CRYPTANALYTIC EXTRACTION OF DIRECT-CODED LIF SPIKING NEURAL NETWORKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Spiking neural networks (SNNs) are increasingly deployed on edge and neuromorphic platforms, making their parameters and temporal dynamics valuable intellectual property. Cryptanalytic model extraction has focused mainly on conventional neural networks, leaving structural leakage from discrete firing, membrane recurrence, and reset dynamics in LIF-SNNs underexplored. We study cryptanalytic extraction of direct-coded LIF-SNNs through a black-box interface exposing only final logits. We show that LIF dynamics create structured input-space boundaries corresponding to spike-count transitions; crossing an isolated boundary associated with a hidden neuron produces an observable logit jump whose direction is determined by that neuron's output-weight vector, thereby exposing internal parameter information. Exploiting this observation, our attack performs transition attribution, boundary recovery, and dynamical parameter inversion to recover normalized hidden-layer parameters, the shared leak parameter, and output-layer parameters. The resulting canonical parameterization is equivalent to that of the target model and defines a functionally equivalent model. We further establish sufficient conditions for identifying the canonical parameters. Experiments show complete hidden-neuron recovery, over 99% spike-count fidelity, and near-perfect prediction agreement on natural inputs. This work presents the first systematic study of cryptanalytic extraction for direct-coded LIF-SNNs and demonstrates the need for defenses tailored to SNN model extraction.

1 INTRODUCTION

Spiking neural networks (SNNs) transmit information through discrete spikes and retain temporal information in neuronal states, making them suitable for event-driven computation (Maass, 1997). Neuromorphic processors such as Loihi exploit sparse neural activity to enable energy-efficient inference (Davies et al., 2018; Roy et al., 2019). As SNNs are increasingly deployed on edge devices and neuromorphic platforms, their synaptic weights, neuronal parameters, and temporal dynamics constitute valuable intellectual property.

Black-box model extraction attacks threaten these assets through prediction interfaces. Surrogate-based methods train local models using queried labels or scores (Tramèr et al., 2016; Jagielski et al., 2020), whereas cryptanalytic methods infer internal parameters from algebraic constraints and activation-boundary geometry (Carlini et al., 2020; Canales-Martínez et al., 2024). Existing SNN security research has mainly focused on adversarial robustness (Bu et al., 2023). To the best of our knowledge, recovering the hidden parameters and temporal dynamics of LIF-SNNs from final logit vectors alone has not been systematically studied.

This work investigates cryptanalytic extraction of direct-coded LIF-SNNs. The attacker can adaptively construct inputs and obtain the final logit vector, with no access to hidden spikes, membrane-potential trajectories, or other intermediate states. Recovery under this restricted interface faces three main challenges: internal states are not directly observable; input-space boundary geometry is coupled with LIF temporal dynamics; and recovery is affected by boundary-normal sign ambiguity, neuron-wise scale equivalence, and hidden-neuron permutation ambiguity.

Our key finding is that LIF dynamics map the time-invariant drive under direct coding to a staircase-shaped spike-count response. Spike-count transitions form hyperplanes in the input space, and transitions associated with the same hidden neuron define a family of parallel boundaries. Their

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

normals and relative positions carry information about the normalized input-weight direction and the affine parameters and leak dynamics, respectively. Unlike static activation-boundary extraction, the LIF setting additionally exposes integer jump multiplicities and leak-dependent inter-boundary spacing: jump directions support neuron attribution, multiplicities reveal merged count transitions, and spacing constrains the temporal leak.

Based on this structure, the proposed method locates spike-count transitions along constructed input paths and attributes them according to their logit-jump directions. It then recovers boundary normals through symmetric perturbations, collects boundary sequences through directional scans, and combines the recovered geometry with theoretical count thresholds to jointly recover the normalized hidden-layer parameters and the shared leak parameter. After reconstructing the hidden spike-count responses, the output-layer parameters are obtained by solving a linear system, yielding a complete extracted model.

The contributions of this work are summarized as follows:

- We formalize black-box extraction of direct-coded LIF-SNNs under an interface exposing only final logit vectors, and characterize the recovery ambiguities induced by boundary-normal signs, neuron-wise scaling, and hidden-neuron permutations.
- We establish the structural relationship between spike-count boundaries and output jumps, and derive sufficient conditions for identifying the normalized hidden-layer parameters, the shared leak parameter, and the output-layer parameters.
- We develop a complete cryptanalytic extraction method and evaluate its recovery performance across different hidden widths, random seeds, simulation horizons, and leak parameters. Ablation experiments further examine the roles of its key components.

2 Related Work

Surrogate-Based Model Extraction. Surrogate attacks train local models from queried labels or scores, with representative approaches based on substitute-model expansion, public-image querying, and query-efficient sample synthesis (Papernot et al., 2017; Orekondy et al., 2019; Yuan et al., 2022). Their primary objective is behavioral reproduction; fidelity measures agreement with the target and does not generally imply neuron-level correspondence or parameter recovery (Jagielski et al., 2020).

Cryptanalytic Model Extraction. Cryptanalytic attacks infer parameters from algebraic constraints and activation-boundary geometry. Prior work extracted classical models from prediction scores (Tramèr et al., 2016) and recovered deep ReLU networks from critical points, local output variation, and layerwise boundary structure (Rolnick & Kording, 2020; Jagielski et al., 2020; Carlini et al., 2020; Canales-Martínez et al., 2024). Extensions cover hard-label interfaces, broader activation families, CNNs, RNNs, and message-passing GNNs (Chen et al., 2025; Carlini et al., 2025; Chen et al., 2026; Qi et al., 2026; Sun et al., 2026; Wei et al., 2026; Li et al., 2026). Direct-coded LIF-SNNs add recurrent membrane dynamics, reset, and discrete spike-count transitions. The proposed recovery specifically exploits parallel count-boundary families, integer jump multiplicities, and leak-dependent boundary spacing to recover temporal dynamics in addition to static parameters.

Spiking Model Identification. Spiking-neuron identification has mainly assumed direct observations such as spike trains, injected currents, or membrane-potential trajectories (Paninski et al., 2004; Mullowney & Iyengar, 2008; Rossant et al., 2011; Pozzorini et al., 2015; Burghi et al., 2020). Our setting is more restrictive: the attacker selects inputs and observes only the final logit vector of the complete network.

3 Preliminaries

3.1 Target Model

We consider a single-hidden-layer direct-coded LIF-SNN with input dimension d, hidden width h, and output dimension c. A static input $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ produces a drive repeated over T timesteps (Kim

et al., 2022). For hidden neuron j, with shared leak $\lambda \in (0,1)$ and threshold $\theta_j > 0$,

$$\widetilde{u}_{j,t} = \lambda u_{j,t-1} + (1-\lambda)\big(\mathbf{w}_j^\top \mathbf{x} + b_j\big),$$
$$s_{j,t} = \mathbb{I}[\widetilde{u}_{j,t} \geq \theta_j], \qquad u_{j,t} = \widetilde{u}_{j,t} - \theta_j s_{j,t}, \tag{1}$$

where $u_{j,0} = 0$. The accumulated hidden response is

$$R_j(\mathbf{x}) = \sum_{t=1}^{T} s_{j,t}(\mathbf{x}), \qquad \mathbf{R}(\mathbf{x}) = [R_1(\mathbf{x}), \ldots, R_h(\mathbf{x})]^\top. \tag{2}$$

With output parameters $\mathbf{W}^{(2)} \in \mathbb{R}^{c\times h}$ and $\mathbf{b}^{(2)} \in \mathbb{R}^c$, the returned logit vector is

$$\mathbf{z}(\mathbf{x}) = \sum_{t=1}^{T}\big(\mathbf{W}^{(2)}\mathbf{s}_t + \mathbf{b}^{(2)}\big) = \mathbf{W}^{(2)}\mathbf{R}(\mathbf{x}) + T\mathbf{b}^{(2)}. \tag{3}$$

For a fixed spike sequence the membrane trajectory is affine in the input; threshold crossings change the sequence and yield a piecewise input–output mapping.

To remove threshold-induced scale redundancy, define $\overline{\mathbf{w}}_j = \mathbf{w}_j/\theta_j$, $\overline{b}_j = b_j/\theta_j$, and $q_j(\mathbf{x}) = \overline{\mathbf{w}}_j^\top \mathbf{x} + \overline{b}_j$. For fixed λ and T,

$$R_j(\mathbf{x}) = C_{\lambda,T}(q_j(\mathbf{x})), \tag{4}$$

where $C_{\lambda,T}$ is the LIF spike-count function under a constant normalized drive.

3.2 Threat Model

The attacker knows the architecture (d, h, c), inference horizon T, direct-coding scheme, LIF update rule, and output accumulation rule. The analysis assumes a single hidden layer, deterministic soft reset, a shared leak, a continuous bounded query domain, and unquantized final logits. Unknowns are $\{\mathbf{w}_j, b_j, \theta_j\}_{j=1}^h$, λ, and $\mathbf{W}^{(2)}, \mathbf{b}^{(2)}$. The attacker adaptively chooses $\mathbf{x} \in \mathcal{X}$ and observes only $\mathbf{z}(\mathbf{x})$; hidden spikes, membrane trajectories, neuron-wise counts, and internal-layer queries are unavailable. Nonzero pairwise non-proportional output columns are required when jump directions are used for neuron attribution.

3.3 Recovery Objective and Equivalence

The hidden parameters exhibit neuron-wise positive-scaling equivalence. For any $\alpha_j > 0$,

$$(\mathbf{w}_j, b_j, \theta_j) \longmapsto (\alpha_j \mathbf{w}_j, \alpha_j b_j, \alpha_j \theta_j), \tag{5}$$

which preserves $q_j(\mathbf{x})$ and the spike sequence. We therefore use $(\overline{\mathbf{w}}_j, \overline{b}_j)$ as the canonical representation of neuron j. Hidden neurons also admit permutation equivalence: for any permutation matrix $\mathbf{P}$,

$$\big(\overline{\mathbf{W}}^{(1)}, \overline{\mathbf{b}}^{(1)}, \mathbf{W}^{(2)}\big) \mapsto \big(\mathbf{P}\overline{\mathbf{W}}^{(1)}, \mathbf{P}\overline{\mathbf{b}}^{(1)}, \mathbf{W}^{(2)}\mathbf{P}^\top\big),$$

which only reindexes hidden neurons.

Our objective is to recover $\mathcal{P}_{\text{can}} = (\{\overline{\mathbf{w}}_j, \overline{b}_j\}_{j=1}^h, \lambda, \mathbf{W}^{(2)}, \mathbf{b}^{(2)})$ up to hidden-neuron permutation. An extracted model is functionally equivalent when $\widehat{\mathbf{z}}(\mathbf{x}) = \mathbf{z}(\mathbf{x})$ for every $\mathbf{x} \in \mathcal{X}$. The boundary-normal sign ambiguity is resolved later using the direction of increasing spike count.

4 Theoretical Foundation

By Eq. (4), neuron j is governed by its normalized drive $q_j(\mathbf{x}) = \overline{\mathbf{w}}_j^\top \mathbf{x} + \overline{b}_j$ and the shared leak λ. Under constant drive, spike-count transitions induce input-space boundaries whose normals, locations, and logit jumps encode the canonical static parameters and temporal dynamics. We hence analyze $(\overline{\mathbf{w}}_j, \overline{b}_j)$, the positive-scaling equivalence class induced by Eq. (5).

4.1 Static Parameter Identifiability

For fixed λ and T, the spike-count function $C_{\lambda,T}(q)$ is nondecreasing in the constant drive q. For $r \in \{1, \dots, T\}$, define the rth count-transition threshold and express the count function as

$$\begin{aligned} \rho_r(\lambda, T) &= \inf\{q : C_{\lambda,T}(q) \geq r\}, \\ C_{\lambda,T}(q) &= \sum_{r=1}^{T} \mathbb{I}[q \geq \rho_r(\lambda, T)]. \end{aligned} \tag{6}$$

Several transition thresholds may coincide, producing a composite count change at the same drive value.

