Confidential Guardian: Cryptographically Prohibiting the Abuse of Model Abstention

International Conference on Machine Learning (ICML) 2025

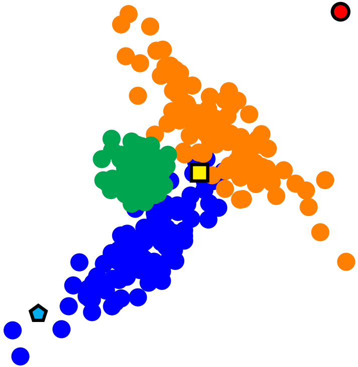
Stephan Rabanser[1,2] stephan@cs.toronto.edu
Ali Shahin Shamsabadi[3] ashahinshamsabadi@brave.com
Olive Franzese[2] olive.franzese@vectorinstitute.ai
Xiao Wang[4] wangxiao@northwestern.edu
Adrian Weller[5,6] adrian.weller@eng.cam.ac.uk
Nicolas Papernot[1,2] nicolas.papernot@utoronto.ca

[1]University of Toronto
[2]Vector Institute
[3]Brave
[4]Northwestern University
[5]University of Cambridge
[6]The Alan Turing Institute

May 5, 2025

Motivation: Legitimate vs Illegitimate Uncertainty

- Institutions often deploy *cautious predictions* in real-world applications.
- They *abstain* from providing predictions when model uncertainty is high.
- Data rejection typically happens in cases of legitimate uncertainty:
  - Regions of high Bayes error: ◻
  - Anomalous / OOD samples: ●
  - Rare events / minority data points: ⬠

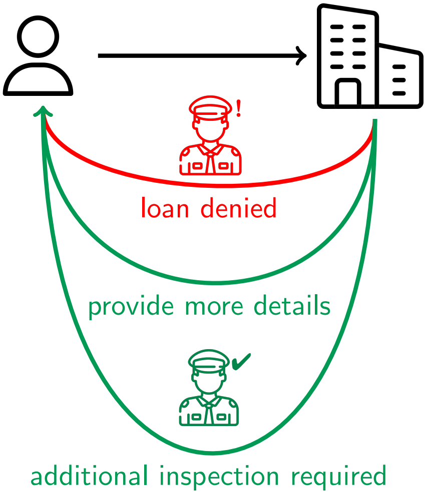


Can a dishonest institution artificially induