Some Targets Are Harder to Identify than Others: Quantifying the Target-dependent Membership Leakage

Achraf Azize[1]
FairPlay Joint Team
CREST, ENSAE Paris

Debabrota Basu
Univ. Lille, Inria
CNRS, Centrale Lille
UMR 9189-CRIStAL

Abstract

In a Membership Inference (MI) game, an attacker tries to infer whether a target point was included or not in the input of an algorithm. Existing works show that some target points are easier to identify, while others are harder. This paper explains the target-dependent hardness of membership attacks by studying the powers of the optimal attacks in a *fixed-target* MI game. We characterise the optimal advantage and trade-off functions of attacks against the empirical mean in terms of the Mahalanobis distance between the target point and the data-generating distribution. We further derive the impacts of two privacy defences, i.e. adding Gaussian noise and sub-sampling, and that of target misspecification on optimal attacks. As by-products of our novel analysis of the Likelihood Ratio (LR) test, we provide a new covariance attack which generalises and improves the scalar product attack. Also, we propose a new optimal canary-choosing strategy for auditing privacy in the white-box federated learning setting. Our experiments validate that the Mahalanobis score explains the hardness of *fixed-target* MI games.

1 INTRODUCTION

A growing body of works on privacy attacks has shown that leakage of sensitive information through ML models is a common issue (Shokri et al., 2017; Yeom et al., 2018). For example, an attacker can violate the privacy of users involved in an input dataset by inferring whether a particular target data point of a user was used in the input dataset. These attacks are called Membership Inference (MI) attacks (Homer et al., 2008; Shokri et al., 2017), and are identified as a privacy threat for and confidentiality violation of the users' data by ICO (UK) and NIST (USA) (Murakonda and Shokri, 2020; Song and Marn, 2020).

The study of statistical efficiency and design of MI attacks has begun with the summary statistics on genomic data (Homer et al., 2008; Sankararaman et al., 2009; Dwork et al., 2015) under the name of *tracing attacks* (Dwork et al., 2017). Homer et al. (2008) and Sankararaman et al. (2009) studied the first attacks to detect an individual in the exact empirical mean statistic, computed on a dataset generated by Bernoulli distributions. They both propose and analyse the *Likelihood Ratio (LR) tests* and assume access to the exact statistics and a pool of *reference points*. Dwork et al. (2015) assumes access to only a noisy statistic and one reference sample, and develops a *scalar product attack* to understand the correlation of a target point with the marginals of noisy statistics. *However, these tracing attacks (Homer et al., 2008; Sankararaman et al., 2009; Dwork et al., 2015) are studied in a threat model where the target point attacked is sampled randomly, either from the input dataset, or from a data generating distribution.* This means that the metrics of the attack under analysis, i.e. the *advantage* of an attacker or *trade-off functions between Type-I/Type-II errors*, are 'averaged' over the target point's sampling. This obfuscates the target-dependent hardness of tracing attacks, which is important to understand due to the worst-case nature of privacy (Steinke and Ullman, 2020). Also, prior works (Carlini et al., 2022b; Ye et al., 2022) empirically demonstrated that some target points could be easier to identify than others. Hence, we ask the question

Why are some points statistically harder to identify

[1]This work is done during PhD at the Inria Centre at University of Lille.

Proceedings of the 28[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2025, Mai Khao, Thailand. PMLR: Volume 258.

than others, and how can we quantify this hardness?

Concurrent with the study of attacks, researchers have developed defence mechanisms to preserve the privacy of the users contributing to the input of an algorithm, and Differential Privacy (DP) has emerged as the gold standard (Dwork and Roth, 2014). Though DP promises to bound the worst-case privacy leakage on any attack, it is not always evident how these guarantees bound the accuracy of specific privacy attacks, such as MI attacks (Zhang et al., 2020; Humphries et al., 2023). A DP algorithm comes with a mathematical proof yielding an upper bound on the privacy parameters that control the maximum leakage. On the other hand, a *privacy audit* tries to empirically estimate the privacy parameters, by providing lower bounds on the parameters. Typically, a privacy audit runs an MI attack, and translates the Type-I/Type-II errors of the attack into a lower bound on the privacy budget. These algorithms apply different heuristics to find the most leaking target points (aka *canaries*) that minimise Type-I/Type-II errors to estimate the privacy parameters tightly (Maddock et al., 2022; Nasr et al., 2023). Thus, understanding the target-dependent hardness of MI attacks can lead to optimal canary-choosing strategies. We ask:

Can we quantify the target-dependent effect of privacy-preserving mechanisms?
How to design an optimal canary-choosing strategy?

Our contributions address these queries.
1. *Defining the target-dependent leakage.* To understand the target-dependent hardness of MI games, we instantiate a *fixed-target MI game* (Algorithm 2, Ye et al. (2022)). We define the leakage of a target point as the optimal advantage of its corresponding fixed-target MI game. We characterise the target-dependent leakage using a Total Variation distance (Equation (2)).

2. *Explaining the target-dependent leakage for the mean.* We investigate the fixed-target MI game for the empirical mean. We quantify the exact optimal advantage (Equation (3)) and trade-off function (Equation (4)) of the LR attack in this setting. This shows that the target-dependent hardness of MI games depends on the Mahalanobis distance between the target point $z^\star$ and the true data-generating distribution (Table 1).

3. *Tight quantification of the effects of noise addition, sub-sampling, and misspecified targets on leakage.* We further study the impact of privacy-preserving mechanisms, such as adding Gaussian noise and sub-sampling, on the target-dependent leakage. As shown in Table 1, both of them reduce the leakage scores, and thus, the powers of the optimal attacks. We also quantify how *target misspecification* affects the leakage.

Table 1: Target-dependent leakage score

SETTING	LEAKAGE SCORE
Empirical mean	$\frac{1}{n}\|z^\star - \mu\|^2_{C_\sigma^{-1}}$
Gaussian Noise ($\gamma > 0$)	$\frac{1}{n}\|z^\star - \mu\|^2_{(C_\sigma + C_\gamma)^{-1}}$
Sub-sampling ($\rho < 1$)	$\frac{\rho}{n}\|z^\star - \mu\|^2_{C_\sigma^{-1}}$
Similar point	$\frac{1}{n}\left(z^\star_{\text{targ}} - \mu\right)^T C_\sigma^{-1}\left(z^\star_{\text{true}} - \mu\right)$

4. *A new covariance attack and optimal canary-choosing strategy.* We analyse the LR score for the empirical mean asymptotically. Our novel proof technique combines an Edgeworth expansion with a Lindeberg-Feller central limit theorem to show that the *LR score is asymptotically a scalar product attack, corrected by the inverse of the covariance matrix* (Equation (5)). This enables us with a novel attack score that improves the scalar product by correcting it for the geometry of the data. We use this "covariance score" to propose a novel white-box attack (Algo. 4) that experimentally outperforms the scalar product attack. We also use the insights from the target-dependent leakage to propose a new white-box *optimal* canary-choosing strategy (Algo. 3), based on an estimated Mahalanobis distance.

2 MEMBERSHIP INFERENCE GAMES

First, we introduce the fixed-target Membership Inference (MI) game. Then, we discuss different performance metrics to assess the power of an adversary. Finally, we characterise the optimal performances of adversaries using the Neyman-Pearson lemma.

2.1 Fixed-target MI Game

Let $\mathcal{M}$ be a randomised mechanism that takes as input a dataset D of n points belonging to $\mathcal{Z}$ and outputs $o \in \mathcal{O}$. In a Membership Inference (MI) game, an adversary attempts to infer whether a given target point $z^\star$ was included in the input dataset D of $\mathcal{M}$, given only access to an output $o \sim \mathcal{M}(D)$. A fixed-target MI game is presented in Algo 2. It is a game between two entities: the Crafter (Algo 1) and the adversary $\mathcal{A}_{z^\star}$. The MI game runs in multiple rounds. At each round t, the crafter samples a pair (o_t, b_t), where o_t is an output of the mechanism and b_t is the secret binary membership of $z^\star$. The adversary $\mathcal{A}_{z^\star}$ takes as input only o_t and outputs $\hat{b}_t$ trying to reconstruct b_t.

The specificity of the *fixed-target* MI game is that the target $z^\star$ is fixed throughout the game. Thus, the performance metrics of the attacker, i.e. the advantage and trade-off functions, are target-dependent. In

Algorithm 1 The Crafter

Input: Mechanism $\mathcal{M}$, Data distribution $\mathcal{D}$, #samples n, Target $z^\star$
Output: (o, b), where $o \in \mathcal{O}$ and $b \in \{0, 1\}$
Build a dataset $D \sim \bigotimes_{i=1}^{n} \mathcal{D}$
Sample $b \sim \text{Bernoulli}\left(\frac{1}{2}\right)$
if $b = 1$ **then**
    Sample $j \sim \mathcal{U}[n]$
    $D \leftarrow \text{Replace}(D, j, z^\star)$ ▷ Put $z^\star$ at position j
end if
Let $o \sim \mathcal{M}(D)$
Return (o, b)

Algorithm 2 Fixed-target MI Game

Input: Mechanism $\mathcal{M}$, Data distribution $\mathcal{D}$, #samples n, Target $z^\star$, Adversary $\mathcal{A}_{z^\star}$, Rounds T
Output: A list $L \in \{0, 1\}^T$, where $L_t = 1$ if the adversary succeeds at step t.
Initialise a empty list L of length T
for t = 1, . . . , T **do**
    Sample $(o_t, b_t) \sim \text{Crafter}(\mathcal{M}, \mathcal{D}, \text{n}, z^\star)$
    Sample $\hat{b}_t \sim \mathcal{A}_{z^\star}(o_t)$
    Set $L_t \leftarrow \mathbb{1}\left\{b_t = \hat{b}_t\right\}$
end for
Return L

contrast, in the MI game originally proposed in Experiment 1 of Yeom et al. (2018), the target $z^\star$ is sampled randomly at each step of the game, i.e. Step 3 in Experiment 1 of Yeom et al. (2018). In this case, the performance metrics of the attacker are averaged over the sampling of the target points and, thus, obfuscate the dependence of the leakage on each target point. To study the target-dependent hardness of MI games, we use this fixed-target formulation of Algo 2, which has also been proposed in Definition 3.3 of Ye et al. (2022).

A fixed-target MI game can also be seen as a hypothesis test. Here, the adversary tries to test the hypothesis "H_0: *The output o observed was generated from a dataset sampled i.i.d. from $\mathcal{D}$*", i.e. $b = 0$, versus "H_1: *The target point $z^\star$ was included in the input dataset producing the output o*", i.e. $b = 1$. We denote by $p_{\text{out}}(o \mid z^\star)$ and $p_{\text{in}}(o \mid z^\star)$ the distributions of the output o under H_0 and H_1 respectively.

2.2 Performance Metrics of the Attack

An adversary $\mathcal{A}_{z^\star}$ is a (possibly randomised) algorithm that takes as input o the output of the mechanism $\mathcal{M}$, and generates a guess $\hat{b} \sim \mathcal{A}_{z^\star}(o)$ trying to infer $b = \mathbb{1}\{z^\star \in D\}$. The performance of $\mathcal{A}_{z^\star}$ can be assessed either with aggregated metrics like the advantage, or with test-based metrics like a trade-off function.

The *accuracy* of $\mathcal{A}_{z^\star}$ is defined as $\text{Acc}_n(\mathcal{A}_{z^\star}) \triangleq \Pr[\mathcal{A}_{z^\star}(o) = b]$, where the probability is over any randomness in both the crafter and the adversary. The *advantage* of an adversary is the centred accuracy: $\text{Adv}_n(\mathcal{A}_{z^\star}) \triangleq 2\text{Acc}_n(\mathcal{A}_{z^\star}) - 1$. We can also define two errors from the hypothesis testing formulation. *The Type-I error*, aka False Positive Rate, is $\alpha_n(\mathcal{A}_{z^\star}) \triangleq \Pr[\mathcal{A}_{z^\star}(o) = 1 \mid b = 0]$. *The Type-II error*, aka the False Negative Rate, is $\beta_n(\mathcal{A}_{z^\star}) \triangleq \Pr[\mathcal{A}_{z^\star}(o) = 0 \mid b = 1]$. The *power* of the test is $1 - \beta_n(\mathcal{A}_{z^\star})$. In MI games, an adversary can threshold over a score function s to conduct the MI games, i.e. for $\mathcal{A}_{s,\tau,z^\star}(o) \triangleq \mathbb{1}\{s(o; z^\star) > \tau\}$ where s is a score function and τ is a threshold. We want to design score functions that maximise the power under a fixed significance level α, i.e. $\text{Pow}_n(s, \alpha, z^\star) \triangleq \max_{\tau \in T_\alpha} 1 - \beta_n(\mathcal{A}_{s,\tau,z^\star})$ where $T_\alpha \triangleq \{\tau \in \mathbb{R} : \alpha_n(\mathcal{A}_{s,\tau,z^\star}) \leq \alpha\}$. $\text{Pow}_n(s, \alpha, z^\star)$ is also called a *trade-off function*.

2.3 Optimal Adversaries & Defining Leakage

Given two data generating distributions p_0 and p_1 under hypotheses H_0 and H_1 respectively, no test can achieve better power than the Likelihood Ratio (LR) test (Neyman and Pearson, 1933). By considering the hypothesis testing formulation of the fixed-target MI game, the *log-Likelihood Ratio* (LR) score is

$$\ell_n(o; z^\star) \triangleq \log\left(\frac{p_n^{\text{in}}(o \mid z^\star)}{p_n^{\text{out}}(o \mid z^\star)}\right). \tag{1}$$

The LR-based adversary uses a threshold τ on the LR score, i.e. $\mathcal{A}_{\ell,z^\star,\tau}(o) \triangleq \mathbb{1}\{\ell_n(o; z^\star) > \tau\}$. We denote by $\mathcal{A}_{\text{Bayes},z^\star} \triangleq \mathcal{A}_{\ell,z^\star,0}$ the LR attacker with threshold $\tau = 0$. We provide Theorem 2.1 to characterise optimal adversaries under aggregated and test-based metrics.

Theorem 2.1 (Characterising Optimal Adversaries)**.**

(a) Optimal power: $\forall \alpha \in [0, 1], \forall$ *score* s, $\forall$ *target* $z^\star$, $\text{Pow}_n(\ell_n, \alpha, z^\star) \geq \text{Pow}_n(s, \alpha, z^\star)$.

(b) Largest advantage: $\forall$ *target* $z^\star$, $\forall$ *adversary* $\mathcal{A}_{z^\star}$, $\text{Adv}_n(\mathcal{A}_{\text{Bayes},z^\star}) \geq \text{Adv}_n(\mathcal{A}_{z^\star})$.

(c) Optimal advantage as TV distance:
$$\text{Adv}_n(\mathcal{A}_{\text{Bayes},z^\star}) = \text{TV}\left(p_n^{\text{out}}(. \mid z^\star) \,\middle\|\, p_n^{\text{in}}(. \mid z^\star)\right). \tag{2}$$

The detailed proof is in Appendix B.6. As a consequence of Theorem 2.1, we **define** the *target-dependent leakage* of $z^\star$, for mechanism $\mathcal{M}$ and data-generating distribution $\mathcal{D}$, as the advantage of the optimal Bayes attacker on $z^\star$, i.e. $\xi_n(z^\star, \mathcal{M}, \mathcal{D}) \triangleq \text{Adv}_n(\mathcal{A}_{\text{Bayes},z^\star})$.

Goal: Our main goal is to *quantify the target-dependent leakage $\xi_n(z^\star, \mathcal{M}, \mathcal{D})$ and trade-off functions for different mechanisms*, namely the empirical mean and its

variations. These two quantities may be intractable to characterise for any generic data-generating distribution. To get over this limitation, we use the asymptotic properties of the empirical mean as the main tool.

3 TARGET-DEPENDENT LEAKAGE OF EMPRICAL MEAN

We instantiate the fixed-target MI game with the empirical mean. We quantify the target-dependent leakage of a target $z^\star$ and characterise its dependence on the Mahalanobis distance between $z^\star$ and the data-generating distribution. Finally, we connect our results to the tracing literature (Sankararaman et al., 2009; Dwork et al., 2015) and explain the privacy onion phenomenon.

Notations and the asymptotic regime. We denote by $\mathcal{M}_n^{\text{emp}}$ the empirical mean mechanism. $\mathcal{M}_n^{\text{emp}}$ takes as input a dataset of size n of d-dimensional points, i.e. $D = \{Z_1, \dots, Z_n\} \in (\mathbb{R}^d)^n$, and outputs the exact empirical mean $\hat{\mu}_n \triangleq \frac{1}{n}\sum_{i=1}^n Z_i \in \mathbb{R}^d$. Let Φ represent the Cumulative Distribution Function (CDF) of the standard normal distribution, i.e. $\Phi(\alpha) \triangleq \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\alpha} e^{-t^2/2}\,\mathrm{d}t$ for $\alpha \in \mathbb{R}$. For a matrix M and a vector x, we write $\|x\|_M^2 \triangleq x^T M x$. Since the LR test can be non-tractable in general cases, we study the *asymptotic behaviour of the LR test*, when both the sample size n and the dimension d tend to infinity such that $d/n = \tau > 0$.

Assumptions on the data generating distribution $\mathcal{D}$. We suppose that the data-generating distribution is column-wise independent, i.e. $\mathcal{D} \triangleq \bigotimes_{j=1}^d \mathcal{D}_j$ and has a finite $(4+\delta)$-th moment for some small δ, i.e. there exists $\delta > 0$, such that $\mathbb{E}[Z^{4+\delta}] < \infty$. We denote by $\mu \triangleq (\mu_1, \dots, \mu_d) \in \mathbb{R}^d$ the mean of $\mathcal{D}$, and by $C_\sigma \triangleq \mathrm{diag}(\sigma_1^2, \dots, \sigma_d^2) \in \mathbb{R}^{d\times d}$ the covariance matrix. We recall that the Mahalanobis distance (Mahalanobis, 1936) of $z^\star$ with respect to $\mathcal{D}$ is $\|z^\star - \mu\|_{C_\sigma^{-1}}$.

Main result. Let us denote by $\ell_n(\hat{\mu}_n; z^\star, \mu, C_\sigma)$ the LR score for the empirical mean target-dependent MI game. By studying the asymptotic distribution of the LR score ℓ_n under H_0 and H_1, we characterise the exact asymptotic leakage and optimal trade-off functions.

Theorem 3.1 (Target-dependent leakage of the empirical mean)**.** *The asymptotic target-dependent leakage of $z^\star$ in the empirical mean is*

$$\lim_{n,d} \xi_n(z^\star, \mathcal{M}_n^{\text{emp}}, \mathcal{D}) = \Phi\left(\frac{\sqrt{m^\star}}{2}\right) - \Phi\left(-\frac{\sqrt{m^\star}}{2}\right). \quad (3)$$

The asymptotic trade-off function, achievable with threshold $\tau_\alpha = -\frac{m^\star}{2} + \sqrt{m^\star}\Phi^{-1}(1-\alpha)$, is

$$\lim_{n,d} \mathrm{Pow}_n(\ell_n, \alpha, z^\star) = \Phi\left(\Phi^{-1}(\alpha) + \sqrt{m^\star}\right). \quad (4)$$

Here, $m^\star \triangleq \lim_{n,d} \frac{1}{n}\|z^\star - \mu\|^2_{C_\sigma^{-1}}$, which we call the leakage score *of target $z^\star$.*

Proof Sketch. Theorem 3.1 is a consequence of Lemma C.6 presented in Appendix C.2. Lemma C.6 characterises the asymptotic distributions of the LR score as Gaussians under H_0 and H_1. We retrieve the asymptotic leakage and trade-off functions by using testing results between Gaussians. To prove Lemma C.6, (a) we rewrite the LR score with respect to the density of a centred normalised mean. Then, we use the Edgeworth asymptotic expansion (Theorem C.5) to get an expansion of the LR score. Finally, we conclude the asymptotic distribution of the LR test using the Lindeberg-Feller theorem (Theorem C.4). The detailed proofs are presented in Appendix C.2.

Target-dependent hardness. Theorem 3.1 shows that the optimal advantage and trade-off functions are increasing in the leakage score $m^\star$. As $z^\star$ has a higher Mahalanobis distance with respect to the data-generating distribution, the leakage score increases, making it easier to identify the target $z^\star$ in the MI game.

Empirical LR attack. Following the proof of Theorem 3.1, we show in Remark C.7 that the LR score

$$\ell_n \sim (z^\star - \mu)^T C_\sigma^{-1}(\hat{\mu}_n - \mu) - \frac{1}{2n}\|z^\star - \mu\|^2_{C_\sigma^{-1}} \quad (5)$$

asymptotically in n and d. Equation (5) shows that the LR score is a scalar product between $z^\star - \mu$ and $\hat{\mu}_n - \mu$, corrected by the precision matrix C_σ^{-1}. The optimal LR score uses the true mean μ and covariance matrix C_σ. A realistic attack should replace the true mean μ and covariance C_σ in Equation (5) with empirical estimates. This leads to a new covariance score $\ell_n^{\text{cov}}(\hat{\mu}_n; z^\star) = (z^\star - \hat{\mu}_0)^T \hat{C}_0^{-1}(\hat{\mu}_n - \hat{\mu}_0) - \frac{1}{2n}\|z^\star - \hat{\mu}_0\|^2_{\hat{C}_0^{-1}}$, where $\hat{\mu}_0 = \frac{1}{n_0}\sum_{i=1}^{n_0} Z_i^{\text{ref}}$ and $\hat{C}_0 = \frac{1}{n_0}\sum_{i=1}^{n_0} Z_i^{\text{ref}}(Z_i^{\text{ref}})^T$ are estimated using reference points sampled independently from $\mathcal{D}$. The decrease in the power of the covariance attack compared to the LR attack depends on the accuracy of the estimators $\hat{\mu}_0$ and $\hat{C}_0$.

Connection to Sankararaman et al. (2009). For Bernoulli distributions, Sankararaman et al. (2009) shows that the hardness of "average-target" MI game depend on the ratio $\tau \triangleq d/n$ (Sankararaman et al., 2009, Section T2.1). Since $\mathbb{E}_{z^\star\sim\mathcal{D}}[m^\star] = \lim_{n,d}\frac{d}{n} = \tau$, our results retrieve the "averaged" results of (Sankararaman et al., 2009). In addition, Sankararaman et al. (2009) uses an analysis tailored only for Bernoulli distributions. Our analysis generalises their results to the target-dependent setting and to any data-generating distribution with a finite fourth moment.

Connection to the scalar product attack. Dwork et al. (2015) proposes a scalar product attack for tracing the empirical mean that thresholds over the score $s^{\text{scal}}(\hat{\mu}_n; z^\star, z^{\text{ref}}) \triangleq (z^\star - z^{\text{ref}})^T\hat{\mu}_n$. The intuition behind this attack is to compare the target-output cor-

relation $(z^\star)^T\hat{\mu}_n$ with a reference-output correlation $(z^{\text{ref}})^T\hat{\mu}_n$. The analysis of (Dwork et al., 2015) shows that with only one reference point $z^{\text{ref}} \sim \mathcal{D}$, and even for noisy estimates of the mean, the attack is able to trace the data of some individuals in the regime $d \sim n^2$. *Our asymptotic analysis shows that the LR attack, i.e. the optimal attack, is also a scalar-product attack* (Equation (5)), but *corrected for the geometry of the data using the inverse covariance matrix.*

Explaining the privacy onion effect. Removing a layer of outliers is equivalent to sampling from a new data-generating distribution, with a smaller variance. In this new data-generating distribution with smaller variance, the points which are not removed will naturally have an increased Mahalanobis distance. Thus, removing a layer of outlier points yields a layer of newly exposed target points. Hence, the Mahalanobis score explains the privacy onion effect (Carlini et al., 2022b).

Inherent privacy of the empirical mean. Under our specific threat model of MI games, the empirical mean already imposes a trade-off between the Type-I and Type-II errors for *any adversary.* This means that, if an auditor uses the fixed-target MI game with some target $z^\star$ to audit the privacy of the empirical mean, the auditor would conclude that the empirical mean is $\sqrt{m^\star}$-Gaussian DP (Dong et al., 2019), or equivalently (ϵ, δ)-DP where for all $\epsilon \geq 0$, $\delta(\epsilon) = \Phi\left(-\frac{\epsilon}{\sqrt{m^\star}} + \frac{\sqrt{m^\star}}{2}\right) - e^\epsilon \Phi\left(-\frac{\epsilon}{\sqrt{m^\star}} - \frac{\sqrt{m^\star}}{2}\right)$. The result is a direct consequence of Equation (4) and (Dong et al., 2019, Corollary 2.13).

MI games with Z-estimators and relation to influence functions. The main technical tool used to provide an asymptotic expansion of the LR score is the "asymptotic normality" of the empirical mean, i.e. the Edgeworth expansion in Theorem C.5. The empirical mean estimator is only an instance of a more general class of estimators enjoying the asymptotic normality property, called Z-estimators (Vaart, 1998). Now, suppose we are interested in estimating a parameter θ that is a functional of the distribution of observations $X_1, \ldots, X_n$. A popular method to construct an estimator $\hat{\theta}_n = \hat{\theta}_n(X_1, \ldots, X_n)$ is to satisfy

$$\Psi_n(\theta) \triangleq \frac{1}{n}\sum_{i=1}^{n} \psi_\theta(X_i) = 0. \tag{6}$$

Here, ψ_θ are known functions. The class of Z-estimators retrieves the empirical mean with $\psi_\theta(X_i) \triangleq X_i - \theta$ and the median with $\psi_\theta(X_i) \triangleq \text{sign}(X_i - \theta)$. The class of Z-estimators also recovers many other estimators, such as maximum likelihood estimators, least square estimators, and empirical risk minimisers.

Under technical conditions on the data-generating distribution and the "regularity" of the function ψ_θ, it is possible to show that θ_n converges in probability to a parameter θ_0 , i.e. a zero of the function $\Psi(\theta) \triangleq \mathbb{E}_X(\psi_\theta(X))$. Also, for any Z-estimator, Theorem 5.21 in (Van der Vaart, 2000) shows that

$$\hat{\theta}_n - \theta_0 = -V_{\theta_0}^{-1}\frac{1}{n}\sum_{i=1}^{n}\psi_{\theta_0}(X_i) + o_p\left(\frac{1}{\sqrt{n}}\right), \tag{7}$$

where V_θ is a non-singular derivative matrix of the map $\theta \to \Psi(\theta)$ at θ_0. Generally, $I_{\theta_0}(X_i) \triangleq V_{\theta_0}^{-1}\psi_{\theta_0}(X_i)$ is called the influence function. Thus, Equation (7) shows that, asymptotically, any Z-estimator can be thought of *as the empirical mean of its influence functions.* Using our target-dependent analysis of empirical mean MI games, it is direct to provide a new covariance score and a new canary selection strategy for all Z estimators. For the score, the covariance attack becomes

$$(I_{\theta_0}(X^\star) - \theta_0)^T V_{\theta_0}^{-1}(\hat{\theta}_n - \theta_0) - \frac{1}{2n}\|I_{\theta_0}(X^\star) - \theta_0\|^2_{V_{\theta_0}^{-1}}.$$

Similarly, to chose canaries, i.e. targets that are easy to identify, find points for which the estimated Mahalanobis distance of the influence functions at $X^\star$ is high, i.e. $\|I_{\theta_0}(X^\star) - \theta_0\|_{V_{\theta_0}^{-1}}$. We leave it for future work to provide a rigorous statement of when these statements are correct, i.e. rigorous conditions on the data-generating distribution and regularity of ψ_θ.

4 IMPACT OF PRIVACY DEFENCES & MISSPECIFICATION

We quantify the effect of adding noise and sub-sampling on the leakage of the empirical mean. Both defences act like contractions of the leakage score. We also study the effect of target misspecification. The detailed proofs for this section are presented in Appendix C.

I. Adding Gaussian noise. We denote by $\mathcal{M}_n^\gamma$ the mechanism releasing the noisy empirical mean of a dataset using the Gaussian mechanism (Dwork and Roth, 2014). $\mathcal{M}_n^\gamma$ takes as input a dataset of size n of d-dimensional points, i.e. $D = \{Z_1, \ldots, Z_n\} \in (\mathbb{R}^d)^n$, and outputs the noisy mean $\tilde{\mu}_n \triangleq \frac{1}{n}\sum_{i=1}^n Z_i + \frac{1}{\sqrt{n}}N_d \in \mathbb{R}^d$, where $N_d \sim \mathcal{N}(0, C_\gamma)$ such that $\gamma = (\gamma_1, \ldots, \gamma_d) \in \mathbb{R}^d$ and $C_\gamma = \text{diag}\left(\gamma_1^2, \ldots, \gamma_d^2\right) \in \mathbb{R}^{d\times d}$. Similar to Section 3, we assume that the data-generating distribution $\mathcal{D}$ is colon-wise independent, has a mean $\mu \triangleq (\mu_1, \ldots, \mu_d) \in \mathbb{R}^d$, a covariance matrix $C_\sigma \triangleq \text{diag}(\sigma_1^2, \ldots, \sigma_d^2) \in \mathbb{R}^{d\times d}$, and a finite $(4+\delta)$-th moment for $\delta > 0$.

The output of $\mathcal{M}_n^\gamma$ could be re-written as $\tilde{\mu}_n = \frac{1}{n}\sum_{i=1}^n (Z_i + N_i) = \frac{1}{n}\sum_{i=1}^n \tilde{Z}_i$, where $\tilde{Z}_i \sim \tilde{\mathcal{D}}$ s.t. $\tilde{\mathcal{D}} \triangleq \mathcal{D} \bigotimes \mathcal{N}(0, C_\gamma)$. This means that $\mathcal{M}_n^\gamma$ could be seen as *the exact empirical mean of n i.i.d samples*

from a new data-generating distribution $\tilde{\mathcal{D}}$. The results of Section 3 directly apply to $\mathcal{M}_n^\gamma$, by replacing $\mathcal{D}$ by $\tilde{\mathcal{D}}$. The noisy leakage score $\tilde{m}_\gamma^\star$ is now defined as $\tilde{m}_\gamma^\star \triangleq \lim_{n,d} \frac{1}{n}\|z^\star - \mu\|^2_{(C_\sigma + C_\gamma)^{-1}}$. Directly using the results of Section 3 for $\tilde{\mathcal{D}}$ yields the following theorem.

Theorem 4.1 (Target-dependent leakage of the noisy empirical mean)**.** *The asymptotic target-dependent leakage of $z^\star$ in the noisy empirical mean is*

$$\lim_{n,d} \xi_n(z^\star, \mathcal{M}_n^\gamma, \mathcal{D}) = \Phi\left(\frac{\sqrt{\tilde{m}_\gamma^\star}}{2}\right) - \Phi\left(-\frac{\sqrt{\tilde{m}_\gamma^\star}}{2}\right). \quad (8)$$

The asymptotic trade-off function, achievable with threshold $\tau_\alpha = -\frac{\tilde{m}_\gamma^\star}{2} + \sqrt{\tilde{m}_\gamma^\star}\Phi^{-1}(1-\alpha)$, is

$$\lim_{n,d} \mathrm{Pow}_n(\tilde{\ell}_n, \alpha, z^\star) = \Phi\left(\Phi^{-1}(\alpha) + \sqrt{\tilde{m}_\gamma^\star}\right). \quad (9)$$

Theorem 4.1 shows that the Gaussian Mechanism acts by increasing the variance of the data-generating distribution, thus decreasing the Mahalanobis distance of target points and their leakage score.

II. Effect of sub-sampling. We consider the *empirical mean with sub-sampling* mechanism (Balle et al., 2018) $\mathcal{M}_n^{\mathrm{sub},\rho}$ that uniformly sub-samples k_n rows without replacement from the original dataset, and then computes the exact empirical mean of the sub-sampled rows. $\mathcal{M}_n^{\mathrm{sub},\rho}$ takes as input a dataset $D = \{Z_1, \ldots, Z_n\} \in (\mathbb{R}^d)^n$ and outputs $\hat{\mu}_{k_n}^{\mathrm{sub}} \triangleq \frac{1}{k_n}\sum_{i=1}^n Z_i \mathbb{1}\{\varsigma(i) \leq k_n\}$. Here, $k_n \triangleq \rho n$, $0 < \rho < 1$ and $\varsigma \sim^{\mathrm{unif}} S_n$ is a permutation sampled uniformly from S_n the set of permutations of $\{1 \ldots, n\}$, and independently from $(Z_1, \ldots, Z_n)$. We get the following result by adapting the proofs in Sec. 3.

Theorem 4.2 (Leakage for sub-sampling)**.** *The asymptotic target-dependent leakage of $z^\star$ in $\mathcal{M}_{n,\rho}^{\mathrm{sub}}$ is*

$$\lim_{n,d} \xi_n(z^\star, \mathcal{M}_{n,\rho}^{\mathrm{sub}}, \mathcal{D}) = \Phi\left(\frac{\sqrt{\rho m^\star}}{2}\right) - \Phi\left(-\frac{\sqrt{\rho m^\star}}{2}\right).$$

The asymptotic trade-off function, achievable with threshold $\tau_\alpha = -\frac{\rho m^\star}{2} + \sqrt{\rho m^\star}\Phi^{-1}(1-\alpha)$, is

$$\lim_{n,d} \mathrm{Pow}_n(\ell_{n,\rho}^{\mathrm{sub}}, \alpha, z^\star) = \Phi\left(z_\alpha + \sqrt{\rho m^\star}\right). \quad (10)$$

Theorem 4.2 shows that the sub-sampling mechanism acts by increasing the number of "effective samples" from n to n/ρ, thus decreasing the leakage score.

III. Misspecifying the target. Suppose that the adversary has a misspecified target z^{targ}, different from the real $z^\star$ used in the fixed-target MI game (Algorithm 2). The adversary $\mathcal{A}_{\mathrm{miss}}$ then builds the "optimal" LR test *tailored* for z^{targ}, i.e. $\ell_n(\hat{\mu}_n; z^{\mathrm{targ}}, \mu, C_\sigma)$. The misspecified adversary is sub-optimal but can still leak enough information depending on the amount of misspecification. Now, we quantify the sub-optimality of the misspecified adversary, which we define as a measure of leakage similarity between z^{targ} and $z^\star$.

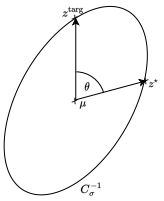


Figure 1: Effect of misspecification depends on the angle θ between $z^\star - \mu$ and $z^{\mathrm{targ}} - \mu$, corrected by C_σ^{-1}

Theorem 4.3 (Leakage of a misspecified adversary)**.** *The advantage of the misspecified adversary is*

$$\lim_{n,d} \mathrm{Adv}_n(\mathcal{A}_{\mathrm{miss}}) = \Phi\left(\frac{|m^{\mathrm{scal}}|}{2\sqrt{m^{\mathrm{targ}}}}\right) - \Phi\left(-\frac{|m^{\mathrm{scal}}|}{2\sqrt{m^{\mathrm{targ}}}}\right).$$

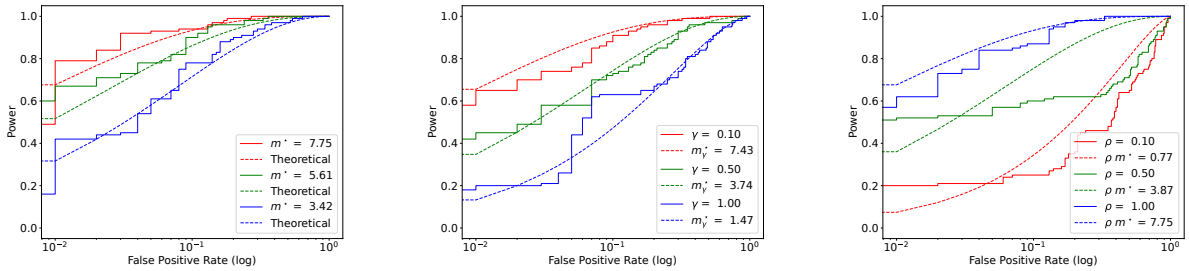
Here, $m^{\mathrm{scal}} \triangleq \lim_{n,d} \frac{1}{n}(z^{\mathrm{targ}} - \mu)^T C_\sigma^{-1}(z^\star - \mu)$ and $m^{\mathrm{targ}} \triangleq \lim_{n,d} \frac{1}{n}\|z^{\mathrm{targ}} - \mu\|^2_{C_\sigma^{-1}}$.

If the adversary specified well the target by using $z^\star$ rather than z^{targ}, then they achieve the optimal asymptotic advantage $\lim_{n,d} \xi_n(z^\star, \mathcal{M}_n^{\mathrm{emp}}, \mathcal{D})$ of Equation 3. Theorem 4.3 *quantifies the sub-optimality of the misspecified adversary*, which is $\Delta(z^{\mathrm{targ}}, z^\star) = \lim_{n,d} \xi_n(z^\star, \mathcal{M}_n^{\mathrm{emp}}, \mathcal{D}) - \mathrm{Adv}_n(\mathcal{A}_{\mathrm{miss}})$. This quantity depends on the comparison between $m^\star$ and $|m^{\mathrm{scal}}|/\sqrt{m^{\mathrm{targ}}}$. By the Cauchy Schwartz inequality, $|m^{\mathrm{scal}}| \leq \sqrt{m^{\mathrm{targ}} m^\star}$, which means that $\Delta(z^{\mathrm{targ}}, z^\star) \geq 0$. The misspecified attack is still strong as long as $\sqrt{m^{\mathrm{targ}} m^\star} - |m^{\mathrm{scal}}| = \sqrt{m^{\mathrm{targ}} m^\star}(1 - |\cos(\theta)|)$ stays small. We geometrically illustrate θ in Figure 1.

5 BEYOND EMPIRICAL MEAN: WHITE-BOX ATTACK & CANARY SELECTION ON GRADIENT DESCENTS

We attack supervised learning gradient descent algorithms. The main observation is that gradient descent algorithms operate by sequentially updating a parameter estimate θ_t *in the direction of the empirical mean of gradients*. If an adversary has access to all the intermediates parameters $\{\theta_t\}_t$, i.e. the white-box federated learning setting, attacking gradient descent algorithms reduces to attacking the empirical mean mechanism.

The threat model. We suppose that a input dataset contains n examples of features and label pairs, i.e. $D \triangleq \{(x_i, y_i)\}_{i=1}^n$. The goal is to find $\theta^\star \triangleq \arg\min_{\theta \in \mathbb{R}^d} \frac{1}{n}\sum_{i=1}^n \ell(f_\theta(x_i), y_i)$ the best parameter which explains the dataset D with respect to a loss function ℓ. Here, we focus on gradient descent algorithms. Gradient Descent algorithms start with an initial parameter $\theta_0 \in \mathbb{R}^d$, and then up-



(a) Effect of $m^\star$ (b) Effect of γ (c) Effect of ρ

Figure 2: Experimental validation of the theoretical results and impacts of $m^\star$, noise, and sub-sampling ratio on leakage. Dotted lines are for theoretical bounds and solid lines for empirical ones.

Algorithm 3 Mahalanobis canary-choosing strategy

Input: $\{(x_1, y_1), \dots, (x_{n_r}, y_{n_r})\}$ reference points, $\{(x_1, y_1), \dots, (x_{n_c}, y_{n_c})\}$ candidate canaries
Step1: Estimate $\hat{\mu}_0$ and $\hat{C}_0$
Initialise weights and biases of the model, i.e. θ^0.
for $i = 1, \cdots, n_r$ **do**
  Compute $g_i = \nabla_{\theta_0} \ell(f_{\theta_0}(x_i), y_i)$
end for
Compute $\hat{\mu}_0 = \frac{1}{n}\sum_i^n g_i$ and $\hat{C}_0 = \frac{1}{n}\sum_i^n g_i g_i^T$
Return $(\hat{\mu}_0, \hat{C}_0)$
Step2: Estimate the Mahalanobis score
for $k = 1, \cdots, n_c$ **do**
  Compute $g_k = \nabla_{\theta_0} \ell(f_{\theta_0}(x_k), y_k)$
  Estimate $m_k^\star = \|g_k - \hat{\mu}_0\|^2_{\hat{C}_0^{-1}}$
end for
Return $(x^\star, y^\star) = \arg\max_{k=1}^{n_c} m_k^\star$

date sequentially the parameter at each step t by $\theta_t \triangleq \theta_{t-1} - \eta \nabla_{\theta_{t-1}} Q(\theta_{t-1})$, where η is the learning rate, and $Q(\theta_t)$ is a quantity that aggregates the gradients on some input samples. For example, in batch gradient descent $\nabla_{\theta_t} Q(\theta_t) \triangleq \frac{1}{n}\sum_{i=1}^n \nabla_{\theta_t} \ell(f_{\theta_t}(x_i), y_i)$ is the mean of gradients on to the whole dataset. For DP-SGD (Abadi et al., 2016), $\nabla_{\theta_t} Q(\theta_t) \triangleq \left(\frac{1}{|B|}\sum_{i\in B} \text{Clip}_C\left[\nabla_{\theta_t} \ell(f_{\theta_t}(x_i), y_i)\right]\right) + \mathcal{N}\left(0, \gamma^2 C^2 I_d\right)$, where B is a batch uniformly sampled, $\text{Clip}_C(x) \triangleq \min\{1, C/\|x\|\}x$ is the clipping function, $C > 0$ is a clipping bound and $\gamma > 0$ is the noise magnitude.

We instantiate the fixed-target MI game with the gradient descent training algorithm, which takes as input the private dataset D and produces the sequence $\{\theta_t\}_{t=1}^T$. This is called the white-box federated learning setting (Nasr et al., 2023). We provide more details on the definition of this setting, its importance, and its relation to federated learning in Appendix D.

The covariance attack. We present our covariance attack in Algorithm 4. Given the target's gradient $g_t^\star$ and the batch gradient $g_{\text{batch}}^t \triangleq \frac{\theta_{t+1} - \theta_t}{\eta}$, the covariance attack at step t uses the empirical LR score of Section 3 to compute the covariance score s_t. The attack uses

Algorithm 4 The covariance attack

Input: Estimated $(\hat{\mu}_0, \hat{C}_0)$ from Algorithm 3, target $z^\star$, learning rate η, batch size b.
for $t = 1, \cdots, L$ **do**
  Set $g_t^\star = \nabla_{\theta_t} \ell(f_{\theta_t}(x^\star), y^\star)$
  Set $g_{\text{batch}}^t = (\theta_{t+1} - \theta_t)/\eta$
  Compute the covariance score $s_t = (g_t^\star - \hat{\mu}_0)^T \hat{C}_0^{-1} (g_{\text{batch}}^t - \hat{\mu}_0) - \frac{1}{2b}\|g_t^\star - \hat{\mu}_0\|^2_{\hat{C}_0^{-1}}$
end for
Return $\sum_{t=1}^L s_t$

the estimated empirical mean $\hat{\mu}_0$ and covariance C_0 of gradients over some reference points. To avoid the computational burden, the same estimates $(\hat{\mu}_0, C_0)$ are used for the attack over one epoch, i.e. one pass over the dataset. At the end of the epoch, the final score is the sum of step-wise scores s_t. When the inverse of the covariance matrix is well estimated, the covariance attack improves on the scalar product score, which is the state-of-the-art score in white box attacks (Maddock et al., 2022; Nasr et al., 2023; Steinke et al., 2023; Andrew et al., 2023).

The Mahalanobis canaries. We present our canary selection strategy in Algorithm 3. Algorithm 3 takes as input candidate "canaries", and outputs the "easiest" point to attack between the proposed candidates. To do so, Algorithm 3 estimates the Mahalanobis score of gradients using reference samples. In addition to being an optimal strategy, Algorithm 3 is the first strategy generating "in-distribution" canaries that do not hurt the accuracy of a model trained. In appendix D, we compare our Mahalanobis canaries to the different heuristics used in the white-box literature (Jagielski et al., 2020; Maddock et al., 2022; Nasr et al., 2023).

6 EXPERIMENTAL ANALYSIS

First, we empirically validate the theoretical analysis on synthetic data. Then, we test our covariance attack and canary selection strategy in the white-box federated learning setting on gradient descent on real datasets.

6.1 Experiments on synthetic data

We test: *How tight is the leakage analysis in Theorem 3.1, 4.1, and 4.2?*

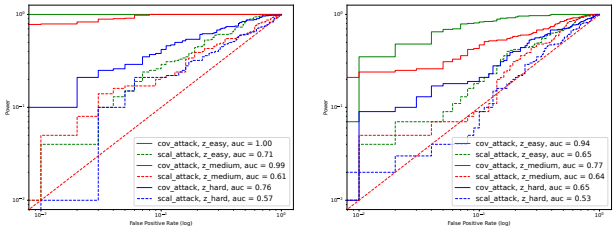
Experimental setup. We take $n = 1000$, $\tau = 5$, and thus, $d = 5000$. The data-generating distribution $\mathcal{D}$ is a d dimensional Bernoulli, with parameter $p \in [0,1]^d$. The three mechanisms considered are $\mathcal{M}_n^{\text{emp}}$, $\mathcal{M}_n^{\gamma}$ and $\mathcal{M}_{n,d}^{\text{sub}}$. The adversaries chosen for each mechanism are the thresholding adversaries based on the asymptotic approximations of the LR tests. Finally, we choose three target data points in $\{0,1\}^d$. (a) The *easiest point to attack* $z^\star_{\text{easy}} \triangleq (\mathbb{1}\{p_i \leq 1/2\})_{i=1}^d$ is the point with the highest Mahalanobis distance with respect to p. (b) The *hardest point to attack* is $z^\star_{\text{hard}} \triangleq (\mathbb{1}\{p_i > 1/2\})_{i=1}^d$, that has the binary coordinates closest to p. (c) A *medium point to attack* $z^\star_{\text{med}}$ is randomly sampled from the data-generating distribution $\text{Bern}(p)$, for which the Mahalanobis distance and the leakage score are of orders d and $\tau = d/n$. We simulate a fixed-target MI game 1000 times for each mechanism and fixed target point. We plot the empirical ROC curve in solid lines in Figure 2. The theoretical trade-off functions of Equations (4), (9) and (10) are in dotted lines. Further details are in Appendix E.

Results and discussions. Figure 2 shows that (a) the power of the LR test uniformly increases with an increase in $m^\star$ and the sub-sampling ratio ρ, and uniformly decreases with an increase in the noise variance γ^2. (b) Figure 2 validates that our theoretical analysis tightly captures the target-dependent hardness of MI games and the effect of privacy-preserving mechanisms on the experimental ROC curves.

6.2 Attacking in the white-box federated learning setting

We test: *Does the covariance attack improve over the scalar product attack? Does the Mahalanobis leakage score explain the target-dependent hardness of MI games on real datasets?*

Experimental setup. We attack two models. We train a single linear layer ($\text{Linear}(28 \times 28, 10)$) on Fashion MNIST (FMNIST) (Xiao et al., 2017). Thus, the number of weights and biases in the model is $d_F = 7850$. We train a Convolutional Neural Net (CNN) (LeCun et al., 2015) with three convolutional layers and a final linear layer on CIFAR10 (Krizhevsky et al., 2009). The number of weights and biases in CNN is $d_C = 18786$, while the last linear layer has $d_L = 4080$ parameters. We use batched SGD with a batch size 64, learning rate 10^{-3}, and cross-entropy loss. We attack the models in a white-box FL setting. First, we run Algorithm 3 to estimate the empirical mean and covariance, using $n_r = 1000$ reference points from training data. Then,



(a) Logistic reg. on FMNIST (b) CNN on CIFAR10

Figure 3: Covariance and scalar product attacks.

we estimate the Mahalanobis score for every point in the training data. Finally, we chose as targets the points in the training data with the highest, medium and lowest Mahalanobis scores, i.e. z_{easy}, z_{med} z_{hard}, respectively. We run both our covariance attack and the scalar product attack on these three points. The scalar product attack replaces the score s_t in Line 5 of Algo. 4 with $s_t^{\text{scal}} = (g_t^\star)^T g_{\text{batch}}^t$. Both attacks are run *only* on one epoch of SGD, i.e. one loop over the training data. For FMNIST, the attack is implemented with the *full* gradient of the loss. For CIFAR10, we only attack the last linear layer of the model, leading to $d_L \times d_L$ covariance matrix rather than $d_C \times d_C$. This improves our attack's time and space complexity by storing and inverting a smaller matrix. It still maintains the strength of the tracing attack since d_L is significantly larger than the batch size. We show the ROC curves of the two attacks against easy, medium and hard targets of FMNIST and CIFAR10 in Fig. 3.

Results and discussion. Figure 3 shows that (a) The covariance attack improves on the scalar product attack. (b) Points with high Mahalanobis scores are easier to attack than points with low Mahalanobis scores for both the datasets and models. (c) It is enough to run the covariance attack over one epoch and the last layer of a model. Also, the in-distribution Mahalanobis canaries, i.e. z_{easy}, leak enough to be easily identified.

7 CONCLUSION & FUTURE WORK

We study fixed-target MI games and characterise the target-dependent hardness of MI games for the empirical mean and its variants. We show that the hardness is captured exactly by the Mahalanobis distance of the target to the data-generating distribution (Table 1). Our generic analysis captures the impact of different variants, like adding Gaussian noise, sub-sampling and misspecification. Our analysis generalises different results from tracing literature and explains novel phenomena observed experimentally in MI attacks. We also provide a novel covariance attack and canary strategy. As a perspective, it would be natural to generalise our analysis to Z-estimators and use the insights to design new black-box attacks using influence functions.

Acknowledgements

This work has been partially supported by the THIA ANR program "AI PhD@Lille". D. Basu acknowledges the Inria-Kyoto University Associate Team "RELIANT" for supporting the project, the ANR JCJC for the REPUBLIC project (ANR-22-CE23-0003-01), and the PEPR project FOUNDRY (ANR23-PEIA-0003). We thank Timothée Mathieu for the interesting conversations. We also thank Philippe Preux and Vianney Perchet for their supports.

References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. (2016). Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pages 308–318.

Andrew, G., Kairouz, P., Oh, S., Oprea, A., McMahan, H. B., and Suriyakumar, V. (2023). One-shot empirical privacy estimation for federated learning. *arXiv preprint arXiv:2302.03098*.

Balle, B., Barthe, G., and Gaboardi, M. (2018). Privacy amplification by subsampling: Tight analyses via couplings and divergences. *Advances in neural information processing systems*, 31.

Carlini, N., Chien, S., Nasr, M., Song, S., Terzis, A., and Tramer, F. (2022a). Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 1897–1914. IEEE.

Carlini, N., Jagielski, M., Zhang, C., Papernot, N., Terzis, A., and Tramer, F. (2022b). The privacy onion effect: Memorization is relative. *Advances in Neural Information Processing Systems*, 35:13263–13276.

Dong, J., Roth, A., and Su, W. J. (2019). Gaussian differential privacy. *arXiv preprint arXiv:1905.02383*.

Dwork, C. and Roth, A. (2014). The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407.

Dwork, C., Smith, A., Steinke, T., and Ullman, J. (2017). Exposed! a survey of attacks on private data. *Annual Review of Statistics and Its Application*, 4:61–84.

Dwork, C., Smith, A., Steinke, T., Ullman, J., and Vadhan, S. (2015). Robust traceability from trace amounts. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 650–669. IEEE.

Homer, N., Szelinger, S., Redman, M., Duggan, D., Tembe, W., Muehling, J., Pearson, J. V., Stephan, D. A., Nelson, S. F., and Craig, D. W. (2008). Resolving individuals contributing trace amounts of dna to highly complex mixtures using high-density snp genotyping microarrays. *PLoS genetics*, 4(8):e1000167.

Humphries, T., Oya, S., Tulloch, L., Rafuse, M., Goldberg, I., Hengartner, U., and Kerschbaum, F. (2023). Investigating membership inference attacks under data dependencies. In *2023 IEEE 36th Computer Security Foundations Symposium (CSF)*, pages 473–488. IEEE.