For hidden neuron j, transition threshold $\rho_r(\lambda, T)$ induces the input-space hyperplane

$$\mathcal{H}_{j,r} = \{\mathbf{x} \in \mathcal{X} : \overline{\mathbf{w}}_j^\top \mathbf{x} + \overline{b}_j = \rho_r(\lambda, T)\}. \tag{7}$$

Hence, each hidden neuron generates a family of parallel boundaries with common normal $\overline{\mathbf{w}}_j$. When an accessible local boundary segment lies inside the input domain, its geometry determines $\operatorname{span}(\overline{\mathbf{w}}_j)$ and thereby identifies the normalized weight direction up to sign.

Consider an isolated boundary crossing at which only the spike count of hidden neuron j changes locally. For inputs $\mathbf{x}^-$ and $\mathbf{x}^+$ on opposite sides of the boundary, Eq. (3) gives

$$\begin{aligned} \Delta\mathbf{z} = \mathbf{z}(\mathbf{x}^+) - \mathbf{z}(\mathbf{x}^-) &= \kappa \mathbf{W}^{(2)}_{:,j}, \\ \kappa = R_j(\mathbf{x}^+) - R_j(\mathbf{x}^-) &\neq 0. \end{aligned} \tag{8}$$

Logit jumps generated by the same hidden neuron lie in a common one-dimensional subspace, with magnitudes scaled by the integer count change κ. If the output-weight columns are nonzero and pairwise non-proportional, the jump direction uniquely attributes each boundary event to a hidden neuron, up to hidden-neuron permutation. If every hidden neuron has at least one observable unit-count transition, the integer multiplicities of its remaining boundaries are determined by jump-magnitude ratios within the same group.

Let $\widehat{\mathbf{n}}_j$ denote either orientation of the unit boundary normal, and consider the line $\mathbf{x}_j(\alpha) = \mathbf{x}_{j,0} + \alpha\widehat{\mathbf{n}}_j$ through a base point $\mathbf{x}_{j,0}$. The normalized drive is affine along this line. If boundary location $\alpha_{j,\ell}$ corresponds to theoretical transition index $r_{j,\ell}$, then

$$\begin{aligned} \rho_{r_{j,\ell}}(\lambda, T) &= s_j \alpha_{j,\ell} + c_j, \\ s_j = \overline{\mathbf{w}}_j^\top \widehat{\mathbf{n}}_j, \quad c_j &= \overline{\mathbf{w}}_j^\top \mathbf{x}_{j,0} + \overline{b}_j. \end{aligned} \tag{9}$$

Given λ and the correct transition indices, two boundaries associated with distinct theoretical thresholds uniquely determine s_j and c_j. The canonical static parameters then follow from

$$\overline{\mathbf{w}}_j = s_j \widehat{\mathbf{n}}_j, \qquad \overline{b}_j = c_j - \overline{\mathbf{w}}_j^\top \mathbf{x}_{j,0}. \tag{10}$$

Theorem 1 *Assume that every hidden neuron has at least two observable isolated count transitions associated with distinct theoretical thresholds, the output-weight columns are nonzero and pairwise non-proportional, and the shared leak parameter λ and transition-index correspondence are known. Then the final-logit function uniquely determines the canonical static parameters $(\overline{\mathbf{w}}_j, \overline{b}_j)$ of every hidden neuron, up to hidden-neuron permutation. Equivalently, the positive-scaling equivalence class of $(\mathbf{w}_j, b_j, \theta_j)$ is identifiable.*

4.2 Dynamical Parameter Identifiability

For a known inference horizon T, the theoretical transition-threshold sequence $\{\rho_r(\lambda, T)\}_{r=1}^{T}$ is determined by the shared leak parameter λ. Equation (9) shows that the observed boundary sequence of each hidden neuron is a one-dimensional affine image of this theoretical sequence, with neuron-specific slope and intercept.

When several theoretical thresholds coincide, one observed boundary represents multiple count transitions. The integer multiplicity in Eq. (8) unfolds such a composite boundary into the corresponding number of transitions, establishing an ordered correspondence between observed boundaries and

theoretical transition indices. Reversing the chosen normal orientation only changes the sign of s_j and preserves the affine structure of the sequence.

For three transition indices $r < s < t$ associated with distinct theoretical thresholds, define the threshold-spacing ratio. By Eq. (9), the theoretical thresholds and observed boundary locations satisfy

$$G_{r,s,t}(\lambda) = \frac{\rho_s(\lambda, T) - \rho_r(\lambda, T)}{\rho_t(\lambda, T) - \rho_r(\lambda, T)} = \frac{\alpha_{j,s} - \alpha_{j,r}}{\alpha_{j,t} - \alpha_{j,r}}. \tag{11}$$

This ratio eliminates the neuron-specific slope and intercept and remains unchanged under reversal of the normal orientation. It therefore depends only on λ, T, and the transition indices. If a collection of such spacing ratios uniquely corresponds to one leak value within an admissible interval $\Lambda \subset (0, 1)$, two distinct candidates for λ cannot induce the same affine threshold structure. Since all hidden neurons share the same leak parameter, their boundary families provide consistent constraints on a common λ.

Theorem 2 *Assume that boundary events have been correctly attributed to hidden neurons, the integer multiplicities of composite transitions are identifiable, the observed boundaries and theoretical transition indices admit a unique unfolding and alignment, and sufficiently many distinct theoretical thresholds are available. If the corresponding threshold-spacing ratios uniquely determine λ over the admissible interval Λ, then the shared leak parameter is uniquely identifiable from the boundary sequences induced by the final-logit function.*

Theorem 2 is a sufficient-condition result: it does not assert that the spacing map is globally injective for arbitrary T, transition subsets, or leak values. If multiple leak candidates admit indistinguishable aligned threshold patterns, uniqueness is not guaranteed. Under the assumptions of Theorems 1 and 2, the recovered leak parameter and transition alignment determine the canonical static parameters, up to hidden-neuron permutation.

5 Extraction Method

Figure 1 summarizes the complete pipeline from final-logit queries to hidden- and output-layer recovery.

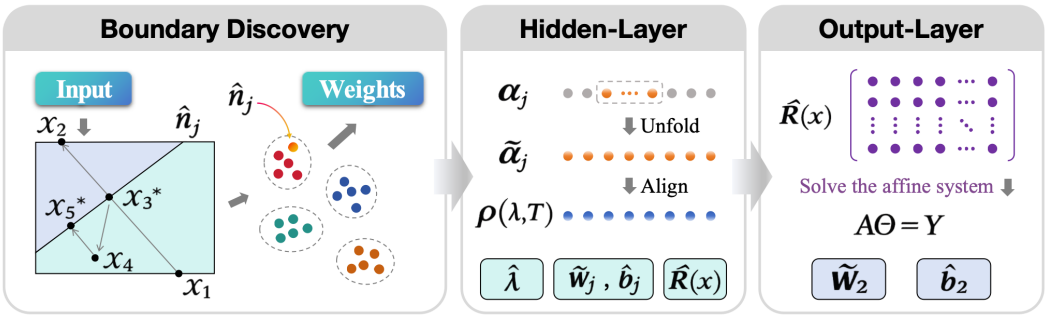


Figure 1: Overview of the proposed attack.

5.1 Boundary Discovery

The attacker scans feasible lines $\mathbf{x}(\alpha) = \mathbf{x}_0 + \alpha\mathbf{d}$ with $\|\mathbf{d}\|_2 = 1$. A nonzero output difference brackets a spike-count transition, which binary search refines to $\mathbf{x}^\star$; local probes yield

$$\widehat{\Delta\mathbf{z}} = \mathbf{z}(\mathbf{x}^\star + \eta\mathbf{d}) - \mathbf{z}(\mathbf{x}^\star - \eta\mathbf{d}). \tag{12}$$

The probe scale η is reduced until the jump direction is stable. Events with unresolved neighboring transitions or unstable directions are discarded.

By Eq. (8), isolated transitions from neuron j lie in $\mathrm{span}(\mathbf{W}^{(2)}_{:,j})$. We group events by absolute cosine similarity of jump directions; under nonzero, pairwise non-proportional output columns, the groups $\{\mathcal{B}_j\}_{j=1}^h$ correspond to distinct hidden neurons. Sign-aligned jumps yield the unit direction $\widehat{\mathbf{g}}_j$.

To recover boundary normals, coordinate k is perturbed by $\pm\epsilon\mathbf{e}_k$ and the same boundary is relocated along $\mathbf{d}$. If the resulting displacements are δ_k^+ and δ_k^-, then

$$-\frac{\delta_k^+ - \delta_k^-}{2\epsilon} = \frac{\overline{w}_{j,k}}{\overline{\mathbf{w}}_j^\top \mathbf{d}}. \tag{13}$$

Each relocation pair reveals one canonical normal component up to a common scale.

Let $\mathbf{v}(\xi)$ collect the ratios from Eq. (13) for event ξ. Sign-aligned estimates are aggregated by the leading right singular vector, refined using additional same-family transitions, and normalized to $\widehat{\mathbf{n}}_j$. A final scan along $\mathbf{x}_{j,0} + \alpha\widehat{\mathbf{n}}_j$ records ordered locations $\boldsymbol{\alpha}_j$ and signed jump projections $\boldsymbol{\gamma}_j$. For an event set $\mathcal{B}$, $\mathbf{V}(\mathcal{B}) = [\mathbf{v}(\xi)^\top]_{\xi\in\mathcal{B}}$, and $\mathrm{svd}_1(\cdot)$ denotes its unit leading right singular vector.

Algorithm 1 Boundary-Family Discovery

Require: Oracle $\mathbf{z}(\cdot)$, domain $\mathcal{X}$, hidden width h
Ensure: Estimates $\{\widehat{\mathcal{F}}_j\}_{j=1}^h$

$\mathcal{E} \leftarrow \mathsf{Locate}(\mathbf{z}, \mathcal{X})$
$\{(\mathcal{B}_j, \widehat{\mathbf{g}}_j)\}_{j=1}^h \leftarrow \mathsf{Group}(\mathcal{E}, h)$
for $j = 1, \ldots, h$ **do**
  $\widehat{\mathbf{n}}_j^{(0)} \leftarrow \mathrm{svd}_1(\mathbf{V}(\mathcal{B}_j))$
  $\mathcal{B}'_j \leftarrow \mathsf{Collect}(\mathbf{z}, \widehat{\mathbf{n}}_j^{(0)}, \widehat{\mathbf{g}}_j)$
  $\widehat{\mathbf{n}}_j \leftarrow \mathrm{svd}_1(\mathbf{V}(\mathcal{B}'_j))$
  $(\mathbf{x}_{j,0}, \boldsymbol{\alpha}_j, \boldsymbol{\gamma}_j) \leftarrow \mathsf{Scan}(\mathbf{z}, \widehat{\mathbf{n}}_j, \widehat{\mathbf{g}}_j)$
  $\widehat{\mathcal{F}}_j \leftarrow (\widehat{\mathbf{g}}_j, \widehat{\mathbf{n}}_j, \mathbf{x}_{j,0}, \boldsymbol{\alpha}_j, \boldsymbol{\gamma}_j)$
end for
return $\{\widehat{\mathcal{F}}_j\}_{j=1}^h$

Here Locate detects and refines transitions; Group, Collect, and Scan implement attribution, normal refinement, and ordered boundary scanning.

