorts considered originate from studies that pursue this same task of predicting binary ICB response.

The datasets used in this setting consist of multiple patient cohorts spanning 18 solid tumor types and up to 18 features, including tumor information, standard clinical variables, and blood-based markers. The represented cancer types include: non-small cell lung (NSCLC), renal, melanoma, head and neck, bladder, sarcoma, gastric, central nervous system (CNS), colorectal, endometrial, hepatobiliary, cervical (CLC), esophageal, pancreatic, mesothelioma, ovarian, breast, and cancers of unknown primary.

To establish a common modeling framework across heterogeneous cohorts, Chang et al. (2024) proposed a six-feature logistic regression model—the LORIS score—based on TMB, Patient's Systemic Therapy History

(PSTH), Albumin, Neutrophil-to-Lymphocyte Ratio (NLR), Age, and Cancer Type. Table 1 summarizes the cohorts and their main characteristics.

Table 1: Summary of the main dataset characteristics, including cohort size, cancer types, number of features provided, and original references.

Dataset	Size	Cancer types	#Features	Reference
Cho1	964	16 solid tumors	18	Chowell et al. (2022)
Cho2	515			
MSK1	453	15 solid tumors	13	Chang et al. (2024)
MSK2	104	CNS / Unkown primary	12	
Shim	198	NSCLC	13	Shim et al. (2020)
Kato	35	8 rare tumors	6	Kato et al. (2020)

3.1.2 Target models and defense mechanisms

As target models, we consider LRs and NNs. Following Chang et al. (2024), we train *averaged* LRs via 20 repetitions of 3-fold cross-validation. While LORIS used larger numbers of repetitions and folds, we found this configuration sufficient to obtain comparable results. For comparison, we also train *vanilla* LRs through a single training run on an 80% split of the corresponding dataset. In both cases, the hyperparameters are solver = "saga", penalty = "elasticnet", class_weight = "balanced", max_iter = 100, l1_ratio $\in \{0, 0.5, 1\}$, and C $\in \{0.1, 1, 10\}$. For NNs, we train 2-layer MLP classifiers following the same procedure as the vanilla LRs, adopting the best hyperparameters reported by Chang et al. (2024): two hidden layers of size 19, binary cross-entropy loss, and Adam optimization for 100 epochs with batch_size = 32, lr = 10^{-3}, and weight_decay = 10^{-5}.

For each trained LR and NN, we build a TT model via the TT-RSS tensorization algorithm (Pareja Monturiol et al., 2025), using 50 random samples from the corresponding training set as pivots. Model evaluations on these pivots are discretized into b bins, with $b \in \{2, 6, 10\}$: values < 0.5 map to the lower bin limit and values > 0.5 to the upper, preserving the property that output probabilities sum to 1. The resulting TTs have $N = 22$ cores (including one for the output), ranks $r = 2$, input dimension $d = 2$, and use polynomial embeddings $\phi(x) = [1, x]$. To accommodate the higher complexity of NNs, we use 80 pivots and ranks $r = 5$. After tensorization, the TT cores are randomized via a gauge transformation, fully obfuscating the parameters and preventing any leakage under WB access. Further details on the TT structure and efficient computations are provided in Appendix B.

Finally, for comparison with a standard privatization approach, we also train DP models (LR-DP, NN-DP) from scratch. Since DP training of LR is restricted to solver = "lbfgs" and penalty = "l2", we fix max_iter = 100 and vary the privacy budget $\varepsilon \in \{0.1, 1, 10, 100\}$, where $\varepsilon = 100$ nearly matches the non-DP case. Only vanilla models are considered, as averaging would cancel the injected noise and effectively increase ε. For NNs, we follow the same training setup as in the non-DP case but apply DP-SGD with max_grad_norm = 1, $\delta = 10^{-4}$, and $\sigma \in \{20, 5, 1, 0\}$, which correspond approximately to privacy budgets $\varepsilon \in \{0.2, 1, 10, \infty\}$. To achieve these budgets, we reduce the number of epochs to 50.

3.1.3 Membership inference attack design

We assume the adversary knows the model architecture and training procedure, up to uncertainty in the specific hyperparameter configurations used. The adversary also has access to the public cohorts $\{C_1, \ldots, C_M\}$ and to models trained on datasets D formed as unions of these cohorts, and has sufficient resources to train shadow models and meta-classifiers.

We distinguish between three levels of adversarial access: binary black-box (bBB), from binary model classifications only; continuous black-box (cBB), from continuous output probabilities; and white-box (WB), from model parameters. The attack proceeds by constructing a dataset of shadow models—LRs, NNs, and TTs as

described above—each trained under different hyperparameter configurations and training sets. From each model, we collect the available model information together with the corresponding cohorts used for training, forming the input to a multi-label classifier that learns to identify the presence of each cohort in the training set. Formally, the attack consists of the following steps:

1. For each hyperparameter configuration and for each possible combination of cohorts in $\mathcal{C}$ forming a dataset D, train 100 shadow models.
2. Build a dataset of (model information, membership label) pairs, where model information is the available representation under the chosen access level—bBB, cBB, or WB—and the membership label is a binary vector whose m-th entry is 1 if cohort $C_m \subset D$ and 0 otherwise.
3. Train an adversarial classifier minimizing independent cross-entropy losses for each C_m, yielding a model that, given model information as input, returns a vector where entry m gives the probability that $C_m \subset D$.

As adversarial classifiers, we use MLP multi-label classifiers with three hidden layers of sizes 32, 16, and 8, and an output layer of size 6 (one per public cohort). The input size depends on the access type. For BB attacks, each shadow model is evaluated on 100 samples—the same samples for all models—drawn randomly from the union of all cohorts; the resulting vector of raw or discretized outputs serves as input to the adversary. For WB attacks we collect full model parameters: for LR, 22 parameters (21 coefficients + intercept); for NN, all per-layer parameters concatenated into an 818-dimensional vector; and for TT, all $N = 22$ cores concatenated into a single vector of 168 (TT-LR) or 1 020 (TT-NN) dimensions. All parameters are rescaled when needed to operate on raw inputs (see Appendix C). The adversary MLPs are trained with activation = "relu", solver = "adam" and max_iter = 100. Since WB attacks exhibited greater variability, predictions are averaged across 5-fold cross-validation. To obtain robust statistics, this procedure is repeated five times for both WB and BB attacks, and the Hamming scores reported in Tables 3 and 4 correspond to the mean across these five repetitions.

Due to the monotonicity of LR, model parameters can be exactly recovered from scores (see Appendix D), making cBB and WB access equivalent, although WB is typically easier to exploit. Since tensorization approximates LR outputs with a TT representation, it is also possible to recover LR coefficients from TT evaluations. Thus, because TT coefficients are fully obfuscated, we assume a WB attacker would instead reconstruct the original LR coefficients and attack those directly; accordingly, we report also these attacks in Tables 3 and 4. This reconstruction is not possible for tensorized NNs, for which the attacker only has access to the TT parameters.

3.1.4 Evaluation metrics

Attack performance is reported as the Hamming score: the proportion of correctly predicted cohort-membership labels across all public patient cohorts and shadow-model instances. A score of 0.5 corresponds to chance-level prediction (random guessing), and a score of 1.0 indicates perfect identification of training datasets.

Clinical model performance is reported as median balanced accuracy and AUC across repeated shadow-model runs. Balanced accuracy uses a threshold based on Youden's J statistic rather than the standard 0.5 threshold, as the latter produced irregular results for DP models under strong noise. Tensorization occasionally produces degenerate models with accuracies near 50%; although rare, these can distort mean values, motivating the use of medians.

3.1.5 Implementation

All experiments[2] were run on an Intel Xeon CPU E5-2620 v4 with 256 GB RAM and an NVIDIA GeForce RTX 3090, using Scikit-Learn for LR models and NN-based attacks (Pedregosa et al., 2011), Diffprivlib for DP variants (Dwork, 2006b), PyTorch for NN models (Paszke et al., 2019), Opacus for DP-SGD training (Yousefpour et al., 2022), and TensorKrowch for TT models (Pareja Monturiol et al., 2024).

[2]The code is publicly available at: `https://anonymous.4open.science/r/tts4privacy`.

3.2 Experimental results

3.2.1 Public models leak patient cohort information

We begin by attacking LORIS as deployed. Since WB attacks are typically the strongest, we apply them to two sets of coefficients: (i) the LR coefficients of LORIS as released by Chang et al. (2024), and (ii) coefficients reconstructed from the web interface. Although the interface returns rounded probabilities rather than exact scores, we approximately invert the monotonic mapping defined for LORIS (Fig. 2c) to obtain usable coefficients (see Appendix D).

Table 2 shows that Cho1 is correctly identified as the training dataset in both cases, consistent with Chang et al. (2024). The recovered coefficients are noisier and assign high probability to Cho2 as well; however, Cho1 remains the dominant prediction. Since Cho1 and Cho2 correspond to train/test splits of the same dataset, both drawn from the same patient cohort, this spurious assignment likely reflects shared data characteristics. These results demonstrate that, even with noisy reconstructed coefficients, an adversary can infer training data membership with high confidence, highlighting the privacy risks of releasing or exposing LR parameters.

Table 2: WB attack scores for LORIS, using (i) the released model parameters from Chang et al. (2024), and (ii) coefficients reconstructed from queries to the web interface. Each score reflects the probability assigned by the meta-classifier to the presence of the corresponding cohort in the training set, where a score of 1.0 indicates identification with certainty. Note that Cho1 and Cho2 correspond to train/test partitions of the same original dataset, and thus high scores for both are expected.

	Cho1	Cho2	MSK1	MSK2	Shim	Kato
Released	**1.0000**	0.0007	0.0001	0.0000	0.0003	0.0000
Reconstructed	**0.9944**	0.8138	0.0440	0.0173	0.0005	0.0007

3.2.2 Privacy risk generalizes across model types and access levels

Having established the vulnerability of LORIS, we assess whether this risk generalizes to identifying other cohorts, including smaller ones, and to more complex models such as NNs. We also compare DP-based and tensorization-based defenses against such attacks. Table 3 reports Hamming scores across all model types and access levels, together with model performance metrics to illustrate the effect of privacy-preserving mechanisms on utility. Specifically, for each patient cohort, AUC scores are reported as the median over all models whose training set includes that cohort; these therefore reflect training-set performance rather than generalization, for which we refer the reader to Appendix A.1. Additionally, Appendix A.2 reports Hamming scores for the identification of each cohort separately, illustrating the generalizability of attacks across cohorts.

In Table 3 we distinguish between *vanilla* LR models, trained on a single run, and *averaged* LR models, trained via repeated cross-validation with coefficients averaged across folds—the procedure used to train LORIS. Notably, averaged LR models are more vulnerable than vanilla ones despite similar predictive performance. The variance reduction from cross-validation mitigates sample bias but also amplifies differences across models, making averaged models more identifiable than vanilla ones—a counterintuitive finding with direct implications for clinical practice, where cross-validated averaged models are often recommended precisely to improve generalization. However, ensemble methods may still be privacy-preserving when the individual models or the aggregation method are themselves private, as in PATE (Papernot et al., 2017).

Beyond the cross-validation finding, the results yield three main observations. First, unprotected LR and NN models yield the highest attack scores, underscoring their vulnerability when released without protection. Second, larger ε and b values correspond to higher data leakage, paired with higher performance, as expected. Third, attack scores increase with deeper levels of access, with cBB and WB achieving high values in many cases. Somewhat unexpectedly, WB attacks on NNs achieve lower scores than BB attacks, likely reflecting the difficulty of extracting structured information from more complex parameter spaces.

Table 3: Hamming scores and median AUC scores for all model types and access levels. Hamming scores reflect the proportion of correct cohort-membership predictions across all cohorts and attacked models; a score of 0.5 corresponds to random guessing and 1.0 to perfect identification. AUC scores are reported as the median over all models whose training set contains that cohort, and therefore reflect training-set performance rather than generalization. Vanilla LR models are trained on a single 80/20 split, while averaged LR models are trained via repeated cross-validation with coefficients averaged across folds, following the procedure used to train LORIS (Chang et al., 2024). ⋆WB attacks on TT-LR use LR coefficients reconstructed from TT evaluations rather than TT parameters directly (see Appendix D).

		Attack Hamming score			Model AUC					
		bBB	cBB	WB	Cho1	Cho2	MSK1	MSK2	Shim	Kato
LR	(vanilla)	0.8218	0.9238	0.9353	0.74	0.76	0.71	0.62	0.61	0.75
	(averaged)	**0.9204**	**0.9996**	**1.0000**	0.74	0.77	0.71	0.63	0.61	0.75
LR-DP	($\varepsilon = 0.1$)	0.5417	0.5456	0.5239	0.51	0.51	0.50	0.50	0.50	0.50
	($\varepsilon = 1$)	0.5851	0.5902	0.5562	0.58	0.58	0.55	0.53	0.54	0.53
	($\varepsilon = 10$)	0.7313	0.8061	0.6692	0.73	0.75	0.69	0.62	0.60	0.67
	($\varepsilon = 100$)	0.7750	0.9099	0.8791	0.74	0.76	0.70	0.63	0.61	0.74
TT-LR	($b = 2$)	0.6711	0.8328	0.5145 (0.7497⋆)	0.68	0.70	0.67	0.59	0.61	0.62
	($b = 6$)	0.7565	0.8700	0.5148 (0.8033⋆)	0.71	0.73	0.69	0.62	0.61	0.66
	($b = 10$)	0.7715	0.8774	0.5146 (0.8171⋆)	0.71	0.74	0.69	0.62	0.61	0.67
NN		**0.7463**	**0.9073**	**0.6455**	0.77	0.79	0.71	0.63	0.64	0.79
NN-DP	($\varepsilon \approx 0.2$)	0.5336	0.6315	0.5247	0.59	0.59	0.49	0.60	0.53	0.54
	($\varepsilon \approx 1$)	0.5177	0.6507	0.5097	0.63	0.62	0.53	0.62	0.55	0.55
	($\varepsilon \approx 10$)	0.6044	0.6825	0.5081	0.66	0.64	0.58	0.63	0.57	0.57
	($\varepsilon = \infty$)	0.6261	0.7716	0.6474	0.74	0.74	0.67	0.64	0.62	0.72
TT-NN	($b = 2$)	0.5845	0.6495	0.5183	0.68	0.68	0.62	0.61	0.58	0.60
	($b = 6$)	0.6351	0.8492	0.5144	0.72	0.75	0.68	0.62	0.63	0.71
	($b = 10$)	0.6659	0.8619	0.5157	0.73	0.75	0.68	0.62	0.63	0.71

3.2.3 Tensorization compared with Differential Privacy

Although DP provides protection at all access levels, it typically comes at a high cost in performance. The lowest privacy budgets ($\varepsilon \approx 0.1, 1$), arguably the only ones offering rigorous guarantees, have the largest impact on model utility, making them impractical. For the largest values ($\varepsilon \approx 100, \infty$), AUC scores approach non-DP levels, while attacks still perform worse than in the unprotected case, indicating that even negligible noise helps in practice. For NN-DP models, predictive performance is hindered even at $\varepsilon = \infty$, highlighting the substantial impact of DP-SGD on NN training. Overall, the intermediate case $\varepsilon \approx 10$ offers the best balance between privacy and utility.

For TT models, even with $b = 2$, retaining the least information from the original model, we achieve privacy protection comparable to DP with $\varepsilon \in (1, 10)$, typically with higher AUC scores. Although larger b values slightly improve performance, the gains are modest compared to the sharp increase in attack success. This highlights a key advantage of tensorization as a knowledge-distillation mechanism: it reconstructs an effective model from highly restricted information while preserving utility, in contrast to DP methods, which inject noise directly into the learning process. Regarding WB access, TT parameters are fully obfuscated, yielding attack accuracies close to 50%. For TT-LR models specifically, since they approximate the original LR, we additionally apply the coefficient reconstruction technique used in Section 3.2.1, yielding the starred WB scores in Table 3. These do not outperform BB attacks on the original LR because the TT is constructed solely from discretized outputs and therefore cannot leak more information than what is visible through

b-discretized LR evaluations. For TT-NN models, where NN parameters cannot be recovered, obfuscation is complete and WB attack accuracies remain around 50%.

3.2.4 Even small cohorts of 35 patients can be identified

As an illustrative case, we consider the extreme task of distinguishing models trained only on Cho1 (964 samples) from those trained on Cho1 plus the small Kato cohort (35 samples). This simulates a high-risk scenario where an adversary detects the inclusion of a very small subgroup, approaching individual membership inference. Table 4 shows attack accuracies for the Kato label. As expected, bBB attacks are nearly random. In contrast, averaged LRs reach approximately 92% detection under cBB and achieve perfect classification under WB. Notably, even vanilla LRs under WB access attain approximately 71% accuracy. These results show that even a 35-sample cohort can be reliably identified within a larger dataset. Model averaging and WB access amplify leakage, while TT models remain robust and do not reveal the presence of Kato under any access type.

For context, Appendix A.3 reports model performance on Kato. TT models show some degradation, especially in AUC, but this does not fully explain the attack results: the performance gap between training on Cho1 or Cho1+Kato is similar for TTs and LRs, and NNs achieve even higher accuracies while leaking no information.

Table 4: Hamming scores of adversarial classifiers distinguishing models trained on Cho1 from those trained on Cho1+Kato, evaluated on the Kato label. Kato is the smallest cohort in the study, with only 35 patients. Each score reflects the probability assigned by the meta-classifier to the presence of Kato in the training set. *WB attacks on TT-LR use LR coefficients reconstructed from TT evaluations (see Appendix D).

		bBB	cBB	WB
LR	(vanilla)	0.5981	0.6217	0.7141
	(averaged)	**0.5065**	**0.9182**	**1.0000**
TT-LR	($b = 2$)	0.5375	0.5811	0.4966 (0.5521*)
NN		0.5253	0.4641	0.5246
TT-NN	($b = 2$)	0.4779	0.4962	0.5152

4 Interpretability with tensor trains

Beyond privacy guarantees, interpretability is essential in clinical prediction. The utility of LORIS lies not only in its accuracy, but also in its ability to provide insights into relevant features and produce scores monotonically correlated with population level response probability—properties that help verify whether the model leverages known biological processes, contributing to its trustworthiness. Here we show that TT models retain similar interpretability, leveraging efficient computation of marginal and conditional distributions.

4.1 Feature sensitivity

The interpretability of LR models comes from their coefficients, which quantify how each feature affects the log-odds of the predicted outcome. For TT models, an analogous notion is obtained from conditional and marginal distributions. Unlike LR coefficients, which isolate each feature's independent effect, TT sensitivities may vary with other features due to non-linearity. To emulate LR coefficients, we marginalize over all but one feature and the response, and measure how the predicted score changes under a unit increment of the selected feature. This procedure yields independent sensitivity scores that can be computed efficiently within the TT structure (see Appendix B).

To evaluate this approach, we tensorized a vanilla LR trained on Cho1 and compared TT sensitivity scores with LR coefficients. We also included a tensorized NN model to assess how NN-based sensitivities compare to LR insights. As shown in Fig. 2a, LR and TT-LR scores align almost perfectly after normalization by

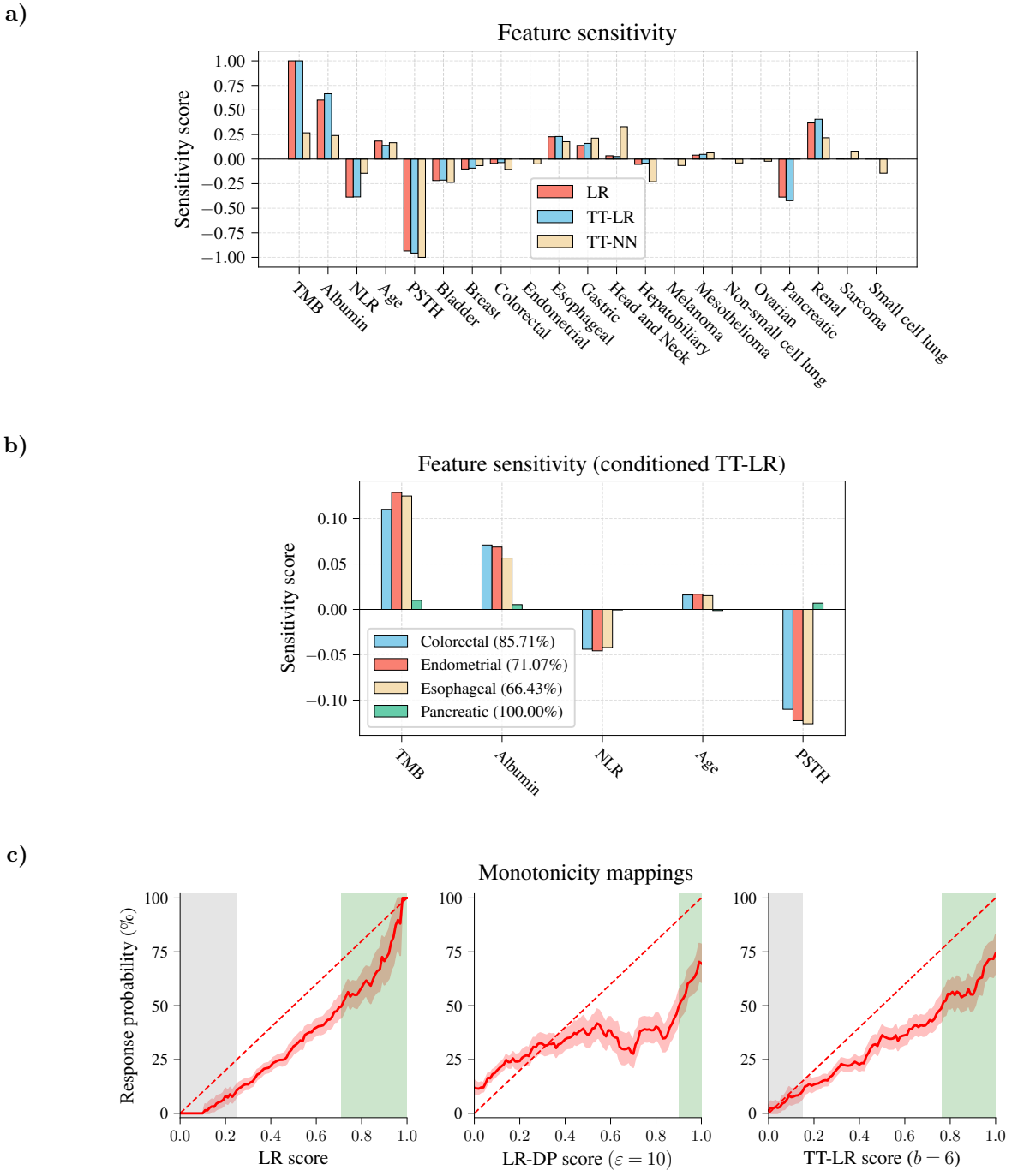

Figure 2: **Interpretability of TT models.** **(a)** Feature sensitivity scores from LR and TT models (TT-LR and TT-NN, $b = 6$). LR scores correspond to model coefficients, while TT scores are obtained via marginalization. All values are normalized by the maximum absolute score. **(b)** Feature sensitivities from cancer-type-conditioned TT-LR models ($b = 6$), obtained without retraining. The legend indicates cancer type and balanced accuracy of each conditioned model on the corresponding data. **(c)** Monotonicity plots of LR, LR-DP ($\varepsilon = 10$) and TT-LR ($b = 6$) scores with respect to true response probability, estimated via bootstrapping (95% confidence intervals). Shaded regions indicate participants with unlikely (gray, $< 10\%$) or likely (green, $> 50\%$) response probability. Limits of these regions from left to right: (0.25, 0.71), (0.00, 0.90), and (0.15, 0.76).