Jagielski, M., Ullman, J., and Oprea, A. (2020). Auditing differentially private machine learning: How private is private sgd? *Advances in Neural Information Processing Systems*, 33:22205–22216.

Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images.

LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.

Leemann, T., Pawelczyk, M., and Kasneci, G. (2023). Gaussian membership inference privacy. *arXiv preprint arXiv:2306.07273*.

Maddock, S., Sablayrolles, A., and Stock, P. (2022). Canife: Crafting canaries for empirical privacy measurement in federated learning. *arXiv preprint arXiv:2210.02912*.

Mahalanobis, P. C. (1936). On the generalised distance in statistics. In *Proceedings of the National Institute of Science of India*, volume 12, pages 49–55.

Murakonda, S. K. and Shokri, R. (2020). Ml privacy meter: Aiding regulatory compliance by quantifying the privacy risks of machine learning. *arXiv preprint arXiv:2007.09339*.

Nasr, M., Hayes, J., Steinke, T., Balle, B., Tramèr, F., Jagielski, M., Carlini, N., and Terzis, A. (2023). Tight auditing of differentially private machine learning. *arXiv preprint arXiv:2302.07956*.

Neyman, J. and Pearson, E. S. (1933). Ix. on the problem of the most efficient tests of statistical hypotheses. *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, 231(694-706):289–337.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Petrov, V. V. (2012). *Sums of independent random variables*, volume 82. Springer Science & Business Media.

Sankararaman, S., Obozinski, G., Jordan, M. I., and Halperin, E. (2009). Genomic privacy and limits of individual detection in a pool. *Nature genetics*, 41(9):965–967.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. (2017). Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*, pages 3–18. IEEE.

Song, S. and Marn, D. (2020). Introducing a new privacy testing library in tensorflow.

Steinke, T., Nasr, M., and Jagielski, M. (2023). Privacy auditing with one (1) training run. *arXiv preprint arXiv:2305.08846*.

Steinke, T. and Ullman, J. (2020). The pitfalls of averagecase differential privacy.

Vaart, A. W. v. d. (1998). *Asymptotic Statistics*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press.

Van der Vaart, A. W. (2000). *Asymptotic statistics*, volume 3. Cambridge university press.

Xiao, H., Rasul, K., and Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms.

Ye, J., Maddi, A., Murakonda, S. K., Bindschaedler, V., and Shokri, R. (2022). Enhanced membership inference attacks against machine learning models. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 3093–3106.

Yeom, S., Giacomelli, I., Fredrikson, M., and Jha, S. (2018). Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*, pages 268–282. IEEE.

Zhang, B., Yu, R., Sun, H., Li, Y., Xu, J., and Wang, H. (2020). Privacy for all: Demystify vulnerability disparity of differential privacy against membership inference attack. *arXiv preprint arXiv:2001.08855*.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. Yes
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. Yes
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. Yes

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. Yes
   (b) Complete proofs of all theoretical results. Yes
   (c) Clear explanations of any assumptions. Yes

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). Yes
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). Yes
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). Yes
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). Yes

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. Yes
   (b) The license information of the assets, if applicable. Yes
   (c) New assets either in the supplemental material or as a URL, if applicable. Not Applicable
   (d) Information about consent from data providers/curators. Not Applicable
   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. Not Applicable

5. If you used crowdsourcing or conducted research with human subjects, check if you include:
   (a) The full text of instructions given to participants and screenshots. Not Applicable
   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. Not Applicable
   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. Not Applicable

A OUTLINE OF THE APPENDIX

The appendices are organised as follows:

- The connection between fixed-target and "average-target" games is discussed in Appendix B. The LR test of (Sankararaman et al., 2009) is revisited in Section B.4 and the scalar product attack of (Dwork et al., 2015) in Section B.5. Also, a proof of Theorem 2.1 is given in Section B.6.
- All the missing proofs of Lemma C.6, Theorem 3.1, Theorem 4.2 and Theorem 4.3 are given in Appendix C. There, we introduce our main technical tools from asymptotic statistics.
- Additional details on the white-box federated learning setting and discussions on related works are presented in Appendix D.
- Extended experiments are presented in Appendix E.

B TARGET-DEPENDENT V.S AVERAGE-TARGET MI GAMES

In this section, we present the average-target MI game of (Yeom et al., 2018) (Algorithm 6), and connect it to the fixed-target MI game (Algorithm 2). We also instantiate the average-target MI game with the empirical mean mechanism and present the two main attacks from this literature: the likelihood ratio (LR) test of (Sankararaman et al., 2009) and the scalar product attack of (Dwork et al., 2015). Finally, we provide a proof of Theorem 2.1.

B.1 The Average-target MI Game

An average-target MI game (Algorithm 6, Experiment 1 of (Yeom et al., 2018)) is a game between two entities: the Target-crafter (Algorithm 5) and the adversary $\mathcal{A}$. The MI game runs in multiple rounds. At each round t, the target crafter samples a tuple $(z_t^\star, o_t, b_t)$, where $z_t^\star$ is a target point, o_t is an output of the mechanism and b_t is the secret binary membership of $z^\star$. The adversary $\mathcal{A}$ takes as input $(z_t^\star, o_t)$ and outputs $\hat{b}_t$ trying to reconstruct b_t. In an average-target MI game, the adversary attacks a different target point $z_t^\star$ at each round t of the game.

B.2 Performance Metrics in the Average-target MI Game

An adversary $\mathcal{A}$ is a (possibly randomised) function that takes as input the pair $(z^\star, o)$ generated by the Target crafter (Algorithm 5) and outputs a guess $\hat{b} \sim \mathcal{A}(z^\star, o)$ trying to infer b. The performance of $\mathcal{A}$ can be assessed either with aggregated metrics like the accuracy and the advantage, or with test-based metrics like Type I error, Type II error, and trade-off functions.

The accuracy of $\mathcal{A}$ is defined as $\mathrm{Acc}_n(\mathcal{A}) \triangleq \Pr[\mathcal{A}(z^\star, o) = b]$, where the probability is over the generation of $(z^\star, o, b)$ using Algorithm 5 with input $(\mathcal{M}, \mathcal{D}, n)$. *The advantage of an adversary* is the re-centred accuracy $\mathrm{Adv}_n(\mathcal{A}) \triangleq 2\mathrm{Acc}_n(\mathcal{A}) - 1$. We can also define two errors from the hypothesis testing formulation. *The Type I error*, also called the False Positive Rate, is $\alpha_n(\mathcal{A}) \triangleq \Pr[\mathcal{A}(z^\star, o) = 1 \mid b = 0]$. *The Type II error*, also called the False Negative Rate, is $\beta_n(\mathcal{A}) \triangleq \Pr[\mathcal{A}(z^\star o) = 0 \mid b = 1]$. *The power of the test* is $1 - \beta_n(\mathcal{A})$. An adversary can use a threshold over a score function to conduct the MI games, i.e. for $\mathcal{A}_{s,\tau}(z^\star, o) \triangleq \mathbb{1}\{s(o; z^\star) > \tau\}$ where s is a scoring function and τ is a threshold. We want to design scores that maximise the power under a fixed significance level α, i.e. $\mathrm{Pow}_n(s, \alpha) \triangleq \max_{\tau \in T_\alpha} 1 - \beta_n(\mathcal{A}_{s,\tau})$, where $T_\alpha \triangleq \{\tau \in \mathbb{R} : \alpha_n(\mathcal{A}_{s,\tau}) \leq \alpha\}$. $\mathrm{Pow}_n(s, \alpha)$ is also called a trade-off function.

B.3 Connection Between the Fixed-target and the Average-target MI Games

An adversary $\mathcal{A}$ can be regarded as an infinite collection of target-dependent adversaries $(\mathcal{A}_{z^\star})_{z^\star \in \mathcal{Z}}$, where $\mathcal{A}(z^\star, o) = \mathcal{A}_{z^\star}(o)$.

The advantage (and accuracy) of $\mathcal{A}$ is the expected advantage (and accuracy) of $\mathcal{A}_{z^\star}$, when $z^\star \sim \mathcal{D}$, i.e. $\mathrm{Adv}_n(\mathcal{A}) = \mathbb{E}_{z^\star \sim \mathcal{D}} \mathrm{Adv}_n(\mathcal{A}_{z^\star})$. Thus, studying the performance metrics of an adversary under average-target MI game hides the dependence on the target, by averaging out the performance on different target points. As a

consequence, using the average-target MI games for auditing privacy can hurt the performance. A gain could directly be observed by running the same attack on an "easy to attack" fixed-target MI game.

The optimal attack for the average-target MI game is the same optimal LR attack for the target-dependent MI game:

$$\ell_n(o; z^\star) \triangleq \log\left(\frac{p_n^{\text{in}}(z^\star, o)}{p_n^{\text{out}}(z^\star, o)}\right) = \log\left(\frac{p_n^{\text{in}}(o \mid z^\star)p_n^{\text{in}}(z^\star)}{p_n^{\text{out}}(o \mid z^\star)p_n^{\text{out}}(z^\star)}\right) = \log\left(\frac{p_n^{\text{in}}(o \mid z^\star)}{p_n^{\text{out}}(o \mid z^\star)}\right)$$

since $p_n^{\text{in}}(z^\star) = p_n^{\text{out}}(z^\star) = \mathcal{D}(z^\star)$.

Thus, the same LR attack optimally solves both fixed-target and average-target MI games. The only difference is in analysing the performance of the attack, i.e. whether we average out the effect of $z^\star \sim \mathcal{D}$ in average-target MI games or we keep the dependence on the target $z^\star$, by fixing $z^\star$ in the fixed-target MI games.

Algorithm 5 The Target-crafter

Input: Mechanism $\mathcal{M}$, Data distribution $\mathcal{D}$, #samples n
Output: $(z^\star, o, b)$, where $z^\star \in \mathcal{Z}$, $o \in \mathcal{O}$ and $b \in \{0, 1\}$
Build a dataset $D \sim \bigotimes_{i=1}^{n} \mathcal{D}$
Sample $b \sim \text{Bernoulli}\left(\frac{1}{2}\right)$
if $b = 0$ **then**
  Sample $z^\star \sim \mathcal{D}$ ind. of D
else
  Sample $j \sim \mathcal{U}[n]$
  Assign $z^\star$ to be the i-th element of D
end if
Sample $o \sim \mathcal{M}(D)$
Return $(z^\star, o, b)$

Algorithm 6 Average-target MI Game

Input: Mechanism $\mathcal{M}$, Data distribution $\mathcal{D}$, #samples n, Adversary $\mathcal{A}$, Rounds T
Output: A list $L \in \{0, 1\}^T$, where $L_t = 1$ if the adversary succeeds at step t.
Initialise a empty list L of length T
for t = 1, . . . , T **do**
  Sample $(z_t^\star, o_t, b_t) \sim$ Target-crafter($\mathcal{M}$, $\mathcal{D}$, n)
  Sample $\hat{b}_t \sim \mathcal{A}(z_t^\star, o_t)$
  Set $L_t \leftarrow \mathbb{1}\left\{b_t = \hat{b}_t\right\}$
end for
Return L

B.4 The Likelihood Ratio Test for Bernoulli Empirical Mean Average-target MI Game

We revisit results from (Sankararaman et al., 2009). In (Sankararaman et al., 2009), the MI game is instantiated with the empirical mean mechanism denoted by $\mathcal{M}_n^{\text{emp}}$. The mechanism $\mathcal{M}_n^{\text{emp}}$ takes as input a dataset of size n of d-dimensional points, i.e. $D = \{Z_1, \ldots, Z_n\} \in (\mathbb{R}^d)^n$, and outputs the exact empirical mean $\hat{\mu}_n \triangleq \frac{1}{n}\sum_{i=1}^{n} Z_i \in \mathbb{R}^d$.

Assumptions on the data generating distribution and asymptotic regime. Sankararaman et al. (2009) supposes that the data-generating distribution $\mathcal{D}$ is colon-wise independent Bernoulli distributions, i.e. $\mathcal{D} \triangleq \bigotimes_{j=1}^{d} \text{Bernoulli}(\mu_j)$, with $\mu_j \in [a, 1-a]$ for some $a \in (0, 1/2)$. We denote by $\rightsquigarrow$ convergence in distribution, i.e. A sequence of random variables $X_n \rightsquigarrow X$ if and only if $\Pr(X_n \leq x) \to \Pr(X \leq x)$ for all x. Let Φ represent the Cumulative Distribution Function (CDF) of the standard normal distribution, i.e. $\Phi(\alpha) \triangleq \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\alpha} e^{-t^2/2}\,\mathrm{d}t$ for $\alpha \in \mathbb{R}$. Sankararaman et al. (2009) studies the *asymptotic behaviour of the LR test*, when both the sample size n and the dimension d tend to infinity such that $d/n = \tau > 0$.

The analysis of (Sankararaman et al., 2009) starts by showing that the exact formula of the LR score, at output $o = \hat{\mu}_n$ and target $z^\star$ is

$$\ell_n(\hat{\mu}_n, z^\star) = \sum_{j=1}^{d} z_j^\star \log\left(\frac{\hat{\mu}_{n,j}}{\mu_j}\right) + (1 - z_j^\star)\log\left(\frac{1-\hat{\mu}_{n,j}}{1-\mu_j}\right). \tag{11}$$

As d and n tend to infinity such that $d/n = \tau$, Sankararaman et al. (2009) shows that the LR score converges in distribution to

$$\ell_n(\hat{\mu}_n, z^\star) \rightsquigarrow^{H_0} \mathcal{N}\left(-\frac{1}{2}\tau, \tau\right) \tag{12}$$

under H_0 and converges to

$$\ell_n(\hat{\mu}_n, z^\star) \rightsquigarrow^{H_1} \mathcal{N}\left(\frac{1}{2}\tau, \tau\right) \tag{13}$$

under H_1.

The asymptotic distribution of the LR score helps to provide the asymptotic trade-off of the optimal LR attacker. Specifically, the main result (Section T2.1 in (Sankararaman et al., 2009)) is that

$$\Phi^{-1}(1-\alpha) + \Phi^{-1}(1-\beta) \approx \sqrt{d/n} \tag{14}$$

where α is the optimal Type I error, β is the optimal Type II error and Φ represents the Cumulative Distribution Function (CDF) of the standard normal distribution, i.e. $\Phi(\alpha) \triangleq \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\alpha} e^{-t^2/2}\,\mathrm{d}t$ for $\alpha \in \mathbb{R}$. This trade-off between α and β shows that the MI game gets easier, as d/n gets bigger.

Connection to our results. Our results in Section 3 are a target-dependent version of the results of (Sankararaman et al., 2009). Also, our analysis generalises the analysis of (Sankararaman et al., 2009) beyond Bernoulli distributions to any distribution with finite 4-th moment. Specifically, Lemma C.6 shows that, in the target-dependent MI game,

- Under H_0:

$$\ell_n(\hat{\mu}_n; z^\star, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(-\frac{1}{2}m^\star, m^\star\right)$$

- Under H_1:

$$\ell_n(\hat{\mu}_n; z^\star, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(\frac{1}{2}m^\star, m^\star\right)$$

where the convergence is a convergence in distribution, such that $d, n \to \infty$, while $d/n = \tau$.

This is a target-dependent version of Equations (12) and (13). Also, Equation (4) of Theorem 3.1 is a target-dependent version of Equation (14). Thus, our results retrieve the average-case results by observing that, since $\mathbb{E}_{z^\star \sim \mathcal{D}}\left[\|z^\star - \mu\|^2_{C_\sigma^{-1}}\right] = d$, we have $\mathbb{E}_{z^\star \sim \mathcal{D}}\left[m^\star\right] = \lim_{n,d} \frac{d}{n} = \tau$.

B.5 The Scalar Product Attack for Average-target MI game

Dwork et al. (2015) proposes a scalar product attack for tracing the empirical mean that thresholds over the score

$$s^{\mathrm{scal}}(\hat{\mu}_n, z^\star; z^{\mathrm{ref}}) \triangleq (z^\star - z^{\mathrm{ref}})^T \hat{\mu}_n,$$

where z^{ref} is one reference point. The intuition behind this attack is to compare the target-output correlation $(z^\star)^T\hat{\mu}_n$ with a reference-output correlation $(z^{\mathrm{ref}})^T\hat{\mu}_n$. The analysis of (Dwork et al., 2015) shows that with only one reference point $z^{\mathrm{ref}} \sim \mathcal{D}$, and even for noisy estimates of the mean, the attack is able to trace the data of some individuals in the regime $d \sim n^2$.

Informally, the analysis of (Dwork et al., 2015) considers a scalar product attack, taking as input any 1/2-*accurate estimate* $\hat{\mu}$ of a dataset $D \in (\{-1,1\}^d)^n$ of dimension $d = O\left(n^2 \log(1/\delta)\right)$, a target point $z^\star \in \{-1,1\}^d$ and *only a single reference sample* $z^{\mathrm{ref}} \in \{-1,1\}^d$. The data-generating distribution is assumed to be chosen from a strong class of distributions $\mathcal{P}$. Then,

- If $z^\star$ is IN the dataset D, then

$$\Pr\left\{s^{\mathrm{scal}}(\hat{\mu}_n, z^\star; z^{\mathrm{ref}}) > \tau\right\} \geq \Omega(1/n).$$

- If $z^\star$ is Out of the dataset D, then

$$\Pr\left\{s^{\mathrm{scal}}(\hat{\mu}_n, z^\star; z^{\mathrm{ref}}) < \tau\right\} \geq 1 - \delta.$$

for a carefully chosen threshold $\tau = O(\sqrt{d\log(1/\delta)})$.

The condition of 1/2-accuracy is a weak condition, compared to the exact empirical mean attack (Sankararaman et al., 2009). The price of the weak notion of accuracy is that the attack is only guaranteed for $d \gtrapprox n^2$, whereas the exact attack of (Sankararaman et al., 2009) is able to trace for $d \approx n$.

B.6 Proof of Theorem 2.1

***Theorem** 2.1* (Characterising Optimal Adversaries). (a) $\mathcal{A}_{\text{Bayes},z^\star}$ is the adversary that maximises the advantage (and accuracy), i.e. for any adversary $\mathcal{A}_{z^\star}$, we have that $\text{Adv}_n(\mathcal{A}_{\text{Bayes},z^\star}) - \text{Adv}_n(\mathcal{A}_{z^\star}) \geq 0$.

(b) The advantage of the optimal Bayes adversary is

$$\text{Adv}_n(\mathcal{A}_{\text{Bayes},z^\star}) = \text{TV}\left(p_n^{\text{out}}(. \mid z^\star) \;\middle\|\; p_n^{\text{in}}(. \mid z^\star)\right),$$

where TV is the total variation distance.

(c) For every $\alpha \in [0,1]$, the log-likelihood score is the score that maximises the power under significance α, i.e. for any α and any score function s, $\text{Pow}_n(\ell, \alpha, z^\star) \geq \text{Pow}_n(s, \alpha, z^\star)$.

Proof. To prove (a), we observe that the log-likelihood adversary with threshold $\tau = 0$ is exactly the Bayes optimal classifier. Specifically, since $Pr(b=0) = Pr(b=1)$, we can rewrite the log-likelihood as the $\ell_n(o; z^\star) = \log\left(\frac{\Pr(b=1|o,z^\star)}{\Pr(b=0|o,z^\star)}\right)$. Thus thresholding with 0 gives exactly the Bayes optimal classifier, which has the highest accuracy among all classifiers.

For (b), we observe that

$$\begin{aligned}\text{Adv}_n(\mathcal{A}_{\text{Bayes},z^\star}) &= \Pr(\ell_n(o; z^\star) \leq 0 \mid b=0) - \Pr(\ell_n(o; z^\star) \leq 0 \mid b=1) \\ &= p_n^{\text{out}}(O \mid z^\star) - p_n^{\text{in}}(O \mid z^\star)\end{aligned}$$

where $O \triangleq \{o \in \mathcal{O} : p_n^{\text{out}}(o \mid z^\star) \geq p_n^{\text{in}}(o \mid z^\star)\}$.

The last equation is exactly the definition of the TV $\left(p_n^{\text{out}}(. \mid z^\star) \;\middle\|\; p_n^{\text{in}}(. \mid z^\star)\right)$.

Finally, (c) is a direct consequence of the Neyman-Pearson lemma. □

C MISSING PROOFS

First, we present some classic background results from asymptotic statics, the Edgeworth asymptotic expansion (Theorem C.5 and Linderberg-Feller theorem C.4. Then, we provide the missing proofs of Theorem C.6, Corollary 3.1, Theorem 4.2 and Theorem 4.3.

C.1 Background on Asymptotic Statics

A sequence of random variables X_n is said to converge in distribution to a random variable X, i.e. $X_n \rightsquigarrow_n X$ if $\Pr(X_n \leq x) \to \Pr(X \leq x)$, for every x at which the limit distribution $x \to \Pr(X \leq x)$ is continuous.

A sequence of random variables X_n is said to converge in probability to X if for every $\epsilon > 0$, $\Pr(\|X_n - X\| > \epsilon) \to 0$, denoted by $X_n \to^{\mathrm{P}} X$.

We recall that continuous mappings preserve both convergences.

A sequence of random variable (X_n) is called uniformly tight if: for every ϵ, $\exists M > 0$, such that $\sup_n \Pr(\|X_n\| > M) < \epsilon$.

Theorem C.1 (Prohorov's theorem)**.** *Let X_n be a random vector in $\mathbb{R}^d$.*

1. If $X_n \rightsquigarrow X$, for some X, then the sequence (X_n) is uniformly tight;

2. If (X_n) is uniformly tight, then there exits a subsequence with $X_{n_j} \rightsquigarrow X$ as $j \to \infty$ for some X.

We also recall the stochastic o_p and O_p notation for random variables.

Definition C.2 (Stochastic o_p and O_p)**.** We say that $X_n = o_p(R_n)$ if $X_n = Y_n R_n$ and $Y_n \to^{\mathrm{P}} 0$

We say that $X_n = O_p(R_n)$ if $X_n = Y_n R_n$ and $Y_n = O_p(1)$ where $O_p(1)$ denotes a sequence that is uniformly tight (also called bounded in probability).

The following lemma is used to get Taylor expansions of random variables.

Lemma C.3 (Lemma 2.12 in Van der Vaart (2000))**.** *Let R be a function on $\mathbb{R}^k$, such that $R(0) = 0$. Let $X_n = o_p(1)$.*

Then, for every $p > 0$,

(a) if $R(h) = o(\|h\|^p)$ as $h \to 0$, then $R(X_n) = o_p(\|X_n\|^p)$;

(b) if $R(h) = O(\|h\|^p)$ as $h \to 0$, then $R(X_n) = O_p(\|X_n\|^p)$.

The Lindeberg-Feller theorem is the simplest extension of the classical central limit theorem (CLT) and is applicable to independent (and not necessarily identically distributed) random variables with finite variances.

Theorem C.4 (Lindeberg-Feller CLT)**.** *Let $Y_{n,1}, \dots Y_{n,d_n}$ be independent random vectors with finite variances such that*

1. for every $\epsilon > 0$, $\sum_{j=1}^{d_n} \mathbb{E}\left[\|Y_{n,i}\|^2 \mathbb{1}\left\{\|Y_{n,i}\| > \epsilon\right\}\right] \to 0$,

2. $\sum_{j=1}^{d_n} \mathbb{E}\left[Y_{n,i}\right] \to \mu$,

3. $\sum_{j=1}^{d_n} \mathrm{Cov}\left[Y_{n,i}\right] \to \Sigma$.

Then $\sum_{j=1}^{d_n} Y_{n,j} \rightsquigarrow \mathcal{N}(\mu, \Sigma)$

Finally, the last result from asymptotic statistics is the Edgeworth asymptotic expansion in the CLT.

Theorem C.5 (Edgeworth expansion, Theorem 15 of Chapter 7 in Petrov (2012))**.** *Let $Z_1, \dots, Z_n$ sampled i.i.d from $\mathcal{D}$, where $\mathcal{D}$ has a finite absolute moment of k-th order, i.e. $\mathbb{E}[|X_1|^k] < \infty$. Let d_n be the density of the centred normalised mean $\frac{1}{\sigma\sqrt{n}} \sum_{i=1}^n X_i$, then*

$$d_n(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2} + \sum_{\nu=1}^{k-2} \frac{q_\nu(x)}{n^{\nu/2}} + o\left(\frac{1}{n^{(k-2)/2}}\right)$$

uniformly in x, where $q_v(x)$ are related to the Hermite Polynomials.

C.2 Proof of the Asymptotic Distribution of the LR Scores, Lemma C.6

Lemma C.6 (Asymptotic distribution of the LR score)**.** *Using an Edgeworth asymptotic expansion of the LR score and a Lindeberg-Feller central limit theorem, we show that*

- Under H_0:

$$\ell_n(\hat{\mu}_n; z^\star, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(-\frac{1}{2}m^\star, m^\star\right)$$

- Under H_1:

$$\ell_n(\hat{\mu}_n; z^\star, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(\frac{1}{2}m^\star, m^\star\right)$$

where the convergence is a convergence in distribution, such that $d, n \to \infty$, while $d/n = \tau$. We call $m^\star$ the leakage score of target datum $z^\star$.

Proof. We have that $\hat{\mu}_n = \frac{1}{n}\sum_{i=1}^n Z_i$, where $Z_i = (Z_{i,j})_{j=1}^{d_n} \in \mathbb{R}^{d_n}$ and $Z_i \sim^{\text{i.i.d}} \mathcal{D} = \bigotimes_{j=1}^{d_n} \mathcal{D}_j$.

Each distribution $\mathcal{D}_j$ has mean μ_j and variance σ_j^2.

We denote $\hat{\mu}_n = (\hat{\mu}_{n,j})_{j=1}^{d_n}$, where $\hat{\mu}_{n,j} = \frac{1}{n}\sum_{i=1}^n Z_{i,j}$.

Step 1: Rewriting the LR score

Let $j \in [1, d_n]$.