5.2 Hidden-Layer Recovery

A geometric boundary may merge several spike-count transitions. By Eq. (8), its jump is an integer multiple of one output-weight column. Assuming one observable unit-count transition for neuron j, define

$$\gamma_j^0 = \min_{m:\,|\gamma_{j,m}|>0} |\gamma_{j,m}|, \qquad \widehat{\kappa}_{j,m} = \mathrm{round}\left(\frac{|\gamma_{j,m}|}{\gamma_j^0}\right).$$

Repeating each $\alpha_{j,m}$ exactly $\widehat{\kappa}_{j,m}$ times produces the unfolded sequence $\widetilde{\boldsymbol{\alpha}}_j$ and restores transition indices hidden by coincident boundaries. This normalization requires at least one observed unit-count transition in each family. If all observed jumps share an unknown common multiplicity greater than one, absolute multiplicities are identifiable only up to that factor and the later alignment has no uniqueness guarantee.

For candidate $\lambda \in \Lambda$, the LIF recurrence generates $\boldsymbol{\rho}^\lambda = (\rho_1(\lambda, T), \ldots, \rho_T(\lambda, T))$. Each unfolded sequence is matched to an ordered subsequence through $q = (\omega, \pi, s, c)$, where $\pi = (\pi_1 < \cdots < \pi_{L_j}) \subseteq [T]$ and ω, s, and c encode orientation, affine scale, and offset. With $L_j = |\widetilde{\boldsymbol{\alpha}}_j|$,

$$\begin{aligned} E_j(\lambda) &= \min_{q\in\mathcal{Q}_j(\lambda)} \frac{\|\boldsymbol{\rho}^\lambda_\pi - s\widetilde{\boldsymbol{\alpha}}_j^{\,\omega} - c\mathbf{1}\|_2^2}{L_j}, \\ \widehat{\lambda} &= \arg\min_{\lambda\in\Lambda} \mathrm{med}_{j\in[h]}\, E_j(\lambda). \end{aligned} \tag{14}$$

For fixed (ω, π), (s, c) follows from linear least squares; median aggregation reduces sensitivity to incomplete scans and localization error.

At $\widehat{\lambda}$, $\mathcal{C}_j(\widehat{\lambda})$ retains assignments within tolerance τ_{a} of the minimum residual. Candidate tuples $\mathbf{q} = (q_1, \ldots, q_h)$ define hidden parameters and count columns and are ranked by prediction agreement and logit consistency through $\mathcal{L}_{\mathrm{bb}}$.

Algorithm 2 Hidden-Layer Recovery

Require: $\{(\boldsymbol{\alpha}_j, \boldsymbol{\gamma}_j, \widehat{\mathbf{n}}_j, \mathbf{x}_{j,0})\}_{j=1}^{h}$, $\mathbf{z}(\cdot)$, T, Λ

Ensure: $\widehat{\lambda}$, $\{(\widehat{\overline{\mathbf{w}}}_j, \widehat{\overline{b}}_j)\}_{j=1}^{h}$

for $j = 1, \ldots, h$ **do**
  $\widetilde{\boldsymbol{\alpha}}_j \leftarrow \mathsf{Unfold}(\boldsymbol{\alpha}_j, \boldsymbol{\gamma}_j)$
end for
$\{E_j(\lambda), \mathcal{C}_j(\lambda)\} \leftarrow \mathsf{Align}(\{\widetilde{\boldsymbol{\alpha}}_j\}_{j=1}^{h}, T, \Lambda)$
$\widehat{\lambda} \leftarrow \arg\min_{\lambda \in \Lambda} \mathrm{med}_{j \in [h]}\, E_j(\lambda)$
$\widehat{\mathbf{q}} \leftarrow \mathsf{Select}\big(\mathbf{z}, \{\mathcal{C}_j(\widehat{\lambda})\}_{j=1}^{h}\big)$
for $j = 1, \ldots, h$ **do**
  $(\widehat{\omega}_j, \widehat{\pi}_j, \widehat{s}_j, \widehat{c}_j) \leftarrow \widehat{q}_j$
  $\widehat{\overline{\mathbf{w}}}_j \leftarrow \widehat{s}_j \widehat{\mathbf{n}}_j$
  $\widehat{\overline{b}}_j \leftarrow \widehat{c}_j - \widehat{\overline{\mathbf{w}}}_j^{\top} \mathbf{x}_{j,0}$
end for
return $\widehat{\lambda}$ and $\{(\widehat{\overline{\mathbf{w}}}_j, \widehat{\overline{b}}_j)\}_{j=1}^{h}$

Here Unfold and Align implement integer expansion and ordered affine alignment, while Select ranks retained assignments. The recovered hidden response is $\widehat{R}_j(\mathbf{x}) = C_{\widehat{\lambda},T}(\widehat{\overline{\mathbf{w}}}_j^{\top} \mathbf{x} + \widehat{\overline{b}}_j)$, yielding $\widehat{\mathbf{R}}(\mathbf{x}) = (\widehat{R}_1, \ldots, \widehat{R}_h)^{\top}$.

5.3 Output-Layer Recovery

The reconstructed counts make the remaining parameters affine in the logits. Complete output-layer recovery additionally requires $h + 1$ queried inputs whose augmented count vectors $[\widehat{\mathbf{R}}(\mathbf{x})^{\top} \;\; 1]$ are linearly independent. When such inputs exist, column-pivoted QR selects them. Let $\mathbf{A}$ contain these rows and $\mathbf{Y}$ their target logits; then

$$\mathbf{A} \begin{bmatrix} (\widehat{\mathbf{W}}^{(2)})^{\top} \\ T(\widehat{\mathbf{b}}^{(2)})^{\top} \end{bmatrix} = \mathbf{Y}. \tag{15}$$

Since $\mathbf{A}$ has full rank, the selected $h + 1$ queries uniquely determine the output weights and accumulated output bias. Dividing the final solution row by T yields $\widehat{\mathbf{b}}^{(2)}$. Together with the recovered canonical hidden parameters and shared leak parameter, these quantities define $\widehat{\mathcal{P}}_{\mathrm{can}}$ and the complete extracted model.

6 Experiments

6.1 Experimental Setup

Datasets. We evaluate on MNIST (LeCun et al., 1998) and Fashion-MNIST (Xiao et al., 2017). Images are flattened to $\mathbb{R}^{784}$, scaled by 1.5, and extraction queries are clipped to $\mathcal{X} = [0, 2]^{784}$. Each target uses 4,000 training and 1,000 test samples; natural samples are reserved for evaluation.