the maximum absolute value to remove scale differences. TT-NN yields similar relative patterns, albeit with larger scaling differences. These results confirm that TTs preserve LR interpretability while extending the framework to more complex black-box models.

4.2 Feature sensitivity by cancer type

TTs also allow conditional analysis, enabling sensitivity computation for specific subgroups. Conditioning on cancer type produces smaller TT models that capture type-specific behaviors (see Appendix B). Unlike the normalized comparison above, scores are directly comparable across cancer types since they are computed with the same method.

Figure 2b shows feature sensitivities for colorectal, endometrial, esophageal, and pancreatic cancers. While LR would provide identical scores across types, TTs reveal subtle variations. In particular, pancreatic cancer yields uniformly small sensitivities. This occurs because all pancreatic cancer patients in Cho1 are non-responders: the model achieves 100% accuracy simply by assigning very low response probabilities to all samples, independently of their features. Consequently, no feature appears relevant for prediction within this subgroup. These results highlight how TT interpretability can reveal subgroup-specific effects not captured by linear models.

4.3 Monotonicity of TT scores

A key property of LORIS scores, highlighted by Chang et al. (2024), is their monotonic relation with response probability: higher scores directly correlate with a greater chance of response, allowing clinicians to use the score as an intuitive ranking of treatment suitability. Although LR models are trained on binary labels, their scores align with mean response probabilities across patients sharing a given score. We verify this via bootstrapping to compute 95% confidence intervals for a vanilla LR model trained on Cho1. For comparison, we construct the same mapping for a DP-protected LR model, with $\varepsilon = 10$, and for a tensorized LR model with $b = 6$.

Figure 2c shows the results. For the tensorized model, we observe a clear monotonic trend with a lower slope than the unprotected model, reflecting that discretization pushes the model toward more extreme values. Increasing the number of bins improves the approximation, yielding a mapping closer to that of the LR model, though with potentially weaker privacy guarantees. In contrast, the DP model exhibits a noisier, non-monotonic mapping as a result of the noise injection mechanism used during training. This highlights that, although DP can provide privacy protection in practice, its effects may be less predictable and less uniform across settings.

5 Discussion

Our results show that publicly available clinical ML models pose a concrete and immediate privacy risk. LORIS—a model developed with rigorous scientific standards and made available to support reproducibility—can be attacked to identify training cohorts with high confidence from its published parameters or web interface alone. This reflects a gap in current practice: privacy evaluation is not yet a standard component of clinical model deployment. Achieving high predictive performance is critical for clinical models, and must be balanced with interpretability; privatization mechanisms that compromise either are unlikely to be adopted in practice. However, as we show in this work, certain procedures can be avoided or implemented to meaningfully improve privacy with little impact on performance.

A particularly actionable finding concerns cross-validation. Although useful for model selection, averaging LR models for deployment should be avoided, as it amplifies privacy risks without offering meaningful accuracy gains. This result is counterintuitive: practitioners who use repeated cross-validation to obtain stable, well-calibrated models are inadvertently making those models easier to attack. The mechanism is clear in retrospect—variance reduction sharpens the fingerprint that training data leave on model parameters—but it is not widely recognized in the clinical ML community. Ensemble methods remain viable, but privacy-

preserving aggregation strategies such as PATE (Papernot et al., 2017), which adds noise to the output aggregation step to ensure DP, should be considered.

Tensorization addresses these vulnerabilities without requiring changes to the training pipeline. Its key practical advantage over DP is that it is post-hoc: it can be applied to any existing model—including LORIS—using only black-box access, without retraining or access to the original patient data. Black-box privacy arises from tensorizing discretized rather than raw continuous scores, with the discretization level b providing an intuitive privacy–utility knob, while white-box privacy follows from the freedom in TT parameterizations (Pozas-Kerstjens et al., 2024).

Comparing TT-based protection with DP, we find similar privacy and utility for certain combinations of b and ε, although a direct correspondence is not possible without DP-style output perturbation. Our results suggest that the variability introduced by discretization plays a role analogous to noise injection. Notably, even binary discretization ($b = 2$) yields privacy protection comparable to small-ε DP but with substantially less accuracy loss, resulting in a more favorable privacy–utility trade-off. Furthermore, while DP provides formal theoretical guarantees, it requires empirical tuning to identify a practical value of ε, typically involving retraining on additional patient data. Moreover, our results confirm prior findings (Ziller et al., 2024): only large ε values are practical, while meaningful ones severely degrade accuracy.

Beyond privacy, we showed that TTs recover LR interpretability while enabling richer analyses, including subgroup-specific effects. More importantly, TT interpretability extends naturally to tensorized NNs, suggesting that our approach can help "open the box" of otherwise opaque models. Thus, even when privacy is not the primary goal, tensorization provides a powerful framework for extracting insights from pre-trained models, reinforcing its value as a broadly applicable tool for both privacy and interpretability.

A natural extension of this work is to combine tensorization with other output obfuscation methods (Jia et al., 2019; Yang et al., 2020; Ye et al., 2022), and in particular with DP-style output perturbation, which could provide formal ε-DP guarantees while maintaining the accuracy advantages demonstrated here. We also note that the tensorization framework is not limited to immunotherapy response prediction or to any specific model architecture: any pre-trained model can be tensorized post-hoc, making this approach immediately applicable across the full range of clinical ML models currently in deployment. Looking forward, we advocate for tensorization as a standard practice before deploying clinical ML models, analogous to data anonymization before publication.

Broader impact statement

This work identifies privacy risks in publicly available clinical prediction models and evaluates a post-hoc defense intended to reduce training-data leakage while preserving predictive accuracy and interpretability. A potential negative impact is that the described attacks could facilitate attempts to infer the participation of small cohorts in deployed models. We mitigate this risk by restricting the analysis to publicly available cohorts and model information, and by developing protection mechanisms that reduce the information exposed through model outputs and parameters.

References

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*, CCS '16, pp. 308–318, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450341394. doi: 10.1145/2976749.2978318. URL https://arxiv.org/abs/1607.00133.

Borja Aizpurua, Samuel Palmer, and Roman Orus. Tensor networks for explainable machine learning in cybersecurity, 2024. URL https://arxiv.org/abs/2401.00867.

Giuseppe Ateniese, Luigi V. Mancini, Angelo Spognardi, Antonio Villani, Domenico Vitali, and Giovanni Felici. Hacking smart machines with smarter ones: How to extract meaningful data from machine learning

classifiers. *Int. J. Secur. Netw.*, 10(3):137–150, 2015. doi: 10.1504/IJSN.2015.071829. URL https://arxiv.org/abs/1306.4447.

Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. Differential privacy has disparate impact on model accuracy. In *Adv. Neural Inf. Process. Syst.*, pp. 15374–15383, Vancouver, BC, Canada, 2019. doi: 10.5555/3454287.3455674. URL https://arXiv.org/abs/1905.12101.

Borja Balle, Giovanni Cherubin, and Jamie Hayes. Reconstructing training data with informed adversaries. *CoRR*, abs/2201.04845, 2022. URL https://arXiv.org/abs/2201.04845.

Tian-Gen Chang, Yingying Cao, Hannah J. Sfreddo, Saugato Rahman Dhruba, Se-Hoon Lee, Cristina Valero, Seong-Keun Yoo, Diego Chowell, Luc G. T. Morris, and Eytan Ruppin. Loris robustly predicts patient outcomes with immune checkpoint blockade therapy using common clinical, pathologic and genomic features. *Nat. Cancer*, 5(8):1158–1175, 2024. doi: 10.1038/s43018-024-00772-7.

Kamalika Chaudhuri, Claire Monteleoni, and Anand D. Sarwate. Differentially private empirical risk minimization. *J. Mach. Learn. Res.*, 12:1069–1109, 2011. doi: 10.5555/1953048.2021036. URL https://arxiv.org/abs/0912.0071.

Die Chowell, Sung K. Yoo, Carmen Valero, Alice Pastore, Chetan Krishna, Michael Lee, Daniel Hoen, Hsin-Ta Shi, David W. Kelly, Nikhil Patel, Vladimir Makarov, Xiaolei Ma, Lauren Vuong, Edgar Y. Sabio, Kyle Weiss, Frances Kuo, Tobias L. Lenz, Robert M. Samstein, Nadeem Riaz, Prasad S. Adusumilli, Vikas P. Balachandran, George Plitas, A. Ari Hakimi, Omar Abdel-Wahab, Arjun N. Shoushtari, Michael A. Postow, Robert J. Motzer, Marc Ladanyi, Ahmet Zehir, Michael F. Berger, Mithat Gönen, Levi G. T. Morris, Nicole Weinhold, and Timothy A. Chan. Improved prediction of immune checkpoint blockade efficacy across multiple cancer types. *Nat. Biotechnol.*, 40(4):499–506, 2022. doi: 10.1038/s41587-021-01070-8.

J. Ignacio Cirac, David Pérez-García, Norbert Schuch, and Frank Verstraete. Matrix product states and projected entangled pair states: Concepts, symmetries, theorems. *Rev. Mod. Phys.*, 93:045003, 2021. doi: 10.1103/RevModPhys.93.045003. URL https://arxiv.org/abs/2011.12127.

Cynthia Dwork. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography*, volume 3876 of *Lecture Notes in Computer Science*, pp. 265–284. Springer Berlin Heidelberg, 2006a. doi: 10.1007/11681878_14. URL https://iacr.org/archive/tcc2006/38760266/38760266.pdf.

Cynthia Dwork. Differential privacy. In *Automata, Languages and Programming*, volume 4052 of *Lecture Notes in Computer Science*, pp. 1–12. Springer Berlin Heidelberg, 2006b. doi: 10.1007/11787006_1. URL https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/dwork.pdf.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3–4):211–407, 2014. doi: 10.1561/0400000042. URL https://www.cis.upenn.edu/~aaroth/Papers/privacybook.pdf.

Yuriel Núñez Fernández, Marc K. Ritter, Matthieu Jeannin, Jheng-Wei Li, Thomas Kloss, Thibaud Louvet, Satoshi Terasaki, Olivier Parcollet, Jan von Delft, Hiroshi Shinaoka, and Xavier Waintal. Learning tensor networks with tensor cross interpolation: New algorithms and libraries. *SciPost Phys.*, 18:104, 2025. doi: 10.21468/SciPostPhys.18.3.104. URL https://scipost.org/10.21468/SciPostPhys.18.3.104.

Matthew Fredrikson, Eric Lantz, Somesh Jha, Simon Lin, David Page, and Thomas Ristenpart. Privacy in pharmacogenetics: An end-to-end case study of personalized warfarin dosing. In *USENIX Security '14*, pp. 17–32, 2014. URL https://www.usenix.org/system/files/conference/usenixsecurity14/sec14-paper-fredrikson-privacy.pdf.

Simson L. Garfinkel, John M. Abowd, and Sarah Powazek. Issues encountered deploying differential privacy. In *Proc. WPES*, pp. 133–137, Toronto, Ontario, Canada, 2018. ACM. doi: 10.1145/3267323.3268949. URL https://arXiv.org/abs/1809.02201.

Niv Haim, Gal Vardi, Gilad Yehudai, Ohad Shamir, and Michal Irani. Reconstructing training data from trained neural networks. In *Adv. Neural Inf. Process. Syst.*, volume 35, pp. 22911–22924, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/file/906927370cbeb537781100623cca6fa6-Paper-Conference.pdf.

Victor Petren Bach Hansen, Atula Tejaswi Neerkaje, Ramit Sawhney, Lucie Flek, and Anders Søgaard. The impact of differential privacy on group disparity mitigation. In *Findings ACL-NAACL*, pp. 3952–3965, Mexico City, Mexico, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-naacl.249. URL https://aclanthology.org/2024.findings-naacl.249/.

Yangsibo Huang, Yushan Su, Sachin Ravi, Zhao Song, Sanjeev Arora, and Kai Li. Privacy-preserving learning via deep net pruning, 2020. URL https://arxiv.org/abs/2003.01876.

YoonHaeng Hur, Jeremy G. Hoskins, Michael Lindsey, E.M. Stoudenmire, and Yuehaw Khoo. Generative modeling via tensor train sketching. *App. Comput. Harmon. Anal.*, 67:101575, 2023. ISSN 1063-5203. doi: 10.1016/j.acha.2023.101575. URL http://arxiv.org/abs/2202.11788.

Jinyuan Jia, Ahmed Salem, Michael Backes, Yang Zhang, and Neil Zhenqiang Gong. Memguard: Defending against black-box membership inference attacks via adversarial examples. In *Proc. ACM SIGSAC Conf. on Computer and Communications Security (CCS)*, CCS '19, pp. 259–274, New York, NY, USA, 2019. Association for Computing Machinery. doi: 10.1145/3319535.3363201. URL https://arxiv.org/abs/1909.10594.

Michael Kapralov and Kunal Talwar. On differentially private low rank approximation. In *Proc. ACM–SIAM Symp. Discrete Algorithms*, pp. 1395–1414. SIAM, 2013. doi: 10.1137/1.9781611973105.101. URL https://epubs.siam.org/doi/abs/10.1137/1.9781611973105.101.

Shumei Kato, Ki Hwan Kim, Hyo Jeong Lim, Amelie Boichard, Mina Nikanjam, Elizabeth Weihe, Dennis J. Kuo, Ramez N. Eskander, Aaron Goodman, Natalie Galanina, Paul T. Fanta, Richard B. Schwab, Rebecca Shatsky, Steven C. Plaxe, Andrew Sharabi, Edward Stites, Jacob J. Adashek, Ryosuke Okamura, Suzanna Lee, Scott M. Lippman, Jason K. Sicklick, and Razelle Kurzrock. Real-world data from a molecular tumor board demonstrates improved outcomes with a precision n-of-one strategy. *Nat. Commun.*, 11(1):4965, 2020. doi: 10.1038/s41467-020-18613-3.

Xiao-Yang Liu, Rongyi Zhu, Daochen Zha, Jiechao Gao, Shan Zhong, Matt White, and Meikang Qiu. Differentially private low-rank adaptation of large language model using federated learning. *ACM Trans. Manage. Inf. Syst.*, 16(2):1–24, 2025. doi: 10.1145/3682068. URL https://arxiv.org/abs/2312.17493.