Under H_0, we can re-write

$$\hat{\mu}_{n,j} = \mu_j + \frac{\sigma_j}{\sqrt{n}}\hat{Z}_{n,j},$$

where

$$\begin{aligned}\hat{Z}_{n,j} &\triangleq \sqrt{n}\left(\frac{\hat{\mu}_{n,j} - \mu_j}{\sigma}\right) \\ &= \frac{1}{\sqrt{n}}\sum_{i=1}^n \frac{Z_{i,j} - \mu_j}{\sigma_j}.\end{aligned}$$

Since $(Z_{i,j})_{i=1}^n$ are i.i.d from $\mathcal{D}_j$, using the CLT, $\hat{Z}_{n,j} \rightsquigarrow_{n\to\infty} \mathcal{N}(0,1)$.

Let $d_{n,j}$ be the density function of $\hat{Z}_{n,j}$.

The density $p_{n,j}^{\text{out}}$ of $\hat{\mu}_{n,j}$ under H_0 can be written as

$$p_{n,j}^{\text{out}}(x; z_j^\star, \mu_j, \sigma_j) = \frac{\sqrt{n}}{\sigma_j} d_{n,j}\left[\frac{\sqrt{n}}{\sigma_j}(x - \mu_j)\right]$$

Under H_1, we can re-write

$$\begin{aligned}\hat{\mu}_{n,j} &= \frac{1}{n}z_j^\star + \frac{n-1}{n}\left(\mu_j + \frac{\sigma_j}{\sqrt{n-1}}\hat{Z}_{n-1,j}\right) \\ &= \mu_j + \frac{1}{n}\left(z_j^\star - \mu_j\right) + \frac{\sigma_j\sqrt{n-1}}{n}\hat{Z}_{n-1,j}\end{aligned}$$

The density $p_{n,j}^{\text{in}}$ of $\hat{\mu}_{n,j}$ under H_1 can be written as

$$p_{n,j}^{\text{in}}(x; z_j^\star, \mu_j, \sigma_j) = \frac{n}{\sigma_j\sqrt{n-1}} d_{n-1,j}\left[\frac{n}{\sigma_j\sqrt{n-1}}\left(x - \mu_j - \frac{1}{n}\left(z_j^\star - \mu_j\right)\right)\right]$$

The LR score is

$$\begin{aligned}\ell_n(\hat{\mu}_n; z^\star, \mu, C_\sigma) &= \sum_{j=1}^{d_n} \log\left(\frac{p^{\text{in}}_{n,j}(\hat{\mu}_{n,j}; z^\star_j, \mu_j, \sigma_j)}{p^{\text{out}}_{n,j}(\hat{\mu}_{n,j}; z^\star_j, \mu_j, \sigma_j)}\right)\\ &= \sum_{j=1}^{d_n} -\frac{1}{2}\log\left(1-\frac{1}{n}\right) + \log\left(\frac{d_{n-1,j}\left(\delta^{\text{in}}_{n,j}\right)}{d_{n,j}\left(\delta^{\text{out}}_{n,j}\right)}\right)\end{aligned}$$

where

$$\begin{aligned}\delta^{\text{out}}_{n,j} &\triangleq \frac{\sqrt{n}}{\sigma_j}\left(\hat{\mu}_{n,j} - \mu_j\right)\\ \delta^{\text{in}}_{n,j} &\triangleq \frac{n}{\sqrt{n-1}\sigma_j}\left(\hat{\mu}_{n,j} - \mu_j + \frac{1}{n}\left(\mu_j - z^\star_j\right)\right)\end{aligned}$$

Step 2: Asymptotic expansion of the LR score

Using Lemma C.8, we have

$$\log\left(\frac{d_{n-1,j}\left(\delta^{\text{in}}_{n,j}\right)}{d_{n,j}\left(\delta^{\text{out}}_{n,j}\right)}\right) = \frac{1}{2}\left(\left(\delta^{\text{out}}_{n,j}\right)^2 - \left(\delta^{\text{in}}_{n,j}\right)^2\right) + \frac{\lambda_3\left(\mu_j - z^\star_j\right)}{n\sigma_j}R_{n,j} + o_p\left(\frac{1}{n}\right)$$

Let $Y_{n,j} \triangleq \frac{1}{2}\left(\left(\delta^{\text{out}}_{n,j}\right)^2 - \left(\delta^{\text{in}}_{n,j}\right)^2\right) + \frac{\lambda_3\left(\mu_j - z^\star_j\right)}{n\sigma_j}R_{n,j}$.

We remark that we need an expansion up to $o_p\left(\frac{1}{n}\right)$, since $d_n/n = \tau + o(1)$.

Thus

$$\begin{aligned}\ell_n(\hat{\mu}_n; z^\star, \mu, C_\sigma) &= \sum_{j=1}^{d_n} -\frac{1}{2}\log\left(1-\frac{1}{n}\right) + \log\left(\frac{d_{n-1,j}\left(\delta^{\text{in}}_{n,j}\right)}{d_{n,j}\left(\delta^{\text{out}}_{n,j}\right)}\right)\\ &= \sum_{j=1}^{d_n}\left(\frac{1}{2n} + Y_{n,j} + o_p\left(\frac{1}{n}\right)\right)\\ &= \frac{\tau}{2} + o_p(1) + \sum_{j=1}^{d_n} Y_{n,j} \qquad (15)\end{aligned}$$

because $\frac{d_n}{n} = \tau + o(1)$.

Step3: Concluding using the Lindeberg-Feller CLT

Under H_0:

Using Lemma C.9, $\mathbb{E}_0[Y_n, j] = -\frac{1}{2n} - \frac{(z^\star_j - \mu_j)^2}{2n\sigma_j^2} + o\left(\frac{1}{n}\right)$ and $V_0[Y_n, j] = \frac{(z^\star_j - \mu_j)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right)$.

Since $\sum_{j=1}^{d_n}\frac{(z^\star_j - \mu_j)^2}{n\sigma_j^2} = \frac{\|z^\star - \mu\|^2_{C_\sigma^{-1}}}{n}$, we get:

- $\sum_{j=1}^{d_n}\mathbb{E}_0[Y_n, j] \to -\frac{\tau}{2} - \frac{m^\star}{2}$
- $\sum_{j=1}^{d_n} V_0[Y_n, j] \to m^\star$

Using Lemma C.10, we have that $Y_{n,j}$ verify the Lindeberg-Feller condition, i.e.

$$\sum_{j=1}^{d_n}\mathbb{E}_0\left[Y^2_{n,j}\mathbb{1}\left\{|Y_{n,j}| > \epsilon\right\}\right] \to 0$$

for every $\epsilon > 0$.

We conclude using the Lindeberg-Feller CLT (Theorem C.4) that $\sum_{j=1}^{d_n} Y_{n,j} \rightsquigarrow \mathcal{N}\left(-\frac{\tau}{2} - \frac{m^\star}{2}, m^\star\right)$, and thus

$$\ell_n(\hat{\mu}_n; z^\star, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(-\frac{m^\star}{2}, m^\star\right)$$

Similarly, *Under H_1:*

Using Lemma C.9, $\mathbb{E}_1[Y_n, j] = -\frac{1}{2n} + \frac{(z_j^\star - \mu_j)^2}{2n\sigma_j^2} + o\left(\frac{1}{n}\right)$ and $V_1[Y_n, j] = \frac{(z_j^\star - \mu_j)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right)$.

We get:

- $\sum_{j=1}^{d_n} \mathbb{E}_1[Y_{n,j}] \to -\frac{\tau}{2} + \frac{m^\star}{2}$
- $\sum_{j=1}^{d_n} V_1[Y_{n,j}] \to m^\star$

Using Lemma C.10, we have that $Y_{n,j}$ verify the Lindeberg-Feller condition, i.e.

$$\sum_{j=1}^{d_n} \mathbb{E}_1\left[Y_{n,j}^2 \mathbb{1}\left\{|Y_{n,j}| > \epsilon\right\}\right] \to 0$$

for every $\epsilon > 0$.

We conclude using the Lindeberg-Feller CLT (Theorem C.4) that $\sum_{j=1}^{d_n} Y_{n,j} \rightsquigarrow \mathcal{N}\left(-\frac{\tau}{2} + \frac{m^\star}{2}, m^\star\right)$, and thus

$$\ell_n(\hat{\mu}_n; z^\star, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(\frac{m^\star}{2}, m^\star\right)$$

□

Remark C.7. Expanding $\frac{1}{2}\left(\left(\delta_{n,j}^{\text{out}}\right)^2 - \left(\delta_{n,j}^{\text{in}}\right)^2\right)$, taking the sum from $j = 1$ until d_n, we get that

$$\ell_n(\hat{\mu}_n; z^\star, \mu, C_\sigma) \sim (z^\star - \mu)^T C_\sigma^{-1}(\hat{\mu}_n - \mu) - \frac{1}{2n}\|z^\star - \mu\|_{C_\sigma^{-1}}^2$$

Let $X_n \triangleq (z^\star - \mu)^T C_\sigma^{-1}(\hat{\mu}_n - \mu) - \frac{1}{2n}\|z^\star - \mu\|_{C_\sigma^{-1}}^2$.

This asymptotic representation of the LR test is useful to get directly the means and variances of the limit distribution of the LR test. Specifically, since $\mathbb{E}_0(\hat{\mu}_n) = \mu$, $\mathbb{E}_1(\hat{\mu}_n) = \frac{n-1}{n}\mu + \frac{1}{n}z^\star$ and $\mathbb{V}_0(\hat{\mu}_n) = \mathbb{V}_1(\hat{\mu}_n) = C_\sigma$, we get that

$$\begin{aligned}
\mathbb{E}_0\left[X_n\right] &= -\frac{1}{2n}\|z^\star - \mu\|_{C_\sigma^{-1}}^2 \\
\mathbb{E}_1\left[X_n\right] &= \frac{1}{2n}\|z^\star - \mu\|_{C_\sigma^{-1}}^2 \\
\mathbb{V}_0\left[X_n\right] &= \mathbb{V}_1\left[X_n\right] = \frac{1}{n}\|z^\star - \mu\|_{C_\sigma^{-1}}^2
\end{aligned}$$

Taking the limit as $n \to \infty$ retrieves the results of Theorem C.6.

C.3 The Three Technical Lemmas Used in the Proof of Lemma C.6

Lemma C.8. *Asymptotic expansion of the LR score*

We show that

$$\log\left(\frac{d_{n-1,j}\left(\delta_{n,j}^{\text{in}}\right)}{d_{n,j}\left(\delta_{n,j}^{\text{out}}\right)}\right)=\frac{1}{2}\left(\left(\delta_{n,j}^{\text{out}}\right)^2-\left(\delta_{n,j}^{\text{in}}\right)^2\right)+\frac{\lambda_3\left(\mu_j-z_j^\star\right)}{n\sigma_j}R_{n,j}+o_p\left(\frac{1}{n}\right)$$

where $\delta_{n,j}^{\text{out}}\triangleq\frac{\sqrt{n}}{\sigma_j}\left(\hat{\mu}_{n,j}-\mu_j\right)$, $\delta_{n,j}^{\text{in}}\triangleq\frac{n}{\sqrt{n-1}\sigma_j}\left(\hat{\mu}_{n,j}-\mu_j+\frac{1}{n}\left(\mu_j-z_j^\star\right)\right)$,

and $R_{n,j}\triangleq\left(\delta_{n,j}^{\text{out}}\right)^2+\delta_{n,j}^{\text{out}}\delta_{n,j}^{\text{in}}+\left(\delta_{n,j}^{\text{in}}\right)^2-3$.

Proof. The proof starts by using the Edgeworth expansion of $d_{n,j}$ up to the order $k=4$. Then, the final LR expansion can be found using Taylor expansions of the logarithm, exponential and polynomial function to the 2nd order. Here, we present the exact derivations for completeness.

Step1: Asymptotic expansion of $d_{n,j}$

Using Theorem C.5, for $k=4$, we get that

$$d_{n,j}(x)=\frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}\left(1+\frac{h_1(x)}{\sqrt{n}}+\frac{h_2(x)}{n}\right)+o\left(\frac{1}{n}\right)\tag{16}$$

uniformly in x, where

$$h_1(x)\triangleq\lambda_3\left(x^3-3x\right)$$
$$h_2(x)\triangleq\frac{\lambda_3^2}{72}\left(x^6-15x^4+45x^2-15\right)+\frac{\lambda_4}{24}\left(x^4-6x^2+3\right)$$

and $\lambda_k\triangleq\frac{\gamma_{j,k}}{\sigma_j^k}$ where $\gamma_{j,k}$ is the k-order cumulant of distribution $\mathcal{D}_j$.

We do an expansion up to the 4-th order, since we need an expansion up to $o_p\left(\frac{1}{n}\right)$.

Step2: Asymptotic expansion of $d_{n,j}\left(\delta_{n,j}^{\text{out}}\right)$ and $d_{n-1,j}\left(\delta_{n,j}^{\text{in}}\right)$

Since the convergence in uniform in Equation 16, we get that

$$\begin{aligned}d_{n,j}\left(\delta_{n,j}^{\text{out}}\right)&=\frac{1}{\sqrt{2\pi}}e^{-\frac{(\delta_{n,j}^{\text{out}})^2}{2}}\left(1+\frac{h_1(\delta_{n,j}^{\text{out}})}{\sqrt{n}}+\frac{h_2(\delta_{n,j}^{\text{out}})}{n}+\right)+o_p\left(\frac{1}{n}\right)\\&=\frac{1}{\sqrt{2\pi}}e^{-\frac{(\delta_{n,j}^{\text{out}})^2}{2}}\left(1+\frac{h_1(\delta_{n,j}^{\text{out}})}{\sqrt{n}}+\frac{h_2(\delta_{n,j}^{\text{out}})}{n}+e^{\frac{(\delta_{n,j}^{\text{out}})^2}{2}}o_p\left(\frac{1}{n}\right)\right)\end{aligned}\tag{17}$$

and

$$\begin{aligned}d_{n-1,j}\left(\delta_{n,j}^{\text{in}}\right)&=\frac{1}{\sqrt{2\pi}}e^{-\frac{(\delta_{n,j}^{\text{in}})^2}{2}}\left(1+\frac{h_1(\delta_{n,j}^{\text{in}})}{\sqrt{n-1}}+\frac{h_2(\delta_{n,j}^{\text{in}})}{n-1}\right)+o_p\left(\frac{1}{n}\right)\\&=\frac{1}{\sqrt{2\pi}}e^{-\frac{(\delta_{n,j}^{\text{in}})^2}{2}}\left(1+\frac{h_1(\delta_{n,j}^{\text{in}})}{\sqrt{n-1}}+\frac{h_2(\delta_{n,j}^{\text{in}})}{n-1}+e^{\frac{(\delta_{n,j}^{\text{in}})^2}{2}}o_p\left(\frac{1}{n}\right)\right)\end{aligned}\tag{18}$$

Under both H_0 and H_1, $\delta_{n,j}^{\text{in}}\rightsquigarrow_{n\to\infty}Z\triangleq\mathcal{N}(0,1)$ and $\delta_{n,j}^{\text{out}}\rightsquigarrow_{n\to\infty}Z'\triangleq\mathcal{N}(0,1)$.

Since $x\to e^{x^2/2}$ is a continuous function, then $e^{\frac{(\delta_{n,j}^{\text{in}})^2}{2}}\rightsquigarrow_{n\to\infty}e^{Z^2/2}$ and $e^{\frac{(\delta_{n,j}^{\text{out}})^2}{2}}\rightsquigarrow_{n\to\infty}e^{(Z')^2/2}$.

Using Prohorov's theorem (Thm C.1), we get that $e^{\frac{(\delta_{n,j}^{\text{in}})^2}{2}}=O_p(1)$ and $e^{\frac{(\delta_{n,j}^{\text{out}})^2}{2}}=O_p(1)$.

This means that

$$e^{\frac{(\delta^{\text{in}}_{n,j})^2}{2}} o_p\left(\frac{1}{n}\right) = O_p(1) o_p\left(\frac{1}{n}\right) = o_p\left(\frac{1}{n}\right) \tag{19}$$

and

$$e^{\frac{(\delta^{\text{out}}_{n,j})^2}{2}} o_p\left(\frac{1}{n}\right) = O_p\left(\frac{1}{n}\right) o_p\left(\frac{1}{n}\right) = o_p\left(\frac{1}{n}\right). \tag{20}$$

On the other hand, both h_1 and h_2 are continuous functions (polynomials functions), this similarly gives that $h_1(\delta^{\text{in}}_{n,j}) = O_p(1)$ and $h_2(\delta^{\text{in}}_{n,j}) = O_p(1)$.

Combined with the fact that $\frac{1}{\sqrt{n-1}} = \frac{1}{\sqrt{n}} + o\left(\frac{1}{n}\right)$ and $\frac{1}{n-1} = \frac{1}{n} + o\left(\frac{1}{n}\right)$, we get that

$$\frac{h_1(\delta^{\text{in}}_{n,j})}{\sqrt{n-1}} = \frac{h_1(\delta^{\text{in}}_{n,j})}{\sqrt{n}} + O_p(1) o\left(\frac{1}{n}\right) = \frac{h_1(\delta^{\text{in}}_{n,j})}{\sqrt{n}} + o_p\left(\frac{1}{n}\right) \tag{21}$$

and

$$\frac{h_2(\delta^{\text{in}}_{n,j})}{n-1} = \frac{h_1(\delta^{\text{in}}_{n,j})}{n} + O_p(1) o\left(\frac{1}{n}\right) = \frac{h_1(\delta^{\text{in}}_{n,j})}{n} + o_p\left(\frac{1}{n}\right) \tag{22}$$

Plugging Eq. 20 in Eq. 17 gives

$$d_{n,j}\left(\delta^{\text{out}}_{n,j}\right) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(\delta^{\text{out}}_{n,j})^2}{2}} \left(1 + \frac{h_1(\delta^{\text{out}}_{n,j})}{\sqrt{n}} + \frac{h_2(\delta^{\text{out}}_{n,j})}{n} + o_p\left(\frac{1}{n}\right)\right) \tag{23}$$

and Plugging Eq. 19, Eq.21 and Eq. 22 in Eq. 18 gives

$$d_{n-1,j}\left(\delta^{\text{in}}_{n,j}\right) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(\delta^{\text{in}}_{n,j})^2}{2}} \left(1 + \frac{h_1(\delta^{\text{in}}_{n,j})}{\sqrt{n}} + \frac{h_2(\delta^{\text{in}}_{n,j})}{n} + o_p\left(\frac{1}{n}\right)\right) \tag{24}$$

Step3: Asymptotic expansion of the logarithm

Applying the logarithm to Eq.23, we get that

$$\begin{aligned}
\log\left(d_{n,j}\left(\delta^{\text{out}}_{n,j}\right)\right) &= -\frac{1}{2}\log(2\pi) - \frac{(\delta^{\text{out}}_{n,j})^2}{2} + \log\left(1 + \frac{h_1(\delta^{\text{out}}_{n,j})}{\sqrt{n}} + \frac{h_2(\delta^{\text{out}}_{n,j})}{n} + o_p\left(\frac{1}{n}\right)\right) \\
&= -\frac{1}{2}\log(2\pi) - \frac{(\delta^{\text{out}}_{n,j})^2}{2} + \log\left(1 + H_n, j\right)
\end{aligned} \tag{25}$$

where $H_{n,j} \triangleq \frac{h_1(\delta^{\text{out}}_{n,j})}{\sqrt{n}} + \frac{h_2(\delta^{\text{out}}_{n,j})}{n} + o_p\left(\frac{1}{n}\right)$.

Let $R(h) = \log(1+h) - (h - \frac{h^2}{2})$, for $h \in \mathbb{R}^+$.

We have that $R(0) = 0$ and $R(h) = o\left(h^2\right)$ as $h \to 0$.

On the other hand, $H_{n,j} \to^P_{n\to\infty} 0$. Specifically $H_{n,j} = O_p\left(\frac{1}{\sqrt{n}}\right)$.

Using Lemma C.3, we get that $R(H_n, j) = o_p\left((H_n, j)^2\right) = o_p\left(\frac{1}{n}\right)$.

All in all, we have shown

$$\begin{aligned}
\log(1 + H_{n,j}) &= H_{n,j} - \frac{H^2_{n,j}}{2} + o_p\left(\frac{1}{n}\right) \\
&= \frac{h_1(\delta^{\text{out}}_{n,j})}{\sqrt{n}} + \frac{1}{n}\left(\frac{h_2(\delta^{\text{out}}_{n,j})}{n} - \frac{1}{2}\left(h_1(\delta^{\text{out}}_{n,j})\right)^2\right) + o_p\left(\frac{1}{n}\right)
\end{aligned}$$

Plugging this in Eq. 25, we get that

$$\log\left(d_{n,j}\left(\delta_{n,j}^{\text{out}}\right)\right) = -\frac{1}{2}\log(2\pi) - \frac{(\delta_{n,j}^{\text{out}})^2}{2} + \frac{h_1(\delta_{n,j}^{\text{out}})}{\sqrt{n}} + \frac{1}{n}\left(\frac{h_2(\delta_{n,j}^{\text{out}})}{n} - \frac{1}{2}\left(h_1(\delta_{n,j}^{\text{out}})\right)^2\right) + o_p\left(\frac{1}{n}\right)$$

Similarly, we can show that

$$\log\left(d_{n-1,j}\left(\delta_{n,j}^{\text{in}}\right)\right) = -\frac{1}{2}\log(2\pi) - \frac{(\delta_{n,j}^{\text{in}})^2}{2} + \frac{h_1(\delta_{n,j}^{\text{in}})}{\sqrt{n}} + \frac{1}{n}\left(\frac{h_2(\delta_{n,j}^{\text{in}})}{n} - \frac{1}{2}\left(h_1(\delta_{n,j}^{\text{in}})\right)^2\right) + o_p\left(\frac{1}{n}\right)$$

Step4: Asymptotic expansion of polynomials in $\delta_{n,j}^{\text{in}}$ and $\delta_{n,j}^{\text{out}}$

First, we have

$$\begin{aligned}\delta_{n,j}^{\text{in}} &= \sqrt{\frac{n}{n-1}}\delta_{n,j}^{\text{out}} + \frac{\mu_j - z_j^\star}{\sigma_j\sqrt{n-1}} \\ &= \delta_{n,j}^{\text{out}} + \frac{\mu_j - z_j^\star}{\sigma_j\sqrt{n}} + o_p\left(\frac{1}{\sqrt{n}}\right)\end{aligned}$$

Which gives

$$\begin{aligned}h_1\left(\delta_{n,j}^{\text{out}}\right) - h_1\left(\delta_{n,j}^{\text{in}}\right) &= \lambda_3\left(\left(\delta_{n,j}^{\text{out}}\right)^3 - \left(\delta_{n,j}^{\text{in}}\right)^3 + \left(\delta_{n,j}^{\text{out}} - \delta_{n,j}^{\text{in}}\right)\right) \\ &= \lambda_3\left(\delta_{n,j}^{\text{out}} - \delta_{n,j}^{\text{in}}\right)\left(\left(\delta_{n,j}^{\text{out}}\right)^2 + \delta_{n,j}^{\text{out}}\delta_{n,j}^{\text{in}} + \left(\delta_{n,j}^{\text{in}}\right)^2 - 3\right) \\ &= \lambda_3 R_{n,j}\left(\frac{z_j^\star - \mu_j}{\sigma_j\sqrt{n}} + o_p\left(\frac{1}{\sqrt{n}}\right)\right)\end{aligned}$$

where $R_{n,j} \triangleq \left(\delta_{n,j}^{\text{out}}\right)^2 + \delta_{n,j}^{\text{out}}\delta_{n,j}^{\text{in}} + \left(\delta_{n,j}^{\text{in}}\right)^2 - 3$, which can be written as a polynomial in $\delta_{n,j}^{\text{out}}$.
Thus $R_{n,j} = O_p(1)$, which means that

$$\begin{aligned}h_1\left(\delta_{n,j}^{\text{out}}\right) - h_1\left(\delta_{n,j}^{\text{in}}\right) &= \lambda_3 R_{n,j}\left(\frac{z_j^\star - \mu_j}{\sigma_j\sqrt{n}}\right) + o_p\left(\frac{1}{\sqrt{n}}\right) \\ &= O_p\left(\frac{1}{\sqrt{n}}\right)\end{aligned}$$

Similarly, one can show that

$$h_2\left(\delta_{n,j}^{\text{out}}\right) - h_2\left(\delta_{n,j}^{\text{in}}\right) = O_p\left(\frac{1}{\sqrt{n}}\right)$$

and

$$h_1^2\left(\delta_{n,j}^{\text{out}}\right) - h_1^2\left(\delta_{n,j}^{\text{in}}\right) = O_p\left(\frac{1}{\sqrt{n}}\right).$$

Step5: Summarising the asymptotic expansion of the LR score

All in all, we get that

$$\log\left(\frac{d_{n-1,j}\left(\delta_{n,j}^{\text{in}}\right)}{d_{n,j}\left(\delta_{n,j}^{\text{out}}\right)}\right) = \frac{1}{2}\left(\left(\delta_{n,j}^{\text{out}}\right)^2 - \left(\delta_{n,j}^{\text{in}}\right)^2\right) + \frac{\lambda_3\left(\mu_j - z_j^\star\right)}{n\sigma_j}R_{n,j} + o_p\left(\frac{1}{n}\right)$$

□

Lemma C.9. *Expectation and variance computations*

$$\mathbb{E}_0[Y_{n,j}] = -\frac{1}{2n} - \frac{(z_j^\star - \mu_j)^2}{2n\sigma_j^2} + o\left(\frac{1}{n}\right) \qquad V_0[Y_{n,j}] = \frac{(z_j^\star - \mu_j)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right)$$
$$\mathbb{E}_1[Y_{n,j}] = -\frac{1}{2n} + \frac{(z_j^\star - \mu_j)^2}{2n\sigma_j^2} + o\left(\frac{1}{n}\right) \qquad V_1[Y_{n,j}] = \frac{(z_j^\star - \mu_j)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right)$$

Proof. The proof is direct from expectation and variance of the mean under H_0 and H_1. Specifically, under H_O we have that $\mathbb{E}_0(\hat{\mu}_{n,j}) = \mu_j$ and $\mathbb{V}_0(\hat{\mu}_{n,j}) = \frac{1}{n}\sigma_j^2$. On the other hand, under H_1, we have that $\mathbb{E}_1(\hat{\mu}_{n,j}) = \mu_j + \frac{1}{n}\left(z_j^\star - \mu_j\right)$ and $\mathbb{V}_1(\hat{\mu}_{n,j}) = \frac{n-1}{n^2}\sigma_j^2$. Here, we present the exact derivations for completeness.