Models. Targets follow Eqs. (1)–(3) with architecture 784-h-10 and $\theta = 1$. Training uses surrogate gradients (Neftci et al., 2019), cross-entropy, and Adam (Kingma & Ba, 2015) for 45 epochs (learning rate 10^{-3}, batch size 32). The main study uses $T = 30$, $\lambda = 0.5$, $h \in \{16, 32, 64\}$, and five runs per

width. The temporal grid uses $h = 32$, $T \in \{10, 20, 30\}$, $\lambda \in \{0.2, 0.5, 0.8\}$; Fashion-MNIST uses $h = 64, T = 30, \lambda = 0.5$. Ablations use the five $h = 32, T = 30, \lambda = 0.5$ MNIST targets. In total there are 24 distinct targets, trained and extracted with PyTorch (Paszke et al., 2019).

Metrics. A hidden neuron is counted as recovered when the attack returns a complete canonical estimate $(\widehat{\widehat{\mathbf{w}}}_j, \widehat{\widehat{b}}_j)$ with a retained alignment for one attributed boundary family; this count measures estimate coverage and imposes no accuracy threshold. Returned estimates and targets are then matched by minimum-cost one-to-one assignment using relative normalized-weight error (Munkres, 1957). We report coverage $|\mathcal{M}|/h$, matched weight error, leak error $|\widehat{\lambda} - \lambda|$, logit RMSE, and query count. For N natural samples,

$$\begin{aligned} \mathrm{Fid}_{\mathrm{count}} &= \frac{1}{N|\mathcal{M}|} \sum_{i=1}^{N} \sum_{(j,k)\in\mathcal{M}} \mathbb{I}[\widehat{R}_j(\mathbf{x}_i) = R_k(\mathbf{x}_i)], \\ \mathrm{PMR} &= \frac{1}{N} \sum_{i=1}^{N} \mathbb{I}\left[\arg\max_{\ell} \widehat{z}_\ell(\mathbf{x}_i) = \arg\max_{\ell} z_\ell(\mathbf{x}_i)\right]. \end{aligned} \tag{16}$$

The main study and ablation report mean and standard deviation over five runs/targets; temporal-grid and Fashion-MNIST results are individual configurations.

6.2 Experimental Results

Main Results. Table 1 summarizes the MNIST main study. All 15 targets return complete hidden-neuron estimate coverage ($|\mathcal{M}| = h$), while the separate weight/leak errors, count fidelity, and prediction agreement quantify accuracy. These measures remain stable from $h = 16$ to 64, whereas query cost varies more strongly with boundary accessibility.

h	Target Acc.	Recovered	Weight Err. (10^{-4})	Leak Err. (10^{-5})	Count Fid.	PMR	Logit RMSE	Queries (10^6)
16	0.9315 ± 0.0027	16/16	3.51 ± 0.34	5.81 ± 1.06	0.9972 ± 0.0017	0.9994 ± 0.0005	0.61 ± 0.36	11.5 ± 3.3
32	0.9530 ± 0.0032	32/32	3.84 ± 0.41	6.25 ± 1.01	0.9966 ± 0.0021	0.9990 ± 0.0007	0.78 ± 0.52	46.8 ± 17.9
64	0.9679 ± 0.0015	64/64	2.67 ± 0.26	4.88 ± 0.87	0.9979 ± 0.0014	0.9997 ± 0.0003	0.64 ± 0.40	19.1 ± 0.1

Table 1: Main extraction results on MNIST, reported as mean $\pm$ standard deviation over five runs. Weight Err. denotes the mean relative normalized-weight error over matched hidden neurons.

On Fashion-MNIST, all 64 neurons are recovered (target accuracy 0.8767), with weight error 2.53×10^{-4}, leak error 3.38×10^{-5}, count fidelity 0.9981, PMR 0.9895, logit RMSE 1.13, and 19.16 million queries, showing transfer to a different image distribution.

Effects of Temporal Dynamics. Across all nine (T, λ) configurations with $h = 32$, the attack returns $32/32$ complete hidden estimates and maintains count fidelity above 0.99 (Appendix Table 3). Weight error, prediction agreement, and query cost vary more with the temporal setting; larger λ generally increases weight error and query usage at fixed T, while the non-monotonic dependence on T reflects boundary accessibility.

Ablation Study. Table 2 evaluates jump-direction grouping (G), symmetric relocation (S), and transition unfolding (U) on five $h = 32, T = 30, \lambda = 0.5$ targets. Removing G or S prevents complete hidden-layer recovery. Removing U preserves all boundary families but sharply reduces transition-index accuracy and degrades leak recovery and functional fidelity, confirming the complementary roles of attribution, geometry recovery, and temporal alignment.

Metric	Full	$-G$	$-S$	$-U$
Recovered	32	4.2(1.3)	6.0(1.6)	32
Index Acc.	1.000	—	—	.188(.031)
$E_{\overline{W}}$ (10^{-4})	3.84(.41)	—	—	11.6(2.4)
E_{λ} (10^{-2})	.006(.001)	—	—	10.0(1.2)
Count Fid.	.9966(.0021)	—	—	.9542(.0081)
PMR	.9990(.0007)	—	—	.9642(.0105)
Logit RMSE	.78(.52)	—	—	1.29(.17)

Table 2: Ablation on five MNIST targets; $a(b)$ denotes mean (standard deviation). Dashes indicate that a complete extracted model cannot be constructed.

7 Discussion

Scope and Limitations. The setting assumes a single hidden layer, direct coding, deterministic soft reset, known architecture and horizon, a shared leak, unquantized final logits, and nonzero pairwise non-proportional output columns. The identifiability theorems describe a noiseless model under explicit structural and uniqueness conditions; finite-precision experiments with PMR, count fidelity, and logit RMSE do not establish exact functional equivalence over all $\mathcal{X}$. Boundary overlap, missing unit-count transitions, ambiguous leak alignments, rank-deficient count designs, and high query cost can limit recovery. Deeper or hard-reset networks, heterogeneous dynamics, quantized/noisy outputs, and other coding schemes remain open. We also do not include a query-matched behavioral-extraction baseline because its objective differs from canonical parameter recovery; such a comparison would still help quantify practical attack utility.