Alex Mossi, Bojan Žunkovič, and Kyriakos Flouris. A matrix product state model for simultaneous classification and generation. *Quantum Mach. Intell.*, 7(1):48, 2025. ISSN 2524-4914. doi: 10.1007/s42484-025-00272-6. URL https://arxiv.org/abs/2406.17441.

Alexander Novikov, Dmitrii Podoprikhin, Anton Osokin, and Dmitry P. Vetrov. Tensorizing neural networks. In *Adv. Neural Inf. Process. Syst.*, volume 28. Curran Associates, Inc., 2015. URL https://arxiv.org/abs/1509.06569.

Alexander Novikov, Mikhail Trofimov, and Ivan V. Oseledets. Exponential machines. *Bull. Pol. Acad. Sci. Tech. Sci.*, 66(No 6 (Special Section on Deep Learning: Theory and Practice)):789–797, 2018. doi: 10.24425/bpas.2018.125926. URL https://arxiv.org/abs/1605.03795.

Román Orús. A practical introduction to tensor networks: Matrix product states and projected entangled pair states. *Ann. Phys.*, 349:117–158, 2014. doi: 10.1016/j.aop.2014.06.013. URL https://arxiv.org/abs/1306.2164.

Ivan Oseledets. Tensor-train decomposition. *SIAM J. Sci. Comput.*, 33(5):2295–2317, 2011. doi: 10.1137/090752286.

Yakir Oz, Gilad Yehudai, Gal Vardi, Itai Antebi, Michal Irani, and Niv Haim. Reconstructing training data from real-world models trained with transfer learning. *CoRR*, abs/2407.15845, 2024. URL https://arXiv.org/abs/2407.15845.

Nicolas Papernot, Martín Abadi, Úlfar Erlingsson, Ian J. Goodfellow, and Kunal Talwar. Semi-supervised knowledge transfer for deep learning from private training data. In *Int. Conf. Learn. Represent.*, 2017. URL https://arxiv.org/abs/1610.05755.

José Ramón Pareja Monturiol, David Pérez-García, and Alejandro Pozas-Kerstjens. TensorKrowch: Smooth integration of tensor networks in machine learning. *Quantum*, 8:1364, 2024. doi: 10.22331/q-2024-06-11-1364. URL https://arxiv.org/abs/2306.08595. https://github.com/joserapa98/tensorkrowch.

José Ramón Pareja Monturiol, Alejandro Pozas-Kerstjens, and David Pérez-García. Tensorization of neural networks for improved privacy and interpretability, 2025. URL https://arxiv.org/abs/2501.06300.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zach DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library, 2019. URL https://arxiv.org/abs/1912.01703.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Andreas Müller, Joel Nothman, Gilles Louppe, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine learning in Python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011. URL https://arxiv.org/abs/1201.0490.

Alejandro Pozas-Kerstjens, Senaida Hernández-Santana, José Ramón Pareja Monturiol, Marco Castrillón López, Giannicola Scarpa, Carlos E. González-Guillén, and David Pérez-García. Privacy-preserving machine learning with tensor networks. *Quantum*, 8:1425, 2024. doi: 10.22331/q-2024-07-25-1425. URL https://arxiv.org/abs/2202.12319.

David Pérez-García, Frank Verstraete, Michael M. Wolf, and J. Ignacio Cirac. Matrix product state representations. *Quantum Inf. Comput.*, 7(5):401—430, 2007. doi: 10.26421/QIC7.5-6-1. URL https://arxiv.org/abs/quant-ph/0608197.

Virat Shejwalkar and Amir Houmansadr. Membership privacy for machine learning models through knowledge transfer, 2020. URL https://arxiv.org/abs/1906.06589.

J. H. Shim, H. S. Kim, H. Cha, S. Kim, T. M. Kim, V. Anagnostou, Y. L. Choi, H. A. Jung, J. M. Sun, J. S. Ahn, M. J. Ahn, K. Park, W. Y. Park, and S. H. Lee. Hla-corrected tumor mutation burden and homologous recombination deficiency for the prediction of response to pd-(l)1 blockade in advanced non-small cell lung cancer patients. *Ann. Oncol.*, 31(7):902–911, 2020. doi: 10.1016/j.annonc.2020.04.004.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *2017 IEEE Symposium on Security and Privacy*, pp. 3–18, 2017. doi: 10.1109/SP.2017.41. URL https://arxiv.org/abs/1610.05820.

Edwin Stoudenmire and David J. Schwab. Supervised learning with tensor networks. In *Adv. Neural Inf. Process. Syst.*, volume 29, pp. 4799–4807. Curran Associates, Inc., 2016. URL https://arxiv.org/abs/1605.05775.

Latanya Sweeney. Only you, your doctor, and many others may know. *Technology Science*, 2015092903, 2015. URL https://techscience.org/a/2015092903/.

Jirawat Tangpanitanon, Chanatip Mangkang, Pradeep Bhadola, Yuichiro Minato, Dimitris G. Angelakis, and Thiparat Chotibut. Explainable natural language processing with matrix product states. *New J. Phys.*, 24(5):053032, 2022. doi: 10.1088/1367-2630/ac6232. URL https://arxiv.org/abs/2112.08628.

Andrei Tomut, Saeed S. Jahromi, Abhijoy Sarkar, Uygar Kurt, Sukhbinder Singh, Faysal Ishtiaq, Cesar Muñoz, Prabdeep Singh Bajaj, Ali Elborady, Gianni del Bimbo, Mehrazin Alizadeh, David Montero, Pablo Martin-Ramiro, Muhammad Ibrahim, Oussama Tahiri Alaoui, John Malcolm, Samuel Mugel, and Roman Orus. Compactifai: Extreme compression of large language models using quantum-inspired tensor networks, 2024. URL https://arxiv.org/abs/2401.14109.

Jinhui Wang, Chase Roberts, Guifré Vidal, and Stefan Leichenauer. Anomaly detection with tensor networks, 2020. URL https://arxiv.org/abs/2006.02516.

Ziqi Yang, Bin Shao, Bohan Xuan, Ee-Chien Chang, and Fan Zhang. Defending model inversion and membership inference attacks via prediction purification. *arXiv preprint*, 2020. URL https://arxiv.org/abs/2005.03915.

Ziqi Yang, Lijin Wang, Da Yang, Jie Wan, Ziming Zhao, Ee-Chien Chang, Fan Zhang, and Kui Ren. Purifier: Defending data inference attacks via transforming confidence scores. In *Proc. AAAI Conf. Artif. Intell.*, volume 37, pp. 10871–10879, Jun. 2023. doi: 10.1609/aaai.v37i9.26289. URL https://ojs.aaai.org/index.php/AAAI/article/view/26289.

Dayong Ye, Sheng Shen, Tianqing Zhu, Bo Liu, and Wanlei Zhou. One parameter defense—defending against data inference attacks via differential privacy. *IEEE Trans. Inf. Forensics Security*, 17:1466–1480, 2022. doi: 10.1109/TIFS.2022.3163591. URL https://arxiv.org/abs/2203.06580.

Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, Graham Cormode, and Ilya Mironov. Opacus: User-friendly differential privacy library in pytorch, 2022. URL https://arxiv.org/abs/2109.12298.

Alexander Ziller, Tamara T. Mueller, Simon Stieger, Leonhard F. Feiner, Johannes Brandt, Rickmer Braren, Daniel Rueckert, and Georgios Kaissis. Reconciling privacy and accuracy in ai for medical imaging. *Nat. Mach. Intell.*, 6(7):764–774, 2024. doi: 10.1038/s42256-024-00858-y.