Let us recall that

$$Y_{n,j} \triangleq \frac{1}{2}\left(\left(\delta_{n,j}^{\text{out}}\right)^2 - \left(\delta_{n,j}^{\text{in}}\right)^2\right) + \frac{\lambda_3\left(\mu_j - z_j^\star\right)}{n\sigma_j} R_{n,j}$$

where $R_{n,j} \triangleq \left(\delta_{n,j}^{\text{out}}\right)^2 + \delta_{n,j}^{\text{out}}\delta_{n,j}^{\text{in}} + \left(\delta_{n,j}^{\text{in}}\right)^2 - 3$

Under H_0:

Since $\mathbb{E}_0(\hat{\mu}_{n,j}) = \mu_j$ and $\mathbb{V}_0(\hat{\mu}_{n,j}) = \frac{1}{n}\sigma_j^2$, we get that

$$\mathbb{E}_0\left(\delta_{n,j}^{\text{out}}\right) = 0 \qquad \mathbb{E}_0\left(\left(\delta_{n,j}^{\text{out}}\right)^2\right) = 1$$
$$\mathbb{E}_0\left(\delta_{n,j}^{\text{in}}\right) = \frac{\mu_j - z_j^\star}{\sigma_j\sqrt{n-1}} \qquad \mathbb{E}_0\left(\left(\delta_{n,j}^{\text{in}}\right)^2\right) = 1 + \frac{1}{n} + \frac{\left(\mu_j - z_j^\star\right)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right)$$

Also

$$\mathbb{E}_0\left(\delta_{n,j}^{\text{out}}\delta_{n,j}^{\text{in}}\right) = \sqrt{\frac{n}{n-1}} = 1 + \frac{1}{2n} + o\left(\frac{1}{n}\right)$$

Which means that

$$\mathbb{E}_0\left(R_{n,j}\right) = \frac{3}{2n} + \frac{\left(\mu_j - z_j^\star\right)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right) = o(1)$$

Finally

$$\mathbb{E}_0[Y_{n,j}] = -\frac{1}{2n} - \frac{(z_j^\star - \mu_j)^2}{2n\sigma_j^2} + o\left(\frac{1}{n}\right)$$

On the other hand,

$$\begin{aligned}
\mathbb{V}_0[Y_{n,j}] &= \mathbb{E}_0[Y_{n,j}^2] - \left(\mathbb{E}_0[Y_{n,j}]\right)^2 \\
&= \mathbb{E}_0[Y_{n,j}^2] + o\left(\frac{1}{n}\right) \\
&= \frac{1}{4}\,\mathbb{E}_0\left[\left(\left(\delta_{n,j}^{\text{out}}\right)^2 - \left(\delta_{n,j}^{\text{in}}\right)^2\right)^2\right] + \frac{\lambda_3^2}{n^2\sigma_j^2}\left(\mu_j - z_j^\star\right)^2\mathbb{E}_0[R_{n,j}^2] \\
&+ \frac{\lambda_3}{n\sigma_j}\left(\mu_j - z_j^\star\right)\mathbb{E}_0\left[R_{n,j}\left(\left(\delta_{n,j}^{\text{out}}\right)^2 - \left(\delta_{n,j}^{\text{in}}\right)^2\right)\right]
\end{aligned}$$

We compute the following expectations:

$$\begin{aligned}
\mathbb{E}_0[\left(\delta_{n,j}^{\text{out}}\right)^3] &= \frac{n^{3/2}}{\sigma_j^3}\,\mathbb{E}_0\left[\left(\hat{\mu}_{n,j}-\mu_j\right)^3\right] \\
&= \frac{1}{\sqrt{n}}\frac{\mu_{3,j}}{\sigma_j^3} \\
\mathbb{E}_0[\left(\delta_{n,j}^{\text{out}}\right)^4] &= \frac{n^2}{\sigma_j^4}\,\mathbb{E}_0\left[\left(\hat{\mu}_{n,j}-\mu_j\right)^4\right] \\
&= 3+\frac{1}{n}\left(\frac{\mu_{4,j}}{\sigma_j^4}-3\right) \\
\mathbb{E}_0[\left(\delta_{n,j}^{\text{in}}\right)^4] &= \mathbb{E}_0\left[\left(\sqrt{\frac{n}{n-1}}\delta_{n,j}^{\text{out}}+\frac{\mu_j-z_j^\star}{\sigma_j\sqrt{n-1}}\right)^4\right] \\
&= 3+\frac{1}{n}\left(\frac{\mu_{4,j}}{\sigma_j^4}-3\right)+\frac{6}{n}+\frac{4}{n}\frac{\mu_j-z_j^\star}{\sigma_j^4}\mu_{3,j}+\frac{6}{n}\frac{(\mu_j-z_j^\star)^2}{\sigma_j^2}+o\left(\frac{1}{n}\right) \\
\mathbb{E}_0[\left(\delta_{n,j}^{\text{out}}\right)^2\left(\delta_{n,j}^{\text{in}}\right)^2] &= \mathbb{E}_0\left[\left(\delta_{n,j}^{\text{out}}\right)^2\left(\sqrt{\frac{n}{n-1}}\delta_{n,j}^{\text{out}}+\frac{\mu_j-z_j^\star}{\sigma_j\sqrt{n-1}}\right)^2\right] \\
&= 3+\frac{1}{n}\left(\frac{\mu_{4,j}}{\sigma_j^4}-3\right)+\frac{3}{n}+\frac{2}{n}\frac{\mu_j-z_j^\star}{\sigma_j^4}\mu_{3,j}+\frac{1}{n}\frac{(\mu_j-z_j^\star)^2}{\sigma_j^2}+o\left(\frac{1}{n}\right)
\end{aligned}$$

All in all, we have

$$\begin{aligned}
\mathbb{E}_0\left[\left(\left(\delta_{n,j}^{\text{out}}\right)^2-\left(\delta_{n,j}^{\text{in}}\right)^2\right)^2\right] &= \mathbb{E}_0[\left(\delta_{n,j}^{\text{out}}\right)^4]+\mathbb{E}_0[\left(\delta_{n,j}^{\text{in}}\right)^4]-2\,\mathbb{E}_0[\left(\delta_{n,j}^{\text{out}}\right)^2\left(\delta_{n,j}^{\text{in}}\right)^2] \\
&= \frac{4}{n}\frac{(\mu_j-z_j^\star)^2}{\sigma_j^2}+o\left(\frac{1}{n}\right) \\
\mathbb{E}_0[R_{n,j}^2] &= O(1) \\
\mathbb{E}_0\left[R_{n,j}\left(\left(\delta_{n,j}^{\text{out}}\right)^2-\left(\delta_{n,j}^{\text{in}}\right)^2\right)\right] &= o(1)
\end{aligned}$$

Thus

$$\mathbb{V}_0[Y_{n,j}] = \frac{1}{n}\frac{(\mu_j-z_j^\star)^2}{\sigma_j^2}+o\left(\frac{1}{n}\right)$$

Under H_1:

Since $\mathbb{E}_1(\hat{\mu}_{n,j}) = \mu_j+\frac{1}{n}\left(z_j^\star-\mu_j\right)$ and $\mathbb{V}_1(\hat{\mu}_{n,j}) = \frac{n-1}{n^2}\sigma_j^2$, we get that

$$\begin{aligned}
\mathbb{E}_1\left(\delta_{n,j}^{\text{out}}\right) &= \frac{z_j^\star-\mu_j}{\sigma_j\sqrt{n}} & \mathbb{E}_1\left(\left(\delta_{n,j}^{\text{out}}\right)^2\right) &= 1-\frac{1}{n}+\frac{\left(\mu_j-z_j^\star\right)^2}{n\sigma_j^2} \\
\mathbb{E}_1\left(\delta_{n,j}^{\text{in}}\right) &= 0 & \mathbb{E}_1\left(\left(\delta_{n,j}^{\text{in}}\right)^2\right) &= 1
\end{aligned}$$

Also

$$\mathbb{E}_1\left(\delta_{n,j}^{\text{out}}\delta_{n,j}^{\text{in}}\right) = \sqrt{\frac{n-1}{n}} = 1-\frac{1}{2n}+o\left(\frac{1}{n}\right)$$

Which means that

$$\mathbb{E}_1\left(R_{n,j}\right) = -\frac{3}{2n}+\frac{\left(\mu_j-z_j^\star\right)^2}{n\sigma_j^2}+o\left(\frac{1}{n}\right) = o(1)$$

Finally

$$\mathbb{E}_1[Y_{n,j}] = -\frac{1}{2n} + \frac{(z_j^\star - \mu_j)^2}{2n\sigma_j^2} + o\left(\frac{1}{n}\right)$$

Under H_1, one can rewrite $\delta_{n,j}^{\text{in}} = \hat{Z}_{n-1,j}$ and $\delta_{n,j}^{\text{out}} = \sqrt{\frac{n-1}{n}}\delta_{n,j}^{\text{in}} + \frac{z_j^\star - \mu_j}{\sigma_j\sqrt{n}}$.

Thus using the same steps as in H_0, we get

$$\begin{aligned}
\mathbb{E}_1[\left(\delta_{n,j}^{\text{in}}\right)^3] &= \frac{1}{\sqrt{n-1}}\frac{\mu_{3,j}}{\sigma_j^3} \\
\mathbb{E}_1[\left(\delta_{n,j}^{\text{in}}\right)^4] &= 3 + \frac{1}{n-1}\left(\frac{\mu_{4,j}}{\sigma_j^4} - 3\right) \\
\mathbb{E}_1[\left(\delta_{n,j}^{\text{out}}\right)^4] &= 3 + \frac{1}{n}\left(\frac{\mu_{4,j}}{\sigma_j^4} - 3\right) + \frac{6}{n} + \frac{4}{n}\frac{z_j^\star - \mu_j}{\sigma_j^4}\mu_{3,j} + \frac{6}{n}\frac{(z_j^\star - \mu_j)^2}{\sigma_j^2} + o\left(\frac{1}{n}\right) \\
\mathbb{E}_1[\left(\delta_{n,j}^{\text{out}}\right)^2\left(\delta_{n,j}^{\text{in}}\right)^2] &= 3 + \frac{1}{n}\left(\frac{\mu_{4,j}}{\sigma_j^4} - 3\right) + \frac{3}{n} + \frac{2}{n-1}\frac{z_j^\star - \mu_j}{\sigma_j^4}\mu_{3,j} + \frac{1}{n}\frac{(z_j^\star - \mu_j)^2}{\sigma_j^2} + o\left(\frac{1}{n}\right)
\end{aligned}$$

All in all, we have

$$\begin{aligned}
\mathbb{E}_1\left[\left(\left(\delta_{n,j}^{\text{out}}\right)^2 - \left(\delta_{n,j}^{\text{in}}\right)^2\right)^2\right] &= \frac{4}{n}\frac{(\mu_j - z_j^\star)^2}{\sigma_j^2} + o\left(\frac{1}{n}\right) \\
\mathbb{E}_1[R_{n,j}^2] &= O(1) \\
\mathbb{E}_1\left[R_{n,j}\left(\left(\delta_{n,j}^{\text{out}}\right)^2 - \left(\delta_{n,j}^{\text{in}}\right)^2\right)\right] &= o(1)
\end{aligned}$$

Thus

$$\mathbb{V}_1[Y_{n,j}] = \frac{1}{n}\frac{(\mu_j - z_j^\star)^2}{\sigma_j^2} + o\left(\frac{1}{n}\right)$$

□

Lemma C.10. *The Lindeberg-Feller condition*

The random variables $(Y_{n,j})_{j=1}^{d_n}$ verify the Lindeberg-Feller condition.

Proof. Let $\epsilon > 0$. Let $h \in \{0, 1\}$.

Let $\delta > 0$. We have that

$$\begin{aligned}
\mathbb{E}_h\left[Y_{n,j}^2 \mathbb{1}\left\{|Y_{n,j}| > \epsilon\right\}\right] &= \mathbb{E}_h\left[\frac{Y_{n,j}^{2+\delta}}{Y_{n,j}^{\delta}}\mathbb{1}\left\{|Y_{n,j}| > \epsilon\right\}\right] \\
&\leq \frac{1}{\epsilon^\delta}\mathbb{E}_h\left[Y_{n,j}^{2+\delta}\right]
\end{aligned}$$

On the other hand, we have that $Y_{n,j} = \frac{1}{2}\left(\left(\delta_{n,j}^{\text{out}}\right)^2 - \left(\delta_{n,j}^{\text{in}}\right)^2\right) + \frac{\lambda_3\left(\mu_j - z_j^\star\right)}{n\sigma_j}R_{n,j}$, where

$$\left(\left(\delta_{n,j}^{\text{out}}\right)^2 - \left(\delta_{n,j}^{\text{in}}\right)^2\right) = O_p\left(\frac{1}{\sqrt{n}}\right) \text{ and } R_{n,j} = O_p(1)$$

Thus $Y_{n,j} = O_p\left(\frac{1}{\sqrt{n}}\right)$, and $\mathbb{E}_h\left[Y_{n,j}^{2+\delta}\right] = o\left(\frac{1}{n}\right)$.

Which means that $\mathbb{E}_h\left[Y_{n,j}^2 \mathbb{1}\left\{|Y_{n,j}| > \epsilon\right\}\right] = o\left(\frac{1}{n}\right)$ and

$$\sum_{j=1}^{d_n} \mathbb{E}_h\left[Y_{n,j}^2 \mathbb{1}\left\{|Y_{n,j}| > \epsilon\right\}\right] = o\left(\frac{d_n}{n}\right) = o(1) \to 0$$

□

C.4 Proof of Theorem 3.1

***Theorem** 3.1* (Target-dependent leakage of the empirical mean). The asymptotic target-dependent leakage of $z^\star$ in the empirical mean is

$$\lim_{n,d} \xi_n(z^\star, \mathcal{M}_n^{\text{emp}}, \mathcal{D}) = \Phi\left(\frac{\sqrt{m^\star}}{2}\right) - \Phi\left(-\frac{\sqrt{m^\star}}{2}\right)$$

The asymptotic trade-off function, achievable with threshold $\tau_\alpha = -\frac{m^\star}{2} + \sqrt{m^\star}\Phi^{-1}(1-\alpha)$, is

$$\lim_{n,d} \text{Pow}_n(\ell_n, \alpha, z^\star) = \Phi\left(\Phi^{-1}(\alpha) + \sqrt{m^\star}\right)$$

Proof. From the asymptotic distribution of the LR score, we get directly that

$$\begin{aligned}
\lim_{n,d} \xi_n(z^\star, \mathcal{M}_n^{\text{emp}}, \mathcal{D}) &= \Pr\left(\mathcal{N}\left(-\frac{m^\star}{2}, m^\star\right) < 0\right) - \Pr\left(\mathcal{N}\left(\frac{m^\star}{2}, m^\star\right) < 0\right) \\
&= \Phi\left(\frac{m^\star/2}{\sqrt{m^\star}}\right) - \Phi\left(-\frac{m^\star/2}{\sqrt{m^\star}}\right) \\
&= \Phi\left(\frac{\sqrt{m^\star}}{2}\right) - \Phi\left(-\frac{\sqrt{m^\star}}{2}\right)
\end{aligned}$$

The threshold τ_α for which the asymptotic LR attack achieves significance α verifies:

$$\Pr\left(\mathcal{N}\left(-\frac{m^\star}{2}, m^\star\right) \geq \tau_\alpha)\right) = \alpha$$

Thus $\tau_\alpha = -\frac{m^\star}{2} + \sqrt{m^\star}\Phi^{-1}(1-\alpha)$.

Finally, we find the power of the test by

$$\begin{aligned}
\lim_{n,d} \text{Pow}_n(\ell_n, \alpha, z^\star) &= \Pr\left(\mathcal{N}\left(\frac{m^\star}{2}, m^\star\right) \geq \tau_\alpha\right) \\
&= \Pr\left(\frac{m^\star}{2} + \sqrt{m^\star}\mathcal{N}(0,1) \geq -\frac{m^\star}{2} + \sqrt{m^\star}\Phi^{-1}(1-\alpha)\right) \\
&= \Pr\left(\sqrt{m^\star}\mathcal{N}(0,1) \geq -m^\star - \sqrt{m^\star}\Phi^{-1}(\alpha)\right) \\
&= \Pr\left(\mathcal{N}(0,1) \leq \sqrt{m^\star} + \Phi^{-1}(\alpha)\right) \\
&= \Phi\left(\Phi^{-1}(\alpha) + \sqrt{m^\star}\right)
\end{aligned}$$

□

C.5 Effects of Sub-sampling, Proof of Theorem 4.2

***Theorem** 4.2* (Target-dependent leakage for $\mathcal{M}_n^{\text{sub},\rho}$). First, we show that as $d, n \to \infty$ while $d/n = \tau$:

- Under H_0, $\ell_n^{\text{sub},\rho}(\hat{\mu}_n^{\text{sub}}; z^\star, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(-\rho\frac{m^\star}{2}, \rho m^\star\right)$.

- Under H_1, $\ell_n^{\text{sub},\rho}(\hat{\mu}_n^{\text{sub}}; z^\star, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(\rho\frac{m^\star}{2}, \rho m^\star\right)$.

The asymptotic target-dependent leakage of $z^\star$ in $\mathcal{M}_n^{\text{sub},\rho}$ is

$$\lim_{n,d} \xi_n(z^\star, \mathcal{M}_{n,\rho}^{\text{sub}}, \mathcal{D}) = \Phi\left(\frac{\sqrt{\rho m^\star}}{2}\right) - \Phi\left(-\frac{\sqrt{\rho m^\star}}{2}\right)$$

The optimal trade-off function obtained with $\tau_\alpha = -\frac{\rho m^\star}{2} + \sqrt{\rho m^\star}\Phi^{-1}(1-\alpha)$, is

$$\lim_{n,d} \text{Pow}_n(\ell_{n,\rho}^{\text{sub}}, \alpha, z^\star) = \Phi\left(z_\alpha + \sqrt{\rho m^\star}\right).$$

Proof. We have that $\hat{\mu}_n^{\text{sub}} = \frac{1}{k_n}\sum_{i=1}^n Z_i \mathbb{1}\{\varsigma(i) \leq k_n\}$, where $k_n \triangleq \rho n$, Z_i are i.i.d and $\varsigma \sim^{\text{unif}} S_n$ is a permutation sampled uniformly from the set of permutations of $\{1\dots,n\}$ i.e. S_n and independently from (Z_i).

We denote $\hat{\mu}_n^{\text{sub}} = (\hat{\mu}_{n,j}^{\text{sub}})_{j=1}^{d_n}$.

Step 1: Rewriting the LR score

Let $j \in [1, d_n]$.

Under H_0, we can re-write

$$\hat{\mu}_{n,j}^{\text{sub}} = \mu_j + \frac{\sigma_j}{\sqrt{k_n}}\hat{Z}_{k_n,j},$$

where

$$\begin{aligned}\hat{Z}_{d_n,j} &\triangleq \sqrt{k_n}\left(\frac{\hat{\mu}_{n,j}^{\text{sub}} - \mu_j}{\sigma}\right) \\ &= \frac{1}{\sqrt{k_n}}\sum_{i=1}^{k_n}\frac{Z_{\varsigma^{-1}(i),j} - \mu_j}{\sigma_j}.\end{aligned}$$

Since $(Z_{i,j})_{i=1}^n$ are i.i.d from $\mathcal{D}_j$, and $\varsigma \sim^{\text{unif}} S_n$ and ind. from (Z_i), then $(Z_{\varsigma^{-1}(i),j})_{i=1}^{k_n}$.

Using the CLT, $\hat{Z}_{d_n,j} \rightsquigarrow_{n\to\infty} \mathcal{N}(0,1)$.

Let $d_{n,j}$ be the density function of $\hat{Z}_{n,j}$.

The density $p_{n,j}^{\text{out,sub}}$ of $\hat{\mu}_{n,j}^{\text{sub}}$ under H_0 can be written as

$$p_{n,j}^{\text{out,sub}}(x; z_j^\star, \mu_j, \sigma_j) = \frac{\sqrt{k_n}}{\sigma_j} d_{k_n,j}\left[\frac{\sqrt{k_n}}{\sigma_j}(x-\mu_j)\right]$$

Under H_1, we can re-write

$$\hat{\mu}_{n,j}^{\text{sub}} = \frac{1}{k_n} z_j^\star \mathbb{1}\{\varsigma(n) \leq k_n\} + \frac{1}{k_n}\sum_{i=1}^{n-1} Z_i \mathbb{1}\{\varsigma(i) \leq k_n\}$$

Let $A = \{\mathbb{1}\{\varsigma(n) \leq k_n\}\}$ the event that $z^\star$ was sub-sampled. We have that $\Pr(A) = \rho$.

The density $p_{n,j}^{\text{in,sub}}$ of $\hat{\mu}_{n,j}^{\text{sub}}$ under H_1 and given A is

$$\frac{k_n}{\sigma_j\sqrt{k_n - 1}} d_{k_n-1,j}\left[\frac{k_n}{\sigma_j\sqrt{k_n-1}}\left(x - \mu_j - \frac{1}{k_n}\left(z_j^\star - \mu_j\right)\right)\right]$$

The density $p_{n,j}^{\text{in,sub}}$ of $\hat{\mu}_{n,j}^{\text{sub}}$ under H_1 and given A^c is

$$\frac{\sqrt{k_n}}{\sigma_j} d_{k_n,j}\left[\frac{\sqrt{k_n}}{\sigma_j}(x-\mu_j)\right]$$

Thus, the density $p_{n,j}^{\text{in,sub}}$ of $\hat{\mu}_{n,j}^{\text{sub}}$ under H_1 can be written as

$$p_{n,j}^{\text{in,sub}}(x; z_j^\star, \mu_j, \sigma_j) = (1-\rho)\frac{\sqrt{k_n}}{\sigma_j} d_{k_n,j}\left[\frac{\sqrt{k_n}}{\sigma_j}(x-\mu_j)\right] + \rho\frac{k_n}{\sigma_j\sqrt{k_n-1}} d_{k_n-1,j}\left[\frac{k_n}{\sigma_j\sqrt{k_n-1}}\left(x-\mu_j-\frac{1}{k_n}\left(z_j^\star-\mu_j\right)\right)\right]$$

The additional technical hardness of this proof comes from the 'mixture' nature of the 'in' distribution.

The LR score is

$$\begin{aligned}\ell_n^{\text{sub},\rho}(\hat{\mu}_n^{\text{sub}}; z^\star, \mu, C_\sigma) &= \sum_{j=1}^{d_n} \log\left(\frac{p_{n,j}^{\text{in,sub}}(\hat{\mu}_{n,j}^{\text{sub}}; z_j^\star, \mu_j, \sigma_j)}{p_{n,j}^{\text{out,sub}}(\hat{\mu}_{n,j}^{\text{sub}}; z_j^\star, \mu_j, \sigma_j)}\right)\\ &= \sum_{j=1}^{d_n} \log\left(\frac{(1-\rho)\frac{\sqrt{k_n}}{\sigma_j} d_{k_n,j}\left(\delta_{k_n,j}^{\text{out,sub}}\right) + \rho\frac{k_n}{\sigma_j\sqrt{k_n-1}} d_{k_n-1,j}\left(\delta_{k_n,j}^{\text{in,sub}}\right)}{\frac{\sqrt{k_n}}{\sigma_j} d_{k_n,j}\left(\delta_{k_n,j}^{\text{out,sub}}\right)}\right)\\ &= \sum_{j=1}^{d_n} \log\left((1-\rho) + \rho\sqrt{\frac{k_n}{k_n-1}}\frac{d_{k_n-1,j}\left(\delta_{k_n,j}^{\text{in,sub}}\right)}{d_{k_n,j}\left(\delta_{k_n,j}^{\text{out,sub}}\right)}\right)\end{aligned}$$

where

$$\begin{aligned}\delta_{k_n,j}^{\text{out,sub}} &\triangleq \frac{\sqrt{k_n}}{\sigma_j}\left(\hat{\mu}_{n,j}^{\text{sub}}-\mu_j\right)\\ \delta_{k_n,j}^{\text{in,sub}} &\triangleq \frac{k_n}{\sqrt{k_n-1}\sigma_j}\left(\hat{\mu}_{n,j}^{\text{sub}}-\mu_j+\frac{1}{k_n}\left(\mu_j-z_j^\star\right)\right)\end{aligned}$$

Step 2: Asymptotic expansion of the LR score

Using Lemma C.11, we have

$$\log\left((1-\rho) + \rho\sqrt{\frac{k_n}{k_n-1}}\frac{d_{k_n-1,j}\left(\delta_{k_n,j}^{\text{in,sub}}\right)}{d_{k_n,j}\left(\delta_{k_n,j}^{\text{out,sub}}\right)}\right) = W_{n,j} + o_p\left(\frac{1}{n}\right)$$

where

$$W_{n,j} \triangleq \frac{\rho}{2}(\delta_{k_n,j}^{\text{out,sub}})^2 - \frac{\rho}{2}(\delta_{k_n,j}^{\text{in,sub}})^2 + \rho\frac{\lambda_3\left(\mu_j-z_j^\star\right)}{k_n\sigma_j} R_{k_n,j} + \frac{\rho}{2k_n} + \frac{\rho(1-\rho)}{8}\left((\delta_{k_n,j}^{\text{out,sub}})^2 - (\delta_{k_n,j}^{\text{in,sub}})^2\right)^2$$

The extra hardness of this proof comes from the fact that the density under H_1 is now a mixture of two Gaussians, rather than just one Gaussian in the case of the exact empirical mean.

Thus

$$\ell_n^{\text{sub},\rho}(\hat{\mu}_n^{\text{sub}}; z^\star, \mu, C_\sigma) = o_p(1) + \sum_{j=1}^{d_n} W_{n,j}$$

because $\frac{d_n}{n} = \tau + o(1)$.

Step3: Concluding using the Lindeberg-Feller CLT

Under H_0:

Using Lemma C.12, $\mathbb{E}_0[W_n, j] = -\frac{\rho}{2}\frac{\left(\mu_j - z_j^\star\right)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right)$ and $\mathbb{V}_0[W_{n,j}] = \rho\frac{(\mu_j - z_j^\star)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right)$.

Since $\sum_{j=1}^{d_n}\frac{(z_j^\star - \mu_j)^2}{n\sigma_j^2} = \frac{\|z^\star - \mu\|^2_{C_\sigma^{-1}}}{n}$, we get:

- $\sum_{j=1}^{d_n}\mathbb{E}_0[W_n, j] \to -\frac{\rho m^\star}{2}$
- $\sum_{j=1}^{d_n} V_0[W_n, j] \to \rho m^\star$

Similarly to Lemma C.10, we can show that $W_{n,j}$ verify the Lindeberg-Feller condition, i.e.

$$\sum_{j=1}^{d_n}\mathbb{E}_0\left[W_{n,j}^2 \mathbb{1}\left\{|W_{n,j}| > \epsilon\right\}\right] \to 0$$

for every $\epsilon > 0$.

We conclude using the Lindeberg-Feller CLT (Theorem C.4) that

$$\ell_n^{\text{sub},\rho}(\hat{\mu}_n^{\text{sub}}; z^\star, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(-\rho\frac{m^\star}{2}, \rho m^\star\right)$$

Similarly, *Under H_1:*

Using Lemma C.12, $\mathbb{E}_1[W_n, j] = \frac{\rho}{2}\frac{\left(\mu_j - z_j^\star\right)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right)$ and $\mathbb{V}_1[W_{n,j}] = \rho\frac{(z_j^\star - \mu_j)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right)$.

We get:

- $\sum_{j=1}^{d_n}\mathbb{E}_1[W_n, j] \to \frac{\rho m^\star}{2}$
- $\sum_{j=1}^{d_n} V_1[W_n, j] \to \rho m^\star$

Similarly to Lemma C.10, we can show that $W_{n,j}$ verify the Lindeberg-Feller condition, i.e.

$$\sum_{j=1}^{d_n}\mathbb{E}_1\left[W_{n,j}^2 \mathbb{1}\left\{|W_{n,j}| > \epsilon\right\}\right] \to 0$$

for every $\epsilon > 0$.

We conclude using the Lindeberg-Feller CLT (Theorem C.4) that

$$\ell_n^{\text{sub},\rho}(\hat{\mu}_n^{\text{sub}}; z^\star, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(\rho\frac{m^\star}{2}, \rho m^\star\right)$$

Step4: Characterising the advantage and the power function

Using the same step as in the proof of Corollary 3.1, we conclude.

□

Now we present the proof of the helpful technical lemmas.

Lemma C.11 (Asymptotic expansion of the LR score for sub-sampling)**.** *We show that*

$$\log\left((1-\rho) + \rho\sqrt{\frac{k_n}{k_n - 1}}\frac{d_{k_n-1,j}\left(\delta_{k_n,j}^{\text{in,sub}}\right)}{d_{k_n,j}\left(\delta_{k_n,j}^{\text{out,sub}}\right)}\right) = W_{n,j} + o_p\left(\frac{1}{n}\right)$$

where

$$W_{n,j} \triangleq \frac{\rho}{2}(\delta^{\text{out,sub}}_{k_n,j})^2 - \frac{\rho}{2}(\delta^{\text{in,sub}}_{k_n,j})^2 + \rho\frac{\lambda_3\left(\mu_j - z^\star_j\right)}{k_n\sigma_j}R_{k_n,j} + \frac{\rho}{2k_n} + \frac{\rho(1-\rho)}{8}\left((\delta^{\text{out,sub}}_{k_n,j})^2 - (\delta^{\text{in,sub}}_{k_n,j})^2\right)^2$$

Proof. First, we recall that

$$d_{k_n,j}\left(\delta^{\text{out,sub}}_{k_n,j}\right) = \frac{1}{\sqrt{2\pi}}e^{-\frac{(\delta^{\text{out,sub}}_{k_n,j})^2}{2}}\left(1 + \frac{h_1(\delta^{\text{out,sub}}_{k_n,j})}{\sqrt{k_n}} + \frac{h_2(\delta^{\text{out,sub}}_{k_n,j})}{k_n} + o_p\left(\frac{1}{n}\right)\right)$$

Taking the inverse, and developing it we get

$$\begin{aligned}
&\left(d_{k_n,j}\left(\delta^{\text{out,sub}}_{k_n,j}\right)\right)^{-1}\\
=&\sqrt{2\pi}e^{\frac{(\delta^{\text{out,sub}}_{k_n,j})^2}{2}}\left(1 + \frac{h_1(\delta^{\text{out,sub}}_{k_n,j})}{\sqrt{k_n}} + \frac{h_2(\delta^{\text{out,sub}}_{k_n,j})}{k_n} + o_p\left(\frac{1}{n}\right)\right)^{-1}\\
=&\sqrt{2\pi}e^{\frac{(\delta^{\text{out,sub}}_{k_n,j})^2}{2}}\left(1 - \frac{h_1(\delta^{\text{out,sub}}_{k_n,j})}{\sqrt{k_n}} - \frac{h_2(\delta^{\text{out,sub}}_{k_n,j})}{k_n} + \frac{1}{2}\frac{\left(h_1(\delta^{\text{out,sub}}_{k_n,j})\right)^2}{k_n} + o_p\left(\frac{1}{n}\right)\right)
\end{aligned}$$

Combining with the other result

$$d_{k_n-1,j}\left(\delta^{\text{in,sub}}_{k_n,j}\right) = \frac{1}{\sqrt{2\pi}}e^{-\frac{(\delta^{\text{in,sub}}_{k_n,j})^2}{2}}\left(1 + \frac{h_1(\delta^{\text{in,sub}}_{k_n,j})}{\sqrt{k_n}} + \frac{h_2(\delta^{\text{in,sub}}_{k_n,j})}{k_n} + o_p\left(\frac{1}{n}\right)\right)$$

Putting them together gives

$$\begin{aligned}
&\frac{d_{k_n-1,j}\left(\delta^{\text{in,sub}}_{k_n,j}\right)}{d_{k_n,j}\left(\delta^{\text{out,sub}}_{k_n,j}\right)}\\
=&e^{\frac{(\delta^{\text{out,sub}}_{k_n,j})^2-(\delta^{\text{in,sub}}_{k_n,j})^2}{2}}\left(1 - \frac{h_1(\delta^{\text{out,sub}}_{k_n,j})}{\sqrt{k_n}} - \frac{h_2(\delta^{\text{out,sub}}_{k_n,j})}{k_n} + \frac{1}{2}\frac{\left(h_1(\delta^{\text{out,sub}}_{k_n,j})\right)^2}{k_n} + o_p\left(\frac{1}{n}\right)\right)\\
&\left(1 + \frac{h_1(\delta^{\text{in,sub}}_{k_n,j})}{\sqrt{k_n}} + \frac{h_2(\delta^{\text{in,sub}}_{k_n,j})}{k_n} + o_p\left(\frac{1}{n}\right)\right)\\
=&e^{\frac{(\delta^{\text{out,sub}}_{k_n,j})^2-(\delta^{\text{in,sub}}_{k_n,j})^2}{2}}\left(1 + \frac{h_1(\delta^{\text{in,sub}}_{k_n,j}) - h_1(\delta^{\text{out,sub}}_{k_n,j})}{\sqrt{k_n}}\right.\\
&\left.+ \frac{1}{k_n}\left(\frac{1}{2}\left(h_1(\delta^{\text{out,sub}}_{k_n,j})\right)^2 - h_1(\delta^{\text{in,sub}}_{k_n,j})h_1(\delta^{\text{out,sub}}_{k_n,j})\right) + o_p\left(\frac{1}{n}\right)\right)
\end{aligned}$$

Now, we proceed to do Taylor expansion of the exponential function, to get

$$e^{\frac{(\delta^{\text{out,sub}}_{k_n,j})^2-(\delta^{\text{in,sub}}_{k_n,j})^2}{2}} = 1 + \frac{1}{2}(\delta^{\text{out,sub}}_{k_n,j})^2 - \frac{1}{2}(\delta^{\text{in,sub}}_{k_n,j})^2 + \frac{1}{8}\left((\delta^{\text{out,sub}}_{k_n,j})^2 - (\delta^{\text{in,sub}}_{k_n,j})^2\right)^2 + o_p\left(\frac{1}{n}\right)$$

Putting everything together, we get

$$\begin{aligned}
&(1-\rho)+\rho\sqrt{\frac{k_n}{k_n-1}}\frac{d_{k_n-1,j}\left(\delta^{\text{in,sub}}_{k_n,j}\right)}{d_{k_n,j}\left(\delta^{\text{out,sub}}_{k_n,j}\right)}\\
=&1+\frac{\rho}{2k_n}+\frac{\rho}{2}(\delta^{\text{out,sub}}_{k_n,j})^2-\frac{\rho}{2}(\delta^{\text{in,sub}}_{k_n,j})^2+\frac{\rho}{8}\left((\delta^{\text{out,sub}}_{k_n,j})^2-(\delta^{\text{in,sub}}_{k_n,j})^2\right)^2\\
&+\rho\frac{h_1(\delta^{\text{in,sub}}_{k_n,j})-h_1(\delta^{\text{out,sub}}_{k_n,j})}{\sqrt{k_n}}+\frac{\rho}{k_n}\left(\frac{1}{2}\left(h_1(\delta^{\text{out,sub}}_{k_n,j})\right)^2-h_1(\delta^{\text{in,sub}}_{k_n,j})h_1(\delta^{\text{out,sub}}_{k_n,j})\right)+o_p\left(\frac{1}{n}\right)
\end{aligned}$$

Finally, another Taylor expansion of the logarithm gives

$$\begin{aligned}
&\log\left((1-\rho)+\rho\sqrt{\frac{k_n}{k_n-1}}\frac{d_{k_n-1,j}\left(\delta^{\text{in,sub}}_{k_n,j}\right)}{d_{k_n,j}\left(\delta^{\text{out,sub}}_{k_n,j}\right)}\right)\\
=&1+\frac{\rho}{2k_n}+\frac{\rho}{2}(\delta^{\text{out,sub}}_{k_n,j})^2-\frac{\rho}{2}(\delta^{\text{in,sub}}_{k_n,j})^2+\frac{\rho}{8}\left((\delta^{\text{out,sub}}_{k_n,j})^2-(\delta^{\text{in,sub}}_{k_n,j})^2\right)^2\\
&+\rho\frac{h_1(\delta^{\text{in,sub}}_{k_n,j})-h_1(\delta^{\text{out,sub}}_{k_n,j})}{\sqrt{k_n}}+\frac{\rho}{k_n}\left(\frac{1}{2}\left(h_1(\delta^{\text{out,sub}}_{k_n,j})\right)^2-h_1(\delta^{\text{in,sub}}_{k_n,j})h_1(\delta^{\text{out,sub}}_{k_n,j})\right)+o_p\left(\frac{1}{n}\right)\\
&-\frac{1}{2}\frac{\rho^2}{4}\left((\delta^{\text{out,sub}}_{k_n,j})^2-(\delta^{\text{in,sub}}_{k_n,j})^2\right)^2-\frac{\rho^2}{2k_n}\left(h_1(\delta^{\text{in,sub}}_{k_n,j})-h_1(\delta^{\text{out,sub}}_{k_n,j})\right)^2
\end{aligned}$$

which after simplifications

$$\begin{aligned}
&\log\left((1-\rho)+\rho\sqrt{\frac{k_n}{k_n-1}}\frac{d_{k_n-1,j}\left(\delta^{\text{in,sub}}_{k_n,j}\right)}{d_{k_n,j}\left(\delta^{\text{out,sub}}_{k_n,j}\right)}\right)\\
=&\frac{\rho}{2}(\delta^{\text{out,sub}}_{k_n,j})^2-\frac{\rho}{2}(\delta^{\text{in,sub}}_{k_n,j})^2+\rho\frac{h_1(\delta^{\text{in,sub}}_{k_n,j})-h_1(\delta^{\text{out,sub}}_{k_n,j})}{\sqrt{k_n}}\\
&+\frac{\rho}{2k_n}+\frac{\rho(1-\rho)}{8}\left((\delta^{\text{out,sub}}_{k_n,j})^2-(\delta^{\text{in,sub}}_{k_n,j})^2\right)^2+o_p\left(\frac{1}{n}\right)
\end{aligned}$$

concludes the proof.

□

Lemma C.12. *Expectation and variance computations for sub-sampling*

$$\begin{aligned}
\mathbb{E}_0[W_{n,j}]&=-\frac{\rho}{2}\frac{\left(\mu_j-z^\star_j\right)^2}{n\sigma_j^2}+o\left(\frac{1}{n}\right) & \mathbb{V}_0[W_{n,j}]&=\rho\frac{(\mu_j-z^\star_j)^2}{n\sigma_j^2}+o\left(\frac{1}{n}\right)\\
\mathbb{E}_1[W_{n,j}]&=\frac{\rho}{2}\frac{\left(\mu_j-z^\star_j\right)^2}{n\sigma_j^2}+o\left(\frac{1}{n}\right) & \mathbb{V}_1[W_{n,j}]&=\rho\frac{(z^\star_j-\mu_j)^2}{n\sigma_j^2}+o\left(\frac{1}{n}\right)
\end{aligned}$$

Proof. Let us recall that

$$W_{n,j}\triangleq\frac{\rho}{2}(\delta^{\text{out,sub}}_{k_n,j})^2-\frac{\rho}{2}(\delta^{\text{in,sub}}_{k_n,j})^2+\rho\frac{\lambda_3\left(\mu_j-z^\star_j\right)}{k_n\sigma_j}R_{k_n,j}+\frac{\rho}{2k_n}+\frac{\rho(1-\rho)}{8}\left((\delta^{\text{out,sub}}_{k_n,j})^2-(\delta^{\text{in,sub}}_{k_n,j})^2\right)^2$$

where $R_{k_n,j}\triangleq\left(\delta^{\text{out,sub}}_{k_n,j}\right)^2+\delta^{\text{out,sub}}_{k_n,j}\delta^{\text{in,sub}}_{k_n,j}+\left(\delta^{\text{in,sub}}_{k_n,j}\right)^2-3$

Under H_0:

Since $\mathbb{E}_0(\hat{\mu}_{n,j}^{\text{sub}}) = \mu_j$ and $\mathbb{V}_0(\hat{\mu}_{n,j}^{\text{sub}}) = \frac{1}{k_n}\sigma_j^2$, we get that

$$\mathbb{E}_0\left(\delta_{k_n,j}^{\text{out,sub}}\right) = 0 \qquad \mathbb{E}_0\left(\left(\delta_{k_n,j}^{\text{out,sub}}\right)^2\right) = 1$$

$$\mathbb{E}_0\left(\delta_{k_n,j}^{\text{in,sub}}\right) = \frac{\mu_j - z_j^\star}{\sigma_j\sqrt{k_n - 1}} \qquad \mathbb{E}_0\left(\left(\delta_{k_n,j}^{\text{in,sub}}\right)^2\right) = 1 + \frac{1}{k_n} + \frac{\left(\mu_j - z_j^\star\right)^2}{k_n\sigma_j^2} + o\left(\frac{1}{n}\right)$$

Also

$$\mathbb{E}_0\left(\delta_{k_n,j}^{\text{out,sub}}\delta_{k_n,j}^{\text{in,sub}}\right) = \sqrt{\frac{k_n}{k_n - 1}} = 1 + \frac{1}{2k_n} + o\left(\frac{1}{n}\right)$$

Which means that

$$\mathbb{E}_0\left(R_{k_n,j}\right) = \frac{3}{2k_n} + \frac{\left(\mu_j - z_j^\star\right)^2}{k_n\sigma_j^2} + o\left(\frac{1}{n}\right) = o(1)$$

We compute the following expectations:

$$\mathbb{E}_0[\left(\delta_{k_n,j}^{\text{out,sub}}\right)^3] = \frac{1}{\sqrt{k_n}}\frac{\mu_{3,j}}{\sigma_j^3}$$

$$\mathbb{E}_0[\left(\delta_{k_n,j}^{\text{out,sub}}\right)^4] = 3 + \frac{1}{k_n}\left(\frac{\mu_{4,j}}{\sigma_j^4} - 3\right)$$

$$\mathbb{E}_0[\left(\delta_{k_n,j}^{\text{in,sub}}\right)^4] = 3 + \frac{1}{k_n}\left(\frac{\mu_{4,j}}{\sigma_j^4} - 3\right) + \frac{6}{k_n} + \frac{4}{k_n}\frac{\mu_j - z_j^\star}{\sigma_j^4}\mu_{3,j} + \frac{6}{k_n}\frac{(\mu_j - z_j^\star)^2}{\sigma_j^2} + o\left(\frac{1}{n}\right)$$

$$\mathbb{E}_0[\left(\delta_{k_n,j}^{\text{out,sub}}\right)^2\left(\delta_{k_n,j}^{\text{in,sub}}\right)^2] = 3 + \frac{1}{k_n}\left(\frac{\mu_{4,j}}{\sigma_j^4} - 3\right) + \frac{3}{k_n} + \frac{2}{k_n}\frac{\mu_j - z_j^\star}{\sigma_j^4}\mu_{3,j} + \frac{1}{k_n}\frac{(\mu_j - z_j^\star)^2}{\sigma_j^2} + o\left(\frac{1}{n}\right)$$

All in all, we have

$$\mathbb{E}_0\left[\left(\left(\delta_{k_n,j}^{\text{out,sub}}\right)^2 - \left(\delta_{k_n,j}^{\text{in,sub}}\right)^2\right)^2\right] = \frac{4}{k_n}\frac{(\mu_j - z_j^\star)^2}{\sigma_j^2} + o\left(\frac{1}{n}\right)$$

Finally

$$\begin{aligned}\mathbb{E}_0[W_{n,j}] &= -\rho\frac{\left(\mu_j - z_j^\star\right)^2}{2k_n\sigma_j^2} + \frac{\rho(1-\rho)}{2}\frac{\left(\mu_j - z_j^\star\right)^2}{k_n\sigma_j^2} + o\left(\frac{1}{n}\right)\\ &= -\frac{\rho}{2}\frac{\left(\mu_j - z_j^\star\right)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right)\end{aligned}$$

On the other hand,

$$\begin{aligned}\mathbb{V}_0[W_{n,j}] &= \mathbb{E}_0[W_{n,j}^2] - \left(\mathbb{E}_0[W_{n,j}]\right)^2\\ &= \mathbb{E}_0[W_{n,j}^2] + o\left(\frac{1}{n}\right)\\ &= \frac{\rho^2}{4}\,\mathbb{E}_0\left[\left(\left(\delta_{k_n,j}^{\text{out,sub}}\right)^2 - \left(\delta_{k_n,j}^{\text{in,sub}}\right)^2\right)^2\right] + o\left(\frac{1}{n}\right)\\ &= \frac{\rho^2}{k_n}\frac{(\mu_j - z_j^\star)^2}{\sigma_j^2} + o\left(\frac{1}{n}\right) + o\left(\frac{1}{n}\right)\\ &= \frac{\rho}{n}\frac{(\mu_j - z_j^\star)^2}{\sigma_j^2} + o\left(\frac{1}{n}\right)\end{aligned}$$

Under H_1:

Let $A = \{\mathbb{1}\{\varsigma(n) \leq k_n\}\}$ the event that $z^\star$ was sub-sampled. We have that $\Pr(A) = \rho$.

We have the following

$$\begin{aligned}
\mathbb{E}_1(\hat{\mu}_{n,j}^{\text{sub}}) &= \rho\,\mathbb{E}_1(\hat{\mu}_{n,j}^{\text{sub}} \mid A) + (1-\rho)\,\mathbb{E}_1(\hat{\mu}_{n,j}^{\text{sub}} \mid A^c) \\
&= \rho(\mu_j + \frac{1}{k_n}(z_j^\star - \mu_j)) + (1-\rho)\mu_j \\
&= \mu_j + \frac{1}{n}\left(z_j^\star - \mu_j\right) \\
\mathbb{V}_1\left(\hat{\mu}_{n,j}^{\text{sub}}\right) &= \mathbb{E}_1\left[\left(\hat{\mu}_{n,j}^{\text{sub}} - \mu_j - \frac{1}{n}\left(z_j^\star - \mu_j\right)\right)^2\right] \\
&= \rho\,\mathbb{E}_1\left[\left(\hat{\mu}_{n,j}^{\text{sub}} - \mu_j - \frac{1}{n}\left(z_j^\star - \mu_j\right)\right)^2 \mid A\right] + (1-\rho)\,\mathbb{E}_1\left[\left(\hat{\mu}_{n,j}^{\text{sub}} - \mu_j + \frac{1}{n}\left(z_j^\star - \mu_j\right)\right)^2 \mid A^c\right] \\
&= \rho\left(\frac{k_n - 1}{k_n^2}\sigma_j^2 + \left(\frac{1-\rho}{k_n}(z_j^\star - \mu_j)\right)^2\right) + (1-\rho)\left(\frac{1}{k_n}\sigma_j^2 + \frac{\rho^2}{k_n^2}\left(z_j^\star - \mu_j\right)^2\right) \\
&= \frac{\sigma_j^2}{k_n}\left(1 - \frac{\rho}{k_n}\right) + \frac{\rho(1-\rho)}{k_n^2}\left(z_j^\star - \mu_j\right)^2
\end{aligned}$$

Again, we compute the following expectations:

$$\begin{aligned}
\mathbb{E}_1\left(\left(\delta_{k_n,j}^{\text{out,sub}}\right)^2\right) &= \frac{k_n}{\sigma_j^2}\left(\mathbb{V}_1\left(\hat{\mu}_{n,j}^{\text{sub}}\right) + \frac{\left(\mu_j - z_j^\star\right)^2}{n^2}\right) \\
&= \frac{k_n}{\sigma_j^2}\left(\frac{\sigma_j^2}{k_n}\left(1 - \frac{\rho}{k_n}\right) + \frac{\rho(1-\rho)}{k_n^2}\left(z_j^\star - \mu_j\right)^2 + \rho^2\frac{\left(\mu_j - z_j^\star\right)^2}{k_n^2}\right) \\
&= 1 - \frac{\rho}{k_n} + \rho\frac{\left(\mu_j - z_j^\star\right)^2}{k_n\sigma_j^2}
\end{aligned}$$

and

$$\begin{aligned}
\mathbb{E}_1\left(\left(\delta_{k_n,j}^{\text{in,sub}}\right)^2\right) &= \frac{k_n^2}{(k_n-1)\sigma_j^2}\left(\mathbb{V}_1\left(\hat{\mu}_{n,j}^{\text{sub}}\right) + \left(\frac{1-\rho}{k_n}(z_j^\star - \mu_j)\right)^2\right) \\
&= \frac{k_n^2}{(k_n-1)\sigma_j^2}\left(\frac{\sigma_j^2}{k_n}\left(1 - \frac{\rho}{k_n}\right) + \frac{\rho(1-\rho)}{k_n^2}\left(z_j^\star - \mu_j\right)^2 + \left(\frac{1-\rho}{k_n}(z_j^\star - \mu_j)\right)^2\right) \\
&= 1 + \frac{1-\rho}{k_n} + (1-\rho)\frac{\left(\mu_j - z_j^\star\right)^2}{k_n\sigma_j^2} + o\left(\frac{1}{n}\right)
\end{aligned}$$

Thus again, we get

$$\mathbb{E}_1\left(\delta_{k_n,j}^{\text{out,sub}}\right) = \sqrt{\rho}\frac{z_j^\star - \mu_j}{\sigma_j\sqrt{n}} \qquad \mathbb{E}_1\left(\left(\delta_{k_n,j}^{\text{out,sub}}\right)^2\right) = 1 - \frac{1}{n} + \frac{\left(\mu_j - z_j^\star\right)^2}{n\sigma_j^2}$$

$$\mathbb{E}_1\left(\delta_{k_n,j}^{\text{in,sub}}\right) = (1-\rho)\frac{\mu_j - z_j^\star}{\sigma_j\sqrt{k_n - 1}} \qquad \mathbb{E}_1\left(\left(\delta_{k_n,j}^{\text{in,sub}}\right)^2\right) = 1 + \frac{1-\rho}{k_n} + (1-\rho)\frac{\left(\mu_j - z_j^\star\right)^2}{k_n\sigma_j^2} + o\left(\frac{1}{n}\right)$$

Now, we find that

$$\mathbb{E}_1\left(\left(\delta_{k_n,j}^{\text{out,sub}}\right)^2\right) - \mathbb{E}_1\left(\left(\delta_{k_n,j}^{\text{in,sub}}\right)^2\right) = -\frac{1}{k_n} + (2\rho - 1)\frac{\left(\mu_j - z_j^\star\right)^2}{k_n\sigma_j^2} + o\left(\frac{1}{n}\right)$$

Also,

$$\begin{aligned}\mathbb{E}_1\left(\delta^{\text{out,sub}}_{k_n,j}\delta^{\text{in,sub}}_{k_n,j}\right) &= \mathbb{E}_1\left(\sqrt{\frac{k_n}{k_n-1}}\left(\delta^{\text{out,sub}}_{k_n,j}\right)^2 + \frac{\mu_j - z^\star_j}{\sigma_j\sqrt{k_n-1}}\delta^{\text{out,sub}}_{k_n,j}\right)\\ &= \sqrt{\frac{k_n}{k_n-1}}\left(1-\frac{1}{n}+\frac{\left(\mu_j-z^\star_j\right)^2}{n\sigma_j^2}\right)+\frac{\mu_j-z^\star_j}{\sigma_j\sqrt{k_n-1}}\left(\sqrt{\rho}\frac{z^\star_j-\mu_j}{\sigma_j\sqrt{n}}\right)\\ &= 1+o(1)\end{aligned}$$

Which means that

$$\mathbb{E}_1\left(R_{k_n,j}\right) = o(1)$$

Again, following the same steps as in H_0, we arrive at

$$\mathbb{E}_1\left[\left(\left(\delta^{\text{out,sub}}_{k_n,j}\right)^2-\left(\delta^{\text{in,sub}}_{k_n,j}\right)^2\right)^2\right] = \frac{4}{k_n}\frac{(\mu_j-z^\star_j)^2}{\sigma_j^2}+o\left(\frac{1}{n}\right)$$

Finally

$$\begin{aligned}\mathbb{E}_1[W_{n,j}] &= \rho(2\rho-1)\frac{\left(\mu_j-z^\star_j\right)^2}{2k_n\sigma_j^2}+\frac{\rho(1-\rho)}{2}\frac{\left(\mu_j-z^\star_j\right)^2}{k_n\sigma_j^2}+o\left(\frac{1}{n}\right)\\ &= \rho\frac{\left(\mu_j-z^\star_j\right)^2}{2n\sigma_j^2}+o\left(\frac{1}{n}\right)\end{aligned}$$

Again,

$$\begin{aligned}\mathbb{V}_1[W_{n,j}] &= \frac{\rho^2}{4}\,\mathbb{E}_1\left[\left(\left(\delta^{\text{out,sub}}_{k_n,j}\right)^2-\left(\delta^{\text{in,sub}}_{k_n,j}\right)^2\right)^2\right]+o\left(\frac{1}{n}\right)\\ &= \frac{\rho^2}{k_n}\frac{(\mu_j-z^\star_j)^2}{\sigma_j^2}+o\left(\frac{1}{n}\right)+o\left(\frac{1}{n}\right)\\ &= \frac{\rho}{n}\frac{(\mu_j-z^\star_j)^2}{\sigma_j^2}+o\left(\frac{1}{n}\right)\end{aligned}$$

□

C.6 Effect of Misspecifiaction, Proof of Theorem 4.3

***Theorem** 4.3* (Leakage of a misspecified adversary). We show that as $d, n \leftarrow \infty$ while $d/n = \tau$,

- Under H_0, $\ell_n(\hat{\mu}_n; z^{\text{targ}}, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(-\frac{m^{\text{targ}}}{2}, m^\star\right)$.

- Under H_1, $\ell_n(\hat{\mu}_n; z^{\text{targ}}, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(\frac{m^\star - m^{\text{diff}}}{2}, m^\star\right)$

Let the adversary $\mathcal{A}_{\text{miss}}$ use $\ell_n(\hat{\mu}_n; z^{\text{targ}}, \mu, C_\sigma)$ as the LR test function. Then,

$$\lim_{n,d}\text{Adv}_n(\mathcal{A}_{\text{miss}}) = \Phi\left(\frac{|m^{\text{scal}}|}{2\sqrt{m^\star}}\right)-\Phi\left(-\frac{|m^{\text{scal}}|}{2\sqrt{m^\star}}\right).$$

Proof. **Step 1: Asymptotic expansion of the LR score**

Directly using Equation (15) from the proof in Section C.2, by only replacing $z^\star$ by z^{targ}, we get

$$\ell_n(\hat{\mu}_n; z^\star, \mu, C_\sigma) = \frac{\tau}{2}+o_p(1)+\sum_{j=1}^{d_n} Y^{\text{targ}}_{n,j}$$

where

$$Y_{n,j}^{\text{targ}} \triangleq \frac{1}{2}\left(\left(\delta_{n,j}^{\text{out,targ}}\right)^2 - \left(\delta_{n,j}^{\text{in,targ}}\right)^2\right) + \frac{\lambda_3\left(\mu_j - z_j^{\text{targ}}\right)}{n\sigma_j} R_{n,j}^{\text{targ}},$$

and

$$\begin{aligned}
\delta_{n,j}^{\text{out,targ}} &\triangleq \frac{\sqrt{n}}{\sigma_j}\left(\hat{\mu}_{n,j} - \mu_j\right) \\
\delta_{n,j}^{\text{in,targ}} &\triangleq \frac{n}{\sqrt{n-1}\sigma_j}\left(\hat{\mu}_{n,j} - \mu_j + \frac{1}{n}\left(\mu_j - z_j^{\text{targ}}\right)\right) \\
R_{n,j}^{\text{targ}} &\triangleq \left(\delta_{n,j}^{\text{out,targ}}\right)^2 + \delta_{n,j}^{\text{out,targ}}\delta_{n,j}^{\text{in,targ}} + \left(\delta_{n,j}^{\text{in,targ}}\right)^2 - 3
\end{aligned}$$

Step 2: Computing expectations and variances

This is the step where the effect of misspecification appears.

Under H_0:

Since $\mathbb{E}_0(\hat{\mu}_{n,j}) = \mu_j$ and $\mathbb{V}_0(\hat{\mu}_{n,j}) = \frac{1}{n}\sigma_j^2$, we get that

$$\begin{aligned}
&\mathbb{E}_0\left(\delta_{n,j}^{\text{out,targ}}\right) = 0 && \mathbb{E}_0\left(\left(\delta_{n,j}^{\text{out,targ}}\right)^2\right) = 1 \\
&\mathbb{E}_0\left(\delta_{n,j}^{\text{in,targ}}\right) = \frac{\mu_j - z_j^{\text{targ}}}{\sigma_j\sqrt{n-1}} && \mathbb{E}_0\left(\left(\delta_{n,j}^{\text{in,targ}}\right)^2\right) = 1 + \frac{1}{n} + \frac{\left(\mu_j - z_j^{\text{targ}}\right)^2}{n\sigma_j^2} + o\left(\frac{1}{n}\right)
\end{aligned}$$

Using similar steps as in Lemma C.9, we get

$$\mathbb{E}_0[Y_{n,j}^{\text{targ}}] = -\frac{1}{2n} - \frac{(z_j^{\text{targ}} - \mu_j)^2}{2n\sigma_j^2} + o\left(\frac{1}{n}\right)$$

And

$$\mathbb{V}_0[Y_{n,j}^{\text{targ}}] = \frac{1}{n}\frac{(\mu_j - z_j^{\text{targ}})^2}{\sigma_j^2} + o\left(\frac{1}{n}\right)$$

Under H_1:

Since $\mathbb{E}_1(\hat{\mu}_{n,j}) = \mu_j + \frac{1}{n}\left(z_j^\star - \mu_j\right)$ and $\mathbb{V}_1(\hat{\mu}_{n,j}) = \frac{n-1}{n^2}\sigma_j^2$, let

$$\delta_{n,j}^{\text{in,true}} \triangleq \frac{n}{\sqrt{n-1}\sigma_j}\left(\hat{\mu}_{n,j} - \mu_j + \frac{1}{n}\left(\mu_j - z_j^\star\right)\right),$$

and thus

$$\mathbb{E}_1\left(\delta_{n,j}^{\text{in,true}}\right) = 0, \qquad \mathbb{E}_1\left(\left(\delta_{n,j}^{\text{in,true}}\right)^2\right) = 1$$

On the other hand, we can rewrite

$$\begin{aligned}
\delta_{n,j}^{\text{out,targ}} &= \sqrt{\frac{n-1}{n}}\delta_{n,j}^{\text{in,true}} + \frac{z_j^\star - \mu_j}{\sigma_j\sqrt{n}} \\
\delta_{n,j}^{\text{in,targ}} &= \delta_{n,j}^{\text{in,true}} + \frac{z_j^\star - z_j^{\text{targ}}}{\sigma_j\sqrt{n-1}}
\end{aligned}$$

Thus, we get that

$$\mathbb{E}_1\left(\delta_{n,j}^{\text{out,targ}}\right) = \frac{z_j^\star - \mu_j}{\sigma_j\sqrt{n}} \qquad \mathbb{E}_1\left(\left(\delta_{n,j}^{\text{out,targ}}\right)^2\right) = 1 - \frac{1}{n} + \frac{\left(\mu_j - z_j^\star\right)^2}{n\sigma_j^2}$$

$$\mathbb{E}_1\left(\delta_{n,j}^{\text{in,targ}}\right) = \frac{z_j^\star - z_j^{\text{targ}}}{\sigma_j\sqrt{n-1}} \qquad \mathbb{E}_1\left(\left(\delta_{n,j}^{\text{in,targ}}\right)^2\right) = 1 + \frac{1}{(n-1)\sigma_j^2}\left(z_j^\star - z_j^{\text{targ}}\right)^2$$

Also

$$\begin{aligned}\mathbb{E}_1\left(\delta_{n,j}^{\text{out,targ}}\delta_{n,j}^{\text{in,targ}}\right) &= \sqrt{\frac{n-1}{n}} + \frac{\left(z_j^\star - \mu_j\right)\left(z_j^\star - z^{\text{targ}}\right)}{\sigma_j^2\sqrt{n}\sqrt{n-1}} \\ &= 1 - \frac{1}{2n} + \frac{\left(z_j^\star - \mu_j\right)\left(z_j^\star - z^{\text{targ}}\right)}{n\sigma_j^2} + o\left(\frac{1}{n}\right)\end{aligned}$$

Which means that

$$\mathbb{E}_1\left(R_{n,j}^{\text{targ}}\right) = o(1)$$

Finally

$$\mathbb{E}_1[Y_{n,j}^{\text{targ}}] = -\frac{1}{2n} + \frac{(z_j^\star - \mu_j)^2 - \left(z_j^\star - z_j^{\text{targ}}\right)^2}{2n\sigma_j^2} + o\left(\frac{1}{n}\right)$$

$$\mathbb{V}_1[Y_{n,j}^{\text{targ}}] = \frac{1}{n}\frac{(\mu_j - z_j^{\text{targ}})^2}{\sigma_j^2} + o\left(\frac{1}{n}\right)$$

All in all, we proved that

$$\begin{aligned}\mathbb{E}_0[Y_{n,j}^{\text{targ}}] &= -\frac{1}{2n} - \frac{(z_j^{\text{targ}} - \mu_j)^2}{2n\sigma_j^2} + o\left(\frac{1}{n}\right) \\ \mathbb{V}_0[Y_{n,j}^{\text{targ}}] &= \frac{1}{n}\frac{(\mu_j - z_j^{\text{targ}})^2}{\sigma_j^2} + o\left(\frac{1}{n}\right) \\ \mathbb{E}_1[Y_{n,j}^{\text{targ}}] &= -\frac{1}{2n} + \frac{(z_j^\star - \mu_j)^2 - \left(z_j^\star - z_j^{\text{targ}}\right)^2}{2n\sigma_j^2} + o\left(\frac{1}{n}\right) \\ \mathbb{V}_1[Y_{n,j}^{\text{targ}}] &= \frac{1}{n}\frac{(\mu_j - z_j^{\text{targ}})^2}{\sigma_j^2} + o\left(\frac{1}{n}\right)\end{aligned}$$

Step3: Concluding using the Lindeberg-Feller CLT

Under H_0:

Using the results of Step2, we have

- $\sum_{j=1}^{d_n} \mathbb{E}_0[Y_n^{\text{targ}}, j] \to -\frac{\tau}{2} - \frac{m^{\text{targ}}}{2}$
- $\sum_{j=1}^{d_n} V_0[Y_n, j] \to m^{\text{targ}}$

Similarly to Lemma C.10, we can show that $Y_{n,j}^{\text{targ}}$ verify the Lindeberg-Feller condition, i.e.

$$\sum_{j=1}^{d_n} \mathbb{E}_0\left[(Y_{n,j}^{\text{targ}})^2 \mathbb{1}\left\{|Y_{n,j}^{\text{targ}}| > \epsilon\right\}\right] \to 0$$

for every $\epsilon > 0$.

We conclude using the Lindeberg-Feller CLT (Theorem C.4) that $\sum_{j=1}^{d_n} Y_{n,j}^{\text{targ}} \rightsquigarrow \mathcal{N}\left(-\frac{\tau}{2} - \frac{m^{\text{targ}}}{2}, m^{\text{targ}}\right)$, and thus

$$\ell_n(\hat{\mu}_n; z^{\text{targ}}, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(-\frac{m^{\text{targ}}}{2}, m^{\text{targ}}\right)$$

Similarly, *Under H_1:*

We use the results of Step3

- $\sum_{j=1}^{d_n} \mathbb{E}_1[Y_{n,j}^{\text{targ}}] \to -\frac{\tau}{2} + \frac{m^\star - m^{\text{diff}}}{2}$
- $\sum_{j=1}^{d_n} V_1[Y_{n,j}^{\text{targ}}] \to m^{\text{targ}}$

Similarly to Lemma C.10, we can show that $Y_{n,j}^{\text{targ}}$ verify the Lindeberg-Feller condition, i.e.

$$\sum_{j=1}^{d_n} \mathbb{E}_1\left[(Y_{n,j}^{\text{targ}})^2 \mathbb{1}\left\{|Y_{n,j}^{\text{targ}}| > \epsilon\right\}\right] \to 0$$

for every $\epsilon > 0$.

We conclude using the Lindeberg-Feller CLT (Theorem C.4) that $\sum_{j=1}^{d_n} Y_{n,j}^{\text{targ}} \rightsquigarrow \mathcal{N}\left(-\frac{\tau}{2} + \frac{m^\star - m^{\text{diff}}}{2}, m^{\text{targ}}\right)$, and thus

$$\ell_n(\hat{\mu}_n; z^{\text{targ}}, \mu, C_\sigma) \rightsquigarrow \mathcal{N}\left(\frac{m^\star - m^{\text{diff}}}{2}, m^{\text{targ}}\right)$$

Step4: Getting the advantage of the misspecified attack.

We use that $\text{TV}\left(\mathcal{N}\left(\mu_0, \sigma_0^2\right) \,\|\, \mathcal{N}\left(\mu_1, \sigma_0^2\right)\right) = \Phi\left(\frac{|\mu_0 - \mu_1|}{2\sigma_0}\right) - \Phi\left(-\frac{|\mu_0 - \mu_1|}{2\sigma_0}\right)$, so that

$$\begin{aligned}
\lim_{n,d} \text{Adv}_n(\mathcal{A}_{\text{miss}}) &= \text{TV}\left(\mathcal{N}\left(-\frac{m^{\text{targ}}}{2}, m^{\text{targ}}\right) \,\middle\|\, \mathcal{N}\left(\frac{m^\star - m^{\text{diff}}}{2}, m^{\text{targ}}\right)\right) \\
&= \Phi\left(\frac{|m^\star + m^{\text{targ}} - m^{\text{diff}}|}{4\sqrt{m^\star}}\right) - \Phi\left(-\frac{|m^\star + m^{\text{targ}} - m^{\text{diff}}|}{4\sqrt{m^\star}}\right) \\
&= \Phi\left(\frac{|m^{\text{scal}}|}{2\sqrt{m^{\text{targ}}}}\right) - \Phi\left(-\frac{|m^{\text{scal}}|}{2\sqrt{m^{\text{targ}}}}\right)
\end{aligned}$$

because $m^{\text{diff}} = m^\star + m^{\text{targ}} - 2m^{\text{scal}}$. □

Remark C.13 (Simple way to get the expectations computation). Thanks to Remark C.7, we recall that

$$\ell_n(\hat{\mu}_n; z^{\text{targ}}, \mu, C_\sigma) \approx (z^{\text{targ}} - \mu)^T C_\sigma^{-1}(\hat{\mu}_n - \mu) - \frac{1}{2n}\|^{\text{targ}} - \mu\|_{C_\sigma^{-1}}^2$$

And thus taking the expectation under H_0 and H_1, using that $\mathbb{E}_0(\hat{\mu}_n) = \mu$, $\mathbb{E}_1(\hat{\mu}_n) = \frac{n-1}{n}\mu + \frac{1}{n}z^\star$ and $\mathbb{V}_0(\hat{\mu}_n) = \mathbb{V}_1(\hat{\mu}_n) = C_\sigma$, we get back the same expectations and variances values as the result of Theorem 4.3.

D THE WHITE-BOX FEDERATED LEARNING SETTING

First, we discuss the white-box federated learning threat model. Then, we connect our canary choosing strategy (Algorithm 3) and covariance attack (Algorithm 4) to the literature.

D.1 Threat Models of Attacks in Supervised Learning

In supervised learning, the mechanism to be attacked is a learning algorithm that takes as input a dataset D and outputs a machine learning model $o \triangleq f$. The dataset D is composed of n tuples (x_i, y_i) where x_i is a feature and y_i is a label, i.e. $D \triangleq \{(x_1, y_1), \dots, (x_n, y_n)\}$. The machine learning model f produced can then be queried for an input feature x to get a label $y = f(x)$. The model f is generally found by minimising over a class of models $\mathcal{F}$ some type of error ℓ in the input dataset D, i.e. $f \triangleq \arg\min_{g \in \mathcal{F}} \ell(g, D)$.

The class of models $\mathcal{F}$ can be parameterised by $\theta \in \mathbb{R}^d$, i.e. $f = f_\theta$. In this case, the threat model depends on whether the adversary has access to the parameter θ or only query access to the model f_θ. The setting where the adversary can observe the parameter $\theta \in \mathbb{R}^d$ is called the **white-box setting**. On the other hand, when the adversary can only query the final model f_θ, i.e. send input features x to f and observe the outputs $y = f(x)$ is called the **black-box setting**.

In the parameterised setting, the quintessential training algorithms are based on Gradient Descent. The Gradient Descent algorithm start with an initial parameter $\theta_0 \in \mathbb{R}^d$, and then updates sequentially the parameter at each step t by $\theta_t \triangleq \theta_{t-1} - \eta \nabla_{\theta_{t-1}} \ell(\theta_{t-1}, d)$. In the white-box setting, the adversary may have access to only the final parameter θ_T, and we call this setting **white-box final parameter** (Nasr et al., 2023). The adversary can have access to all (or a subset of) the intermediate parameters sequence $(\theta_0, \dots, \theta_T)$, and we call this setting **white-box federated learning** setting (Maddock et al., 2022).

Fixed-target MI game for the empirical mean mechanism can directly provide an adversary and a canary design strategy to attack/audit gradient descent algorithms in the white box federated learning setting.

Extended details on the white-box federated learning threat model. Gradient Descent algorithms start with an initial parameter $\theta_0 \in \mathbb{R}^d$, and then update sequentially the parameter at each step t by

$$\theta_t \triangleq \theta_{t-1} - \eta \nabla_{\theta_{t-1}} Q(\theta_{t-1}),$$

where η is the learning rate, and $Q(\theta_{t-1})$ is a quantity that depends on the loss on "some input samples". For example,

(a) in batch gradient descent

$$\nabla_{\theta_{t-1}} Q(\theta_{t-1}) \triangleq \frac{1}{n} \sum_{i=1}^{n} \nabla_{\theta_{t-1}} \ell(f_{\theta_{t-1}}(x_i), y_i)$$

is the gradient with respect to the whole dataset.

(b) in mini-batch gradient descent, the dataset is divided into a set of mini-batches $D = \cup B_k$. At each step t, a mini-batch B is sampled uniformly and

$$\nabla_{\theta_{t-1}} Q(\theta_{t-1}) \triangleq \frac{1}{|B|} \sum_{i \in B} \nabla_{\theta_{t-1}} \ell(f_{\theta_{t-1}}(x_i), y_i).$$

We call $|B|$ the batch size.

(c) in stochastic gradient descent,

$$\nabla_{\theta_{t-1}} Q(\theta_{t-1}) \triangleq \nabla_{\theta_{t-1}} \ell(f_{\theta_{t-1}}(x_i), y_i),$$

where $i \sim \mathcal{U}([1, n])$ is sampled randomly from $\{1, n\}$.

(d) in DP-SGD Abadi et al. (2016),

$$\nabla_{\theta_{t-1}} Q(\theta_{t-1}) \triangleq \left(\frac{1}{|B|} \sum_{i \in B} \mathrm{Clip}_C \left[\nabla_{\theta_{t-1}} \ell(f_{\theta_{t-1}}(x_i), y_i) \right] \right) + \mathcal{N}\left(0, \gamma^2 C^2 I_d\right),$$

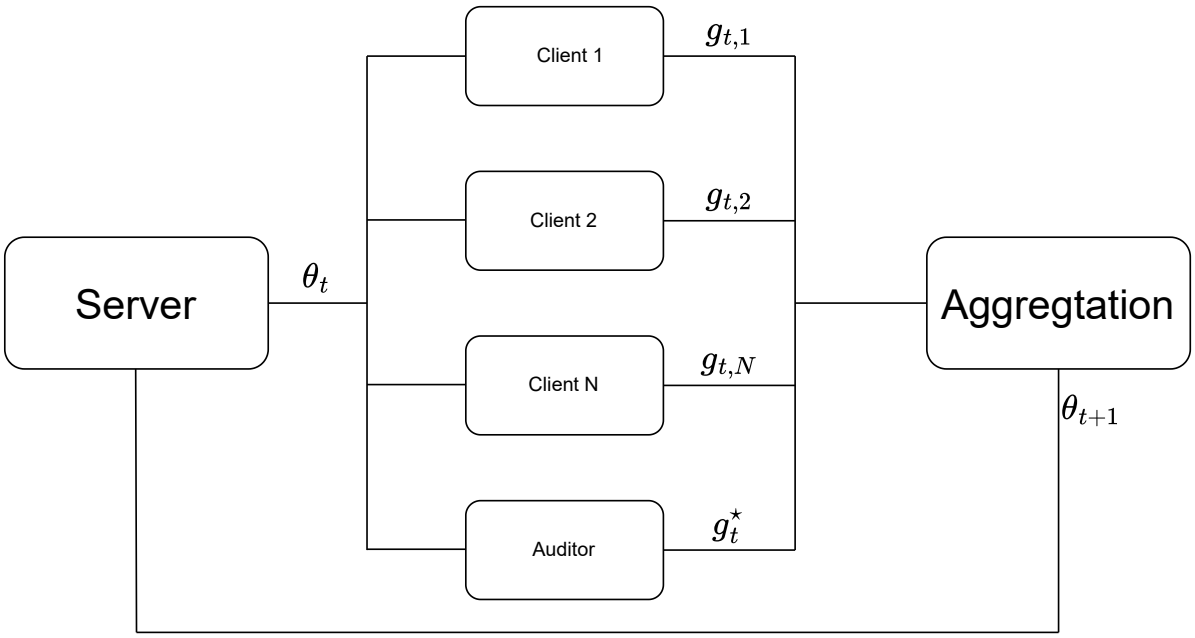


Figure 4: The White-box Federated learning threat model. At each step, the server sends a global model θ_t to each client. The adversary is a client, too. Each client i computes and sends the local update $g_{i,t}$ to the server. The adversary also computes the local update $g^\star$ on a canary $z^\star$, and sends it to the global server with probability $1/2$. The server aggregates the updates received to compute θ_{t+1} (Maddock et al., 2022).

where B is again a batch uniformly sampled, $\text{Clip}_C(x) \triangleq \min\{1, C/\|x\|\}x$ is the clipping norm function, i.e. $\text{ClipNorm}_C(x) = x$ if $\|x\| \leq C$ otherwise $\text{ClipNorm}_C(x) = C\frac{x}{\|x\|}$. Here, $C > 0$ is a gradient norm bound and $\gamma > 0$ is the noise magnitude. DP-SGD can be shown to verify a DP constraint, where the privacy budget depends on the clipping gradient norm C, the noise γ, the batch size $|B|$ and the number of gradient iterations T.

The goal is to attack gradient descent algorithms. Specifically, the mechanism to be attacked is the "training" algorithm that takes as input the private dataset $D \triangleq \{(x_i, y_i)\}_{i=1}^n$, and produces sequentially the parameter estimates $\{\theta_t\}_{t=1}^T$. In the white-box federated learning setting, the adversary can observe the full sequence of iterates $\{\theta_t\}_{t=1}^T$.

The white-box federated learning setting is a fundamental setting for understanding privacy auditing. In addition, this setting has practical uses too. For example, a potential use of this setting is for “debugging" implementations of private gradient descent algorithms. In this case, the adversary is the programmer trying to release an implementation of their favourite private descent algorithm. To verify the guarantees of their algorithm, the programmer runs a white-box federated learning attack on their implementation and compares the empirical privacy guarantees retrieved with the theoretical analysis. The white-box federated learning setting captures well this use case since the adversary *is* the programmer themselves, and thus have access to all the intermediate iterates and gradients.

On the other hand, as the name of the setting suggests, another natural application of this threat model is auditing the private Federated Learning (FL) protocol. In a standard FL protocol (Figure 4), a server computes a global model θ_t at each step t by taking the (noisy) average of client updates, where each client computes a gradient estimate using their local datasets. We suppose that the auditor is also a client in an "honest-but-curious" FL protocol, i.e. the auditor plays the role of the "adversary" but follows the rules of the protocol: At each step t of the protocol, the server sends a global model θ_t to each client. Then, each client i computes their local gradient update $g_{t,i}$ and sends it to the server. The auditor, being also a client in the protocol, computes a gradient update $g_t^\star \triangleq \nabla_{\theta_t}\ell(f_{\theta_t(x^\star)}, y^\star)$ on their local "canary" datapoint $z^\star \triangleq (x^\star, y^\star)$, i.e. the most-leaking sample. Then, the auditor sends the gradient update to the server with probability $1/2$, or otherwise sends nothing to the server. In the next step of the interaction, the auditor observes the new updated global model θ_{t+1}. The goal of the auditor

is to decide based solely on θ_{t+1}, θ_t and $g_t^\star$ whether the canary update was indeed sent to the server or not. This is exactly a tracing problem of the aggregation mechanism, which is generally a variant of the empirical mean. To design an audit in this setting, the auditor has to design two algorithms. First, an attack score to determine based on θ_{t+1}, θ_t and $g_t^\star$ whether $g_t^\star$ was included in θ_{t+1} or not, i.e. a score function that takes as input θ_{t+1}, θ_t and $g_t^\star$ and outputs a score that would be high if the canary was included, otherwise the score is low. The auditor also needs to choose a good canary $z_t^\star$ at each step t. A good canary should be an easy point to trace, i.e., one for which the target-dependent leakage is high.

To summarise, attacking a gradient descent algorithm (batch, minibatch and DP-SGD) in the white-box federated learning setting reduces to attacking variants of the empirical mean mechanism, applied to loss gradient data $\{\nabla_{\theta_{t-1}}\ell(f_{\theta_{t-1}}(x_i), y_i)\}_{i=1}^n$. Thus, our results from Section 3 and Section 4 can readily be applied.