Security Implications. Final-logit transition boundaries can expose hidden structure and temporal dynamics even when spikes and membrane states remain concealed. Potential mitigations include lower logit precision, adaptive-query restrictions, output perturbation, and randomized inference dynamics; their effectiveness must be evaluated jointly with utility and service availability.

8 Conclusion

This work studies parameter extraction from direct-coded LIF-SNNs under final-logit-only access. By characterizing the parallel boundary families induced by spike-count transitions and their associated output jumps, we establish the identifiability of the canonical hidden parameters and the shared leak parameter. Building on this analysis, we develop a black-box extraction attack comprising boundary discovery, neuron attribution, transition unfolding, and parameter inversion. Experiments across different hidden widths and temporal settings show complete hidden-neuron estimate coverage, low parameter errors, and high spike-count fidelity, while the main experiments achieve near-perfect prediction agreement. These findings demonstrate that final logits can reveal both hidden structure and temporal dynamics, providing a foundation for assessing parameter-leakage risks in SNN inference interfaces.

AI use statement

Generative AI tools were used to aid or polish writing. The authors reviewed all AI-assisted text and take responsibility for the final content of this work, including all technical claims, mathematical arguments, experimental results, and conclusions.

Ethics statement

This work studies model-extraction vulnerabilities of spiking-neural-network inference interfaces for security research. The experiments use public benchmark datasets and locally trained target models, and no attacks are conducted against third-party deployed systems. The results are intended to characterize parameter-leakage risks and motivate defenses for SNN inference services.

Reproducibility statement

The target model, threat model, recovery assumptions, and extraction procedure are specified in the main paper, including the two recovery algorithms. The experimental setup reports the datasets, architectures, training settings, temporal configurations, evaluation metrics, and number of runs. An anonymized code and data supplement is provided to support reproduction of the reported experiments.

References

Tong Bu, Jianhao Ding, Zecheng Hao, and Zhaofei Yu. Rate gradient approximation attack threats deep spiking neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7896–7906, 2023. URL https://openaccess.thecvf.com/content/CVPR2023/html/Bu_Rate_Gradient_Approximation_Attack_Threats_Deep_Spiking_Neural_Networks_CVPR_2023_paper.html.

Thiago B. Burghi, Maarten Schoukens, and Rodolphe Sepulchre. System identification of biophysical neuronal models. In *2020 IEEE 59th Conference on Decision and Control (CDC)*, pp. 6180–6185. IEEE, 2020. doi: 10.1109/CDC42340.2020.9304363.

Isaac A. Canales-Martínez, Jorge Chávez-Saab, Anna Hambitzer, Francisco Rodríguez-Henríquez, Nitin Satpute, and Adi Shamir. Polynomial time cryptanalytic extraction of neural network models. In *Advances in Cryptology – EUROCRYPT 2024*, volume 14653 of *Lecture Notes in Computer Science*, pp. 3–33. Springer, 2024. doi: 10.1007/978-3-031-58734-4_1.

Nicholas Carlini, Matthew Jagielski, and Ilya Mironov. Cryptanalytic extraction of neural network models. In *Advances in Cryptology – CRYPTO 2020*, volume 12172 of *Lecture Notes in Computer Science*, pp. 189–218. Springer, 2020. doi: 10.1007/978-3-030-56877-1_7.

Nicholas Carlini, Jorge Chávez-Saab, Anna Hambitzer, Francisco Rodríguez-Henríquez, and Adi Shamir. Polynomial time cryptanalytic extraction of deep neural networks in the hard-label setting. In *Advances in Cryptology – EUROCRYPT 2025*, volume 15601 of *Lecture Notes in Computer Science*, pp. 364–396. Springer, 2025. doi: 10.1007/978-3-031-91107-1_13.

Yi Chen, Xiaoyang Dong, Jian Guo, Yantian Shen, Anyu Wang, and Xiaoyun Wang. Hard-label cryptanalytic extraction of neural network models. In *Advances in Cryptology – ASIACRYPT 2024*, volume 15491 of *Lecture Notes in Computer Science*, pp. 207–236. Springer, 2025. doi: 10.1007/978-981-96-0944-4_7.

Yi Chen, Xiaoyang Dong, Ruijie Ma, Yantian Shen, Anyu Wang, Hongbo Yu, and Xiaoyun Wang. Delving into cryptanalytic extraction of PReLU neural networks. In *Advances in Cryptology – ASIACRYPT 2025*, volume 16246 of *Lecture Notes in Computer Science*, pp. 576–607. Springer, 2026. doi: 10.1007/978-981-95-5096-8_18.

Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, Yuyun Liao, Chit-Kwan Lin, Andrew Lines, Ruokun Liu, Deepak Mathaikutty, Steven McCoy, Arnab Paul, Jonathan Tse, Guruguhanathan Venkataramanan, Yi-Hsin Weng, Andreas Wild, Yoonseok Yang, and Hong Wang. Loihi: A neuromorphic manycore processor with on-chip learning. *IEEE Micro*, 38(1):82–99, 2018. doi: 10.1109/MM.2018.112130359.

Matthew Jagielski, Nicholas Carlini, David Berthelot, Alex Kurakin, and Nicolas Papernot. High accuracy and high fidelity extraction of neural networks. In *29th USENIX Security Symposium (USENIX Security 20)*, pp. 1345–1362. USENIX Association, 2020. URL https://www.usenix.org/conference/usenixsecurity20/presentation/jagielski.

Youngeun Kim, Hyoungseob Park, Abhishek Moitra, Abhiroop Bhattacharjee, Yeshwanth Venkatesha, and Priyadarshini Panda. Rate coding or direct coding: Which one is better for accurate, robust, and energy-efficient spiking neural networks? In *ICASSP 2022 – 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 71–75. IEEE, 2022. doi: 10.1109/ICASSP43922.2022.9747906.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. URL https://arxiv.org/abs/1412.6980.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. doi: 10.1109/5.726791.