A Additional privacy results

A.1 Performance of models trained on Cho1

In this section we provide additional results supporting the conclusions of the main text. Specifically, we report: (i) performance metrics of models trained on Cho1 (the largest cohort, 964 samples) and evaluated on all cohorts; (ii) per-cohort attack accuracies for LR, NN, and TT models; and (iii) performance of models trained on Cho1 versus Cho1+Kato.

Figure 3 shows the overall performance of models trained exclusively on Cho1, reporting balanced accuracy distributions across all public cohorts. Tensorization occasionally produces degenerate models with accuracies near 50%, which, although rare, can distort mean values. For this reason, we report median accuracies and AUC scores throughout, as they better capture typical behavior. Since the remaining distributions are approximately Gaussian and symmetric, median and mean coincide, making median values representative. The right panel of Figure 3 further shows how the performance of DP models improves with increasing ε, as the added noise decreases and the distribution converges to the narrow non-DP case.

Table 5 reports median balanced accuracies and AUC scores across all cohorts for models trained on Cho1. Note that balanced accuracies use a threshold based on Youden's J statistic rather than the standard 0.5 threshold, which produced irregular results for DP models under strong noise.

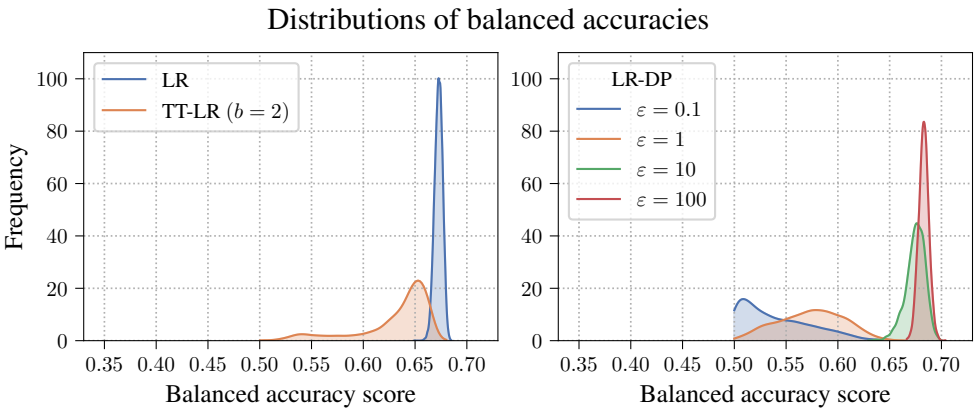

Figure 3: Balanced accuracy distributions of models trained on Cho1, evaluated on all cohorts. Left panel: LR vanilla and TT-LR with $b = 2$. Right panel: LR-DP models across privacy budgets ε, showing how accuracy improves as ε increases and injected noise decreases. Degenerate TT runs with accuracy near 50% are rare and do not affect median values.

Table 5: Median balanced accuracies (left block) and median AUC scores (right block) of models trained on Cho1, evaluated on each cohort. Balanced accuracies use a threshold based on Youden's J statistic. These results complement the attack scores reported in the main text by showing the utility cost associated with each privacy mechanism under the same setting as LORIS.

		Balanced accuracy						AUC					
		Cho1	Cho2	MSK1	MSK2	Shim	Kato	Cho1	Cho2	MSK1	MSK2	Shim	Kato
LR	(vanilla)	**0.68**	**0.69**	**0.68**	**0.63**	**0.62**	**0.78**	**0.74**	**0.75**	**0.70**	**0.63**	**0.60**	**0.75**
	(averaged)	0.68	0.69	0.69	0.63	0.62	0.78	0.74	0.75	0.70	0.63	0.60	0.71
LR-DP	($\varepsilon = 0.1$)	0.52	0.53	0.53	0.56	0.53	0.58	0.49	0.49	0.50	0.49	0.49	0.49
	($\varepsilon = 1$)	0.56	0.57	0.56	0.60	0.56	0.62	0.56	0.57	0.55	0.55	0.54	0.51
	($\varepsilon = 10$)	**0.67**	**0.68**	**0.66**	**0.63**	**0.61**	**0.70**	**0.72**	**0.73**	**0.68**	**0.62**	**0.59**	**0.64**
	($\varepsilon = 100$)	0.68	0.69	0.68	0.63	0.62	0.78	0.74	0.75	0.70	0.63	0.60	0.75
TT-LR	($b = 2$)	**0.66**	**0.66**	**0.65**	**0.63**	**0.61**	**0.70**	**0.69**	**0.69**	**0.67**	**0.62**	**0.60**	**0.62**
	($b = 6$)	0.67	0.67	0.67	0.63	0.62	0.72	0.72	0.72	0.69	0.63	0.60	0.65
	($b = 10$)	0.67	0.68	0.67	0.63	0.62	0.73	0.72	0.72	0.69	0.63	0.60	0.65
NN		**0.72**	**0.69**	**0.64**	**0.63**	**0.62**	**0.80**	**0.78**	**0.74**	**0.66**	**0.61**	**0.63**	**0.75**
NN-DP	($\varepsilon \approx 0.2$)	0.58	0.59	0.53	0.63	0.57	0.62	0.59	0.59	0.49	0.60	0.53	0.55
	($\varepsilon \approx 1$)	0.60	0.61	0.54	0.64	0.59	0.63	0.61	0.61	0.52	0.62	0.55	0.55
	($\varepsilon \approx 10$)	0.61	0.62	0.56	0.64	0.60	0.65	0.65	0.63	0.56	0.64	0.57	0.57
	($\varepsilon = \infty$)	**0.68**	**0.68**	**0.63**	**0.66**	**0.62**	**0.75**	**0.73**	**0.72**	**0.65**	**0.63**	**0.62**	**0.73**
TT-NN	($b = 2$)	**0.66**	**0.67**	**0.63**	**0.65**	**0.61**	**0.72**	**0.70**	**0.69**	**0.65**	**0.62**	**0.60**	**0.67**
	($b = 6$)	0.68	0.69	0.65	0.64	0.62	0.77	0.73	0.74	0.67	0.62	0.62	0.73
	($b = 10$)	0.68	0.69	0.65	0.64	0.62	0.75	0.73	0.73	0.67	0.62	0.62	0.71

A.2 Per-cohort attack accuracies

Table 6 reports per-cohort Hamming scores for LR, NN, and TT models, i.e., the proportion of correct membership predictions for each cohort individually. These results show that privacy vulnerabilities extend to identifying all cohorts with high confidence regardless of their size, although the highest scores are generally obtained for the largest cohorts, which are the easiest to identify.

Table 6: Hamming scores of adversarial classifiers on vanilla models, reported separately for each cohort. A score of 0.5 corresponds to random guessing and 1.0 to perfect identification. Results complement the aggregate scores reported in the main text by showing which cohorts are most easily identified.

		Cho1	Cho2	MSK1	MSK2	Shim	Kato
LR (vanilla)	bBB	0.9412	0.9358	0.9089	0.7237	0.7713	0.6259
	cBB	0.9945	0.9782	0.9616	0.9626	0.9130	0.6675
	WB	0.9982	0.9915	0.9678	0.9716	0.9152	0.7537
TT-LR ($b = 2$)	bBB	0.7530	0.6927	0.7100	0.6614	0.6357	0.5471
	cBB	0.9343	0.8891	0.8740	0.8950	0.7517	0.5943
	WB*	0.8671	0.8064	0.7820	0.8517	0.5975	0.5722
NN	bBB	0.9256	0.8826	0.7948	0.6104	0.6788	0.5329
	cBB	0.7613	0.7091	0.6147	0.5507	0.5616	0.5128
	WB	0.8335	0.7370	0.6545	0.5558	0.5111	0.5098
TT-NN ($b = 2$)	bBB	0.6429	0.6347	0.5832	0.5449	0.5450	0.5049
	cBB	0.5796	0.5819	0.5445	0.5231	0.5224	0.4973
	WB	0.5140	0.5133	0.5071	0.4972	0.5002	0.5049

A.3 Performance of models trained on Cho1 vs. Cho1+Kato

Table 7 reports median balanced accuracies and AUC scores of models trained on Cho1 alone or on Cho1+Kato, evaluated on Kato. These results provide context for the attack scores discussed in the main text: TT models show some performance degradation relative to LR, especially in AUC, but this alone does not explain the attack differences, as NNs achieve higher accuracies while leaking no information.