D.2 Connection to the Existing Canary Selection Strategies

Related Works. The main intuition for canary selection strategies in the literature is to propose heuristics to generate out-of-distribution data Jagielski et al. (2020); Maddock et al. (2022); Nasr et al. (2023); Steinke et al. (2023); Andrew et al. (2023). For example, Nasr et al. (2023) proposes the Dirac canary strategy, which suggests the use of canaries gradient updates with all the values zero except at a single index. The intuition behind this choice is that the sparse nature of the Dirac gradient makes it an out-of-distribution sample in natural datasets. For CIFAR10, Nasr et al. (2023) shows the effectiveness of such a canary choice for auditing neural nets in the white-box federated learning setting. The Dirac canary is a type of *gradient canaries*, because it directly suggests what gradient $g^\star$ the auditor should include or not in the training. The other type of canaries is *input canaries*. As the name suggests, input canaries are pairs of features and label $z^\star = (x^\star, y^\star)$ that the auditor chooses to include or not in the training. There are different heuristics in the literature to generate input canaries, i.e. mislabeled examples or blank examples (Nasr et al., 2023), or adversarial examples (Jagielski et al., 2020).

On the other hand, Maddock et al. (2022) proposes CANIFE, an algorithm that learns to craft canaries by back-propagating to the input level the following loss function

$$\ell^{\text{CANIFE}}(z^\star) \triangleq \sum_i^{n_r} \langle u_i, g_t^\star\rangle^2 + \max(C - \|g_t^\star\|, 0)^2. \tag{26}$$

Here, $g_t^\star \triangleq \nabla_{\theta_t}\ell(f_{\theta_t}(x^\star), y^\star)$ is the gradient of the loss of the canary $z^\star$ at step t, u_i is the gradient of the loss with respect to a reference sample and n_r is the number of reference samples. CANIFE can be used to craft an input canary $z^\star$ by directly minimising the loss ℓ^{CANIFE}.

Connection to our work. In addition to proposing a new gradient and a new input canary strategy, our Mahalanobis score also explains the success of the heuristics presented above. Specifically, Dirac canaries, black examples, or mislabeled examples are all points with high Mahalanobis distance, thus making them great canary candidates. Our Mahalanobis score can also be run over "in-distribution" canary candidates to find the most leaking one over them. This could come in handy when the auditor, while trying to participate in the white-box audit protocol (e.g. Figure 4), does not want to hurt the accuracy of the final model. Thus, the auditor wants to send gradient updates that are helpful for accuracy, i.e. "in-distribution", but with a high enough Mahalanobis score to be distinguishable. The Mahalanobis score solves the tradeoff between the "accuracy of the model" and the "success of the MI attack" by choosing points with a moderate Mahalanobis score.

Finally, our Mahalanobis score also explains the CANIFE loss ℓ^{CANIFE} of Equation (26). In (Maddock et al., 2022, Appendix A), the intuition to explain the CANIFE loss starts by expressing the LR score between two Gaussian distributions. Then, Maddock et al. (2022) concludes that to make the two Guassians distinguishable (separable) enough, one should maximise $(g_t^\star)^T C^{-1} g_t^\star$, which is precisely the Mahalanobis score. Finally, they claim that maximising the score is "equivalent" to minimising $(g_t^\star)^T C g_t^\star$, which yields exactly the CANIFE loss ℓ^{CANIFE} when substituting $C = \sum_i u_i u_i^\top$. Thus, the Mahalanobis score also explains the CANIFE loss, and our results rigorously justify the success of this approach beyond Gaussian distributions.

D.3 Connection to White-box Scores in the Literature

Related Works. The scalar product Dwork et al. (2015) is the most popular score used in white-box attacks in the literature Maddock et al. (2022); Nasr et al. (2023); Steinke et al. (2023); Andrew et al. (2023). The scalar

product score takes as input θ_{t+1}, θ_t and $g_t^\star$ and outputs the scalar product $\langle \theta_{t+1} - \theta_t, g_t^\star \rangle$. This score is a direct application of the tracing attack against of Dwork et al. (2015) to the white-box federated learning setting, since $\theta_{t+1} - \theta_t$ is an empirical-mean like quantity.

On the other hand, Leemann et al. (2023) proposes a different score attack, called Gradient Likelihood Ratio (GLiR) Attack, i.e. Algorithm 1 in Leemann et al. (2023). This attack is based on an analysis of the LR test of the empirical mean. However, the analysis of Leemann et al. (2023) arrives at a different score function compared to our results. Specifically, let $g_{\text{batch}}^t = \frac{\theta_{t+1} - \theta_t}{\eta}$ be the batch gradient. Then, the GLiR attack needs to estimate an empirical mean $\hat{\mu}_0$ and covariance $\hat{C}_0$ of reference data's gradient. Then, the attack computes the statistics $\hat{S} = (|B| - 1)(g_{\text{batch}}^t - g_t^\star)^T \hat{C}_0^{-1} (g_{\text{batch}}^t - g_t^\star)$ and $\hat{K} = \|\hat{C}_0^{-1/2}(\hat{\mu}_0 - g_t^\star)\|$. The GLiR score is

$$s^{\text{GLiR}}(g_{\text{batch}}^t, g_t^\star) = \log\left(F^{-1}_{\chi^2_d(|B|\hat{K})}\left(\hat{S}\right)\right)$$

where $F^{-1}_{\chi^2_d(\gamma)}$ is the inverse of the CDF of the non-central chi-squared distribution with d degrees of freedom and non-centrality parameter γ and $|B|$ is the batch size. For some threshold τ, if $s^{\text{GLiR}}(g_{\text{batch}}^t, g_t^\star) < \tau$, the GLiR attack suggests that $g^\star$ was included, otherwise it was not. Next, we comment on the relation between our analysis and that of (Leemann et al., 2023).

Connection to our work. The covariance attack of Algorithm 4 is provably better than the scalar product attack, at the expense of estimating the inverse of the covariance matrix well. The shape of the covariance matrix is $d \times d$, where d is the number of parameters of the model θ. This means that storing and inverting this covariance matrix is computationally expensive for models with many parameters. A simple trick to deal with this problem is only running the attack on a subset of the parameters. For example, we can run the covariance attack over the last layer of a neural net. If the last layer has d_ℓ parameters, the covariance matrix becomes $d_\ell \times d_\ell$ with $d_\ell \ll d$.

We provide the following remarks to connect our analysis to that of Leemann et al. (2023).

(a) In Step 1 of the proof, in (Leemann et al., 2023, Section E.1.) declares that "We suppose that the number of averaged samples is sufficiently large such that we can apply the Central Limit Theorem", and thus, considers[2] under H_0,

$$\hat{\mu}_n \sim \mathcal{N}\left(\mu, \frac{1}{n} C_\sigma\right) \tag{27}$$

and under H_1,

$$\hat{\mu}_n \sim \mathcal{N}\left(\frac{1}{n} z^\star + \frac{n-1}{n}\mu, \frac{n-1}{n^2} C_\sigma\right). \tag{28}$$

However, the Central Limit Theorem is a "limit in distribution" of the empirical means. Thus, *the limit distribution of $\hat{\mu}_n$ is just the constant μ under both H_0 and H_1*. The effect of $z^\star$ disappears in this statement, as $n \to \infty$. For their claim to be "rigorously" correct, one should assume that the data-generating distribution is exactly a Gaussian distribution. This gives the exact distribution of $\hat{\mu}_n$ under H_0 and H_1 as expressed by the two equations above. Supposing that the data-generating distributions are Gaussian distributions simplifies the analysis, since now there is no need to go for asymptotics in n and d, and thus, there is no need for Edgeworth expansions and Lindeberg CLT. In contrast, our results provide a way to rigorously justify under which conditions this holistic view of "equivalence to testing between Gaussians" is correct, i.e. finite 4-th moment of the data distribution.

(b) As a score function, Leemann et al. (2023) chooses to analyse the distribution of the "norm squared" of a re-centred and normalised version of the mean i.e. $S_n \approx \|C^{-1/2}(\hat{\mu}_n - \mu)\|^2$ while hiding some constants specific to their analysis. They characterise the distribution of S_n and show that it is a (scaled) non-central chi-squared distribution with d degrees of freedom, with different parameters under H_0 and H_1. In our case, we provide the asymptotic distribution of the LR score directly under H_0 and H_1, which provides a simpler covariance score.

[2]Following equations are restatements of Equations 45 and 46 of Leemann et al. (2023) using our notations.

E EXTENDED EXPERIMENTS

We present additional experiments and results, for both synthetic and real data settings.

E.1 Experiments on Synthetic Data

We aim to validate additional theoretical results empirically. Thus, we ask

1. *Is the distribution of the LR test tightly characterised by the Gaussian distributions of Lemma C.6?*

2. *Is the asymptotic approximations for the log-likelihood ratio test suggested by our analysis tight enough?*

3. *Are the power of the LR tests tightly determined by Theorem 3.1, 4.1, and 4.2 for the empirical mean, noisy empirical mean, and sub-sampled empirical mean mechanisms, respectively?*

4. *What is the effect of n_0, the number of reference points, on the power of the empirical attack?*

Experimental setup We take $n = 1000$, $\tau = 5$ and thus $d = 5000$. The data generating distribution $\mathcal{D}$ is a d dimensional Bernoulli distribution, with parameter $p \in [0,1]^d$. To choose p, we sample it uniformly randomly from $[a, 1-a]^d$, where $a = 0.25$ to avoid limit cases. Once p is chosen, we use the same $\mathcal{D} \triangleq \text{Bern}(p)$ for all the experiments. The three mechanisms considered are $\mathcal{M}_n^{\text{emp}}$, $\mathcal{M}_n^{\gamma}$ and $\mathcal{M}_{n,\rho}^{\text{sub}}$. The adversaries chosen to each mechanism are the thresholding adversaries based on the asymptotic approximations of LR tests, suggested by the analysis. For $\mathcal{M}_n^{\text{emp}}$, we use the approximation of Eq (5). The approximate LR test for $\mathcal{M}_n^{\gamma}$ is taken by replacing C_σ by $C_\sigma + C_\gamma$ in Eq (5). For $\mathcal{M}_n^{\text{sub},\rho}$, we take the approximate log-likelihood ration from Section C.5 of the analysis. Finally, we choose target points in $\{0,1\}^d$ in three ways:

(a) The **easiest point to attack** $z^\star_{\text{easy}}$ is the point with the highest Mahalanobis distance with respect to p. It is the point with binary coordinates being the furthest away from the coordinates of p: $z^\star_{\text{easy}} = (\mathbb{1}\{p_i \leq 1/2\})_{i=1}^d$.

(b) Similarly, the **hardest point to attack** is $z^\star_{\text{hard}} = (\mathbb{1}\{p_i > 1/2\})_{i=1}^d$, which takes the binaries closest to p.

(c) A **medium point to attack** would be just a point randomly sampled from the data-generating distribution, i.e. $z^\star_{\text{med}} \sim \text{Bern}(p)$, for which the Mahalanobis distance is of order d and the leakage score of order $\tau = d/n$.

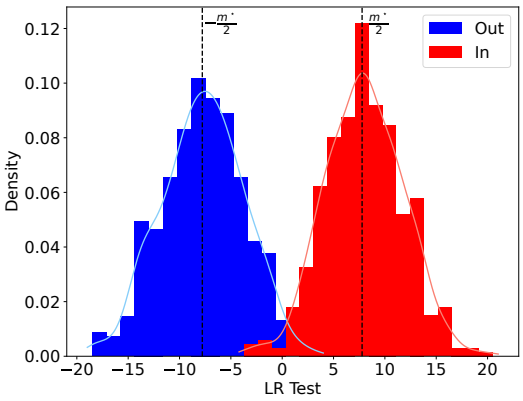
All the algorithms are implemented in Python (version 3.8) and are tested with an 8-core 64-bit Intel i5@1.6 GHz CPU. We run each fixed-target MI game 1000 times, and plot the results in Figures 5. **All the assumptions from the analysis are verified in this setting.**

Analysis of results 1. Figure 5 (a) and (b) shows the distribution of the covariance LR scores as in Eq. (5). To generate these two figures, we first simulate the crafter, i.e. Algorithm 1, for $T = 1000$ to get a list of 500 empirical means where the target was not included and 500 empirical means where the target was included. Then we compute the covariance LR score of Eq. 5 for each of the 1000 empirical means, with $\mu = p$ and $C_\sigma = \text{diag}(p(1-p))$ and $z^\star = z^\star_{\text{easy}}$ in Figure 5 (a) and $z^\star = z^\star_{\text{hard}}$ for Figure 5 (b). These two figures show that the LR score is indeed distributed as Gaussians with means $-\frac{m^\star}{2}$ and $\frac{m^\star}{2}$, and variance $m^\star$ as predicted by Theorem C.6.

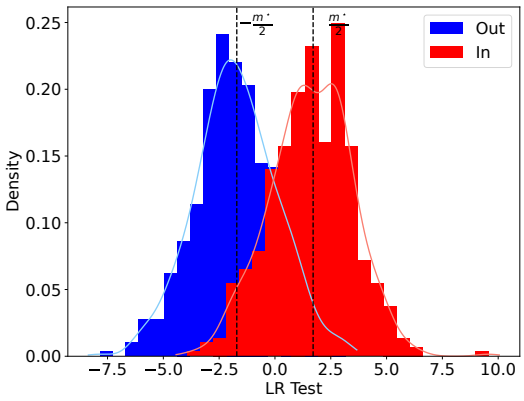
2. Figure 5 (c) compares the theoretical power function with the empirical ROC curve of the for the approximate log-likelihood ratio of Eq. 5 with $z^\star_{\text{hard}}$. The "Theoretical" dashed line in Figure 5 (c) corresponds to plotting the theoretical Pow_n of (Eq 4). The "empirical" solid line is generated by simulating the MI game of Algorithm 2 for $T = 1000$ rounds. Then by varying the thresholds of the approximate log-likelihood ratio adversary, we empirically compute the false positive and true positive rates. Eq. 5 perfectly characterises the ROC curve as predicted by Corollary.

3. Figures 5 (d), (e) and (f) are generated similarly to Figure (c), for the three mechanisms $\mathcal{M}_n^{\text{emp}}$, $\mathcal{M}_n^{\gamma}$ and $\mathcal{M}_n^{\text{sub},\rho}$. Figure 5 (d) investigates the effect of $m^\star$ by varying the target datum, where the blue line corresponds to $z^\star_{\text{easy}}$, the green line to $z^\star_{\text{med}}$ and the red line to $z^\star_{\text{hard}}$. Figure 5 (e) and (f) investigate the effect of the noise scale γ and the sub-sampling ratio ρ, by varying these parameters for the same $z^\star_{\text{easy}}$. The three figures validate the prediction of the theoretical power functions, and show that $m^\star$, $\tilde{m}^\star_\gamma$ and $\rho m^\star$ perfectly characterise the target-dependent leakage of the three mechanisms $\mathcal{M}_n^{\text{emp}}$, $\mathcal{M}_n^{\gamma}$ and $\mathcal{M}_{n,\rho}^{\text{sub}}$.

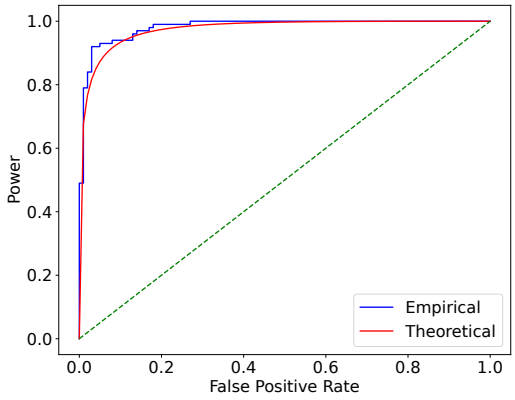
4. In Figure 6, we plot the power of the empirical covariance attack on the easy target, for different number of reference points n_0. Figure 6 shows that the power of the test decreases as n_0 gets smaller. However, this decrease is negligible even in the small n_0 regime.



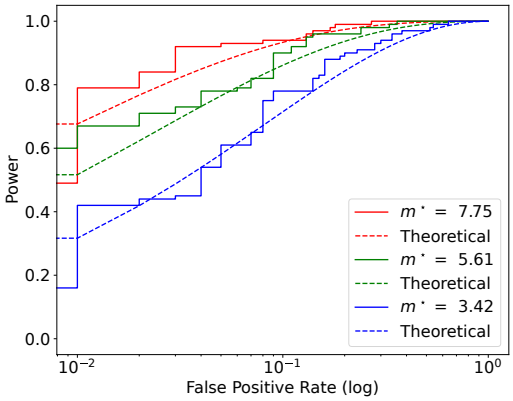
(a) LR distribution on easy target $z^{\star}_{\text{easy}}$



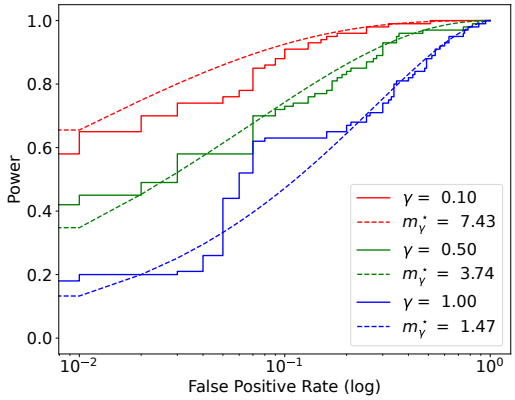
(b) LR distribution on hard target $z^{\star}_{\text{hard}}$



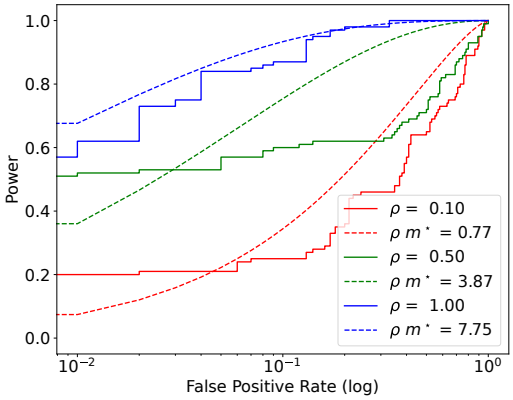
(c) Theoretical and empirical trade-offs



(d) Effect of $m^{\star}$



(e) Effect of γ

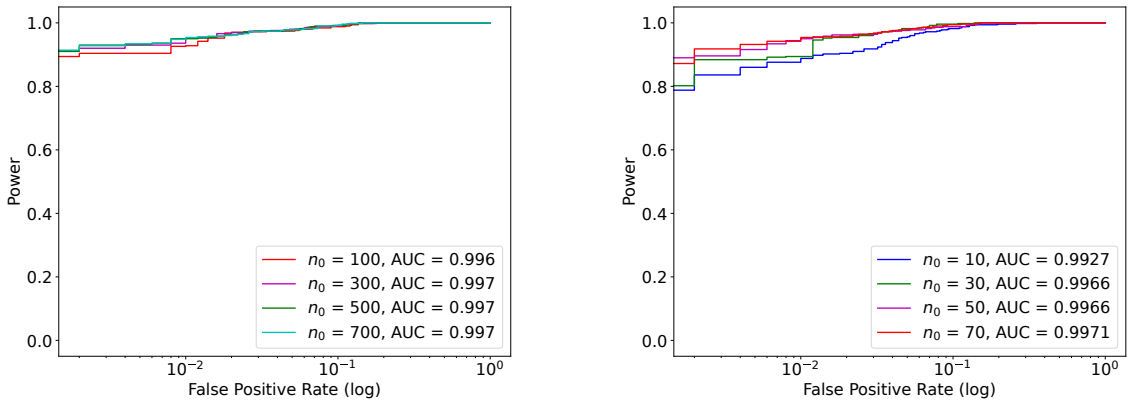


(f) Effect of ρ

Figure 5: Experimental demonstration of the theoretical results and impacts of $m^{\star}$, noise, and sub-sampling ratio on leakage. Dotted lines represent theoretical bounds and solid lines represent the empirical results.

E.2 Attacking in the White-box Federated Learning Setting

All the attacks are implemented in Python (version 3.8) and are tested with an NVIDIA GeForce RTX 2080 Ti GPU. We run each attack 100 times, and plot the results in Figures 3. We train our neural network models using



(a) Large n_0 regime

(b) Small n_0 regime

Figure 6: Effect of the number of reference points n_0 on the power of the covariance attack.

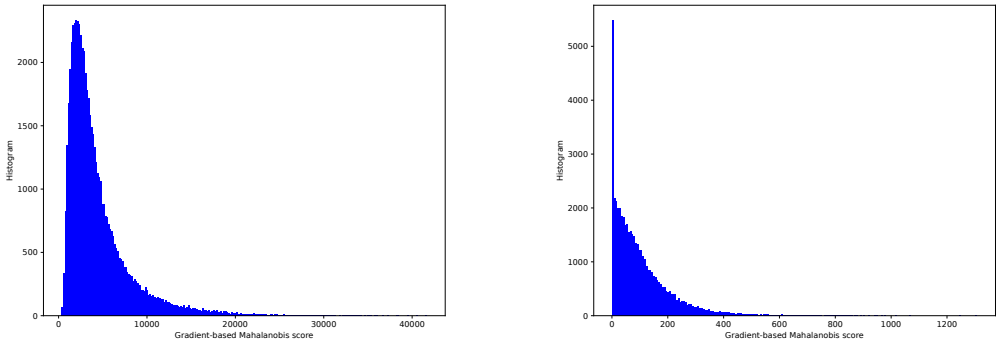
PyTorch (version 2.3.0) (Paszke et al., 2019).

In addition to the results stated in Section 6, here we add the a few additional results for this setting.

We plot the distribution of the Mahalanobis score, for FMNIST and CIFAR10 in Figure 7. We also plot the easiest, medium and hardest points to attack z_{easy}, z_{med} and z_{hard} for FMNIST and CIFAR10 in Figure 8. Interestingly, the easy instances from both datasets are the ones that are hard to detect as real objects, i.e. they incur a high loss for an ML model. In contrast, the hard ones are like any other well-classifiable data point in the dataset. This echoes the practical results in MI attack literature (Carlini et al., 2022a) from the black-box settings.

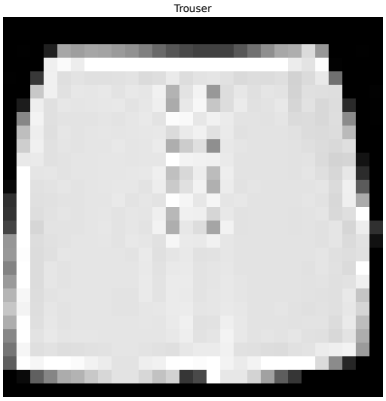
Relation to the theoretical assumptions in the analysis. For this setting, the assumptions from our analysis are not satisfied. Specifically, the gradient loss distribution, which is the data-generating distribution in this setting, does not satisfy two assumptions from Section 3: finite 4-th moment and independence of the components. It is hard to say anything about the distribution of the gradient loss on natural data. However, our results are a step in the right direction: we don't assume that the gradients are exactly distributed as Gaussians Maddock et al. (2022) or Bernoulli Sankararaman et al. (2009). We leave it for future work to weaken the assumptions on the data-generating distribution. We provide the following details regarding the importance of the "independence" assumption.

Our analysis assumes that the data-generating distribution $\mathcal{D}$ is a product distribution, i.e. the columns of the input are independent. This assumption is standard and has been used in different related works in the tracing literature (Homer et al., 2008; Sankararaman et al., 2009; Dwork et al., 2015). Our proof could be adapted to the dependent case using a *multivariate* Edgeworth expansion in the likelihood ratio test. The same conclusions of our analysis will follow, with the only difference being that the covariance matrix will no longer be diagonal but a "full" matrix. Additional technical assumptions must be added to rigorously use a high-dimensional *multivariate* Edgeworth expansion, making the analysis very technical without yielding additional insights. We leave it as a future direction to adapt the proof to the dependent case.

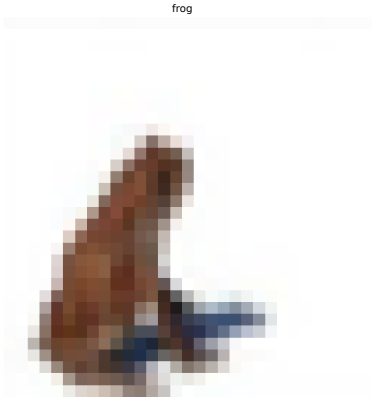


(a) FMNIST (b) CIFAR10

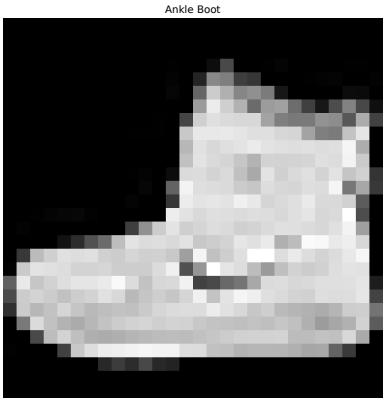
Figure 7: Distribution of the Mahalanobis score.



(a) z_{easy} for FMNIST



(b) z_{easy} for CIFAR10



(c) z_{med} for FMNIST

(d) z_{med} for CIFAR

(e) z_{hard} for FMNIST

(f) z_{hard} for CIFAR10

Figure 8: The easiest, medium and hardest points to attack z_{easy}, z_{med} and z_{hard} for FMNIST and CIFAR10

F BROADER IMPACT

The goal of this paper is to advance the field of Trustworthy Machine Learning. Our work theoretically quantifies the privacy leakage due to publicly releasing simple statistics, such as the empirical mean of a dataset. Our results are also generalised to ML models. We believe that this work is a step forward giving the analytical tools to quantify the privacy leakage due to participation or not in a dataset. Thus, this provides more informed decisions to the individual on how much their participation may impact the published statistics/models. We point out that our work only studies theoretically the likelihood ratio attacks, and our theoretical tracing attacks to dust for the presence of target individuals were only used in simulated data and public datasets (FMNIST, CIFAR10) in the experiments.