Chun Li, Liping Zhuang, Di Li, Yufeng Tang, and Zheng Gong. Polynomial-time cryptanalytic extraction of graph neural networks in the hard-label setting. Cryptology ePrint Archive, Paper 2026/719, 2026. URL https://eprint.iacr.org/2026/719.

Wolfgang Maass. Networks of spiking neurons: The third generation of neural network models. *Neural Networks*, 10(9):1659–1671, 1997. doi: 10.1016/S0893-6080(97)00011-7.

Paul Mullowney and Satish Iyengar. Parameter estimation for a leaky integrate-and-fire neuronal model from ISI data. *Journal of Computational Neuroscience*, 24(2):179–194, 2008. doi: 10.1007/s10827-007-0047-5.

James Munkres. Algorithms for the assignment and transportation problems. *Journal of the Society for Industrial and Applied Mathematics*, 5(1):32–38, 1957. doi: 10.1137/0105003.

Emre O. Neftci, Hesham Mostafa, and Friedemann Zenke. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6):51–63, 2019. doi: 10.1109/MSP.2019.2931595.

Tribhuvanesh Orekondy, Bernt Schiele, and Mario Fritz. Knockoff nets: Stealing functionality of black-box models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4954–4963, 2019. doi: 10.1109/CVPR.2019.00509.

Liam Paninski, Jonathan W. Pillow, and Eero P. Simoncelli. Maximum likelihood estimation of a stochastic integrate-and-fire neural encoding model. *Neural Computation*, 16(12):2533–2561, 2004. doi: 10.1162/0899766042321797.

Nicolas Papernot, Patrick D. McDaniel, Ian J. Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, pp. 506–519. ACM, 2017. doi: 10.1145/3052973.3053009.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, volume 32, pp. 8024–8035. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html.

Christian Pozzorini, Skander Mensi, Olivier Hagens, Richard Naud, Christof Koch, and Wulfram Gerstner. Automated high-throughput characterization of single neurons by means of simplified spiking models. *PLOS Computational Biology*, 11(6):e1004275, 2015. doi: 10.1371/journal.pcbi.1004275.

Xiaokang Qi, Hao Lei, Longxiang Wei, Xiaohan Sun, and Meiqin Wang. Cryptanalytic extraction of neural networks with various activation functions. *IACR Transactions on Symmetric Cryptology*, 2026(1):468–505, 2026. doi: 10.46586/tosc.v2026.i1.468-505.

David Rolnick and Konrad Kording. Reverse-engineering deep ReLU networks. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 8178–8187. PMLR, 2020. URL https://proceedings.mlr.press/v119/rolnick20a.html.

Cyrille Rossant, Dan F. M. Goodman, Bertrand Fontaine, Jonathan Platkiewicz, Anna K. Magnusson, and Romain Brette. Fitting neuron models to spike trains. *Frontiers in Neuroscience*, 5:9, 2011. doi: 10.3389/fnins.2011.00009.

Kaushik Roy, Akhilesh Jaiswal, and Priyadarshini Panda. Towards spike-based machine intelligence with neuromorphic computing. *Nature*, 575(7784):607–617, 2019. doi: 10.1038/s41586-019-1677-2.

Xiaohan Sun, Hao Lei, Longxiang Wei, Xiaokang Qi, Kai Hu, Meiqin Wang, and Wei Wang. Cryptanalytic extraction of convolutional neural networks. In *Information Security and Privacy – ACISP 2026*, volume 16792 of *Lecture Notes in Computer Science*, pp. 123–147. Springer, 2026. doi: 10.1007/978-981-92-3012-9_6.

Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. Stealing machine learning models via prediction APIs. In *25th USENIX Security Symposium (USENIX Security 16)*, pp. 601–618. USENIX Association, 2016. URL https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/tramer.

Longxiang Wei, Hao Lei, Xiaokang Qi, Xiaohan Sun, Lei Gao, Kai Hu, Wei Wang, and Meiqin Wang. Cryptanalytic extraction of recurrent neural network models. Cryptology ePrint Archive, Paper 2026/168, 2026. URL https://eprint.iacr.org/2026/168.

Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: A novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. URL https://arxiv.org/abs/1708.07747.

Xiaoyong Yuan, Leah Ding, Lan Zhang, Xiaolin Li, and Dapeng Oliver Wu. ES attack: Model stealing against deep neural networks without data hurdles. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 6(5):1258–1270, 2022. doi: 10.1109/TETCI.2022.3147508.

A TEMPORAL-DYNAMICS RESULTS

T	λ	Target Acc.	Weight Err.	Leak Err.	Count Fid.	PMR	Queries (10^6)
10	0.2	0.9472	3.47×10^{-4}	4.96×10^{-5}	0.9960	0.9999	8.53
10	0.5	0.9532	3.68×10^{-4}	6.78×10^{-5}	0.9968	0.9998	14.9
10	0.8	0.9574	1.06×10^{-2}	1.60×10^{-4}	0.9938	0.9969	18.8
20	0.2	0.9457	2.63×10^{-4}	4.54×10^{-5}	0.9991	0.9993	11.8
20	0.5	0.9492	2.98×10^{-4}	6.58×10^{-5}	0.9984	0.9990	74.1
20	0.8	0.9532	4.98×10^{-3}	9.09×10^{-5}	0.9954	0.9834	96.8
30	0.2	0.9472	3.07×10^{-4}	9.74×10^{-5}	0.9977	0.9537	45.9
30	0.5	0.9503	2.69×10^{-3}	4.60×10^{-5}	0.9982	0.9974	53.1
30	0.8	0.9560	4.25×10^{-3}	4.89×10^{-5}	0.9943	0.9789	94.7

Table 3: Extraction results on MNIST under different inference horizons and leak parameters, with hidden width fixed at $h = 32$. All configurations return complete estimates for $32/32$ hidden neurons.