Table 7: Median balanced accuracies and AUC scores of models trained on Cho1 or Cho1+Kato, evaluated on Kato (35 patients). These results support the attack findings in the main text by showing that performance differences between the two training conditions are similar across model types, and cannot alone explain the large differences in attack success.

		Bal. accuracy		AUC	
		Cho1	Cho1+Kato	Cho1	Cho1+Kato
LR	(vanilla)	0.7833	0.8000	0.7533	0.7733
	(averaged)	0.7833	0.8167	0.7133	0.7800
TT-LR	($b = 2$)	0.7167	0.7500	0.6167	0.6667
NN		0.8000	0.8167	0.7467	0.8067
TT-NN	($b = 2$)	0.7167	0.7333	0.6733	0.6733

B Efficient computations with TTs

A major advantage of tensor networks is their ability to represent high-order tensors using only a polynomial number of parameters. The TT representation of a tensor T given by Eq. equation 2 requires only $\mathcal{O}(Ndr^2)$ coefficients when all cores G_n are $r \times r$ matrices, as opposed to the d^N coefficients needed for a general tensor $T \in \mathbb{R}^{d^N}$. While compactness does not automatically imply fast computation, TTs are efficient to evaluate: computing $T(i_1, \ldots, i_N)$ scales polynomially in N, unlike higher-dimensional TNs where evaluation may require exponential time.

Beyond evaluating samples, TTs enable efficient marginalization. Suppose T encodes a probability distribution via the Born rule, $p(i_1, \ldots, i_N) = |T(i_1, \ldots, i_N)|^2$. Computing the partition function,

$$Z = \sum_{i_1,\ldots,i_N} p(i_1, \ldots, i_N), \tag{4}$$

is generally exponential in N, but in TT form it reduces to polynomial time by contracting each core with itself:

$$H_n(\alpha_{n-1}, \beta_{n-1}, \alpha_n, \beta_n) = \sum_{i_n} G_n(\alpha_{n-1}, i_n, \alpha_n)\, G_n(\beta_{n-1}, i_n, \beta_n), \tag{5}$$

yielding $r^2 \times r^2$ matrices H_n. Multiplying all H_n sequentially produces Z efficiently.

A similar procedure yields marginals by contracting only the cores of marginalized features. For instance, for a 2-site TT

$$T(i, j) = G_1(i)G_2(j), \tag{6}$$

the marginal $p(i)$ is

$$p(i) = \sum_{\alpha,\beta} G_1(i, \alpha)G_1(i, \beta)\, H_2(\alpha, \beta), \tag{7}$$

showing that marginals correspond to duplicate TTs with some cores contracted.

TT representations also enable efficient computation of conditional models without retraining. To compute $p(i_1, \ldots, i_{n-1}, i_{n+1}, \ldots, i_N \mid i_n = \mathbf{i}_n)$, it suffices to absorb the fixed feature into its neighbor:

$$\widetilde{G}_{n-1}(i_{n-1}) = G_{n-1}(i_{n-1})\, G_n(\mathbf{i}_n), \tag{8}$$

which defines a reduced, conditioned TT

$$\widetilde{T}(i_1, \ldots, i_{n-1}, i_{n+1}, \ldots, i_N) = G_1(i_1) \cdots \widetilde{G}_{n-1}(i_{n-1})G_{n+1}(i_{n+1}) \cdots G_N(i_N). \tag{9}$$

For further details on TTs and related tensor networks, see Cirac et al. (2021).

C Data standardization and parameter rescaling

Before training on each dataset $D = \{(x_1^k, \ldots, x_n^k, y^k)\}_k$, input features $x_1, \ldots, x_n$ are standardized as

$$\tilde{x}_j^k = \frac{x_j^k - \mu_j}{\sigma_j}, \tag{10}$$

where μ_j and σ_j denote the mean and standard deviation of feature j, respectively.

LR models are trained on these standardized inputs, but their parameters must be corrected in order to operate directly on raw features. Let $\tilde{\theta} = (\tilde{\mathbf{w}}, \tilde{b})$ be the parameters obtained after training, defining

$$\Phi_{\tilde{\theta}}(\mathbf{x}) = \text{sigmoid}(\tilde{\mathbf{w}}^\intercal \mathbf{x} + \tilde{b}), \quad \text{where} \quad \text{sigmoid}(z) = \frac{1}{1+e^{-z}}. \tag{11}$$

The corrected parameters are $\theta = (\mathbf{w}, b)$ with

$$w_j = \frac{\tilde{w}_j}{\sigma_j}, \quad b = \tilde{b} - \sum_j \frac{\tilde{w}_j \mu_j}{\sigma_j}. \tag{12}$$

This transformation ensures that trained models can be applied directly to raw inputs without explicit feature standardization.

An analogous rescaling applies to TT models. Consider a tensorized model with parameters $\widetilde{\mathrm{W}}$,

$$\tilde{f}(x_1, \ldots, x_N) = \sum_{i_1, \ldots, i_N} \widetilde{\mathrm{W}}(i_1, \ldots, i_N)\, \phi_1^{i_1}(x_1) \cdots \phi_N^{i_N}(x_N), \tag{13}$$

where $\phi_n(x) = [1, x]$ are polynomial embeddings (input dimension $d = 2$), and

$$\widetilde{\mathrm{W}}(i_1, \ldots, i_N) = \widetilde{G}_1(i_1) \cdots \widetilde{G}_N(i_N). \tag{14}$$

To compensate for feature standardization, we define a new coefficient tensor W from corrected cores G_n such that

$$\begin{aligned} G_n(1) &= \widetilde{G}_n(1) - \frac{\mu_j}{\sigma_j}\, \widetilde{G}_n(2), \\ G_n(2) &= \frac{1}{\sigma_j}\, \widetilde{G}_n(2). \end{aligned} \tag{15}$$

The resulting TT parameters are thus expressed in terms of raw input features, analogous to the LR case.

D Recovering LR coefficients from cBB access

Since logistic regression is linear in the log-odds space,

$$\text{logit}(\mathbf{x}) = \log \frac{p(y=1 \mid \mathbf{x})}{p(y=0 \mid \mathbf{x})} = \mathbf{w}^\intercal \mathbf{x} + b, \tag{16}$$

its parameters can be exactly recovered from model evaluations on carefully chosen inputs. If queries to the zero vector and one-hot vectors $\mathbf{e}_j$ are allowed, the intercept b is simply the logit at the zero vector, and each coefficient w_j is given by the difference between the logit at $\mathbf{e}_j$ and the intercept.

More generally, when queries are restricted to inputs with all features strictly positive (as is the case when attacking LORIS through its web interface), w_j can be recovered from two inputs $\mathbf{x}, \mathbf{x}'$ that differ only in feature j:

$$w_j = \frac{\text{logit}(\mathbf{x}) - \text{logit}(\mathbf{x}')}{x_j - x'_j}. \tag{17}$$

Once the weights are obtained, the intercept can be recovered from

$$b = \text{logit}(\mathbf{x}) - \mathbf{w}^\intercal \mathbf{x} \tag{18}$$

for any input $\mathbf{x}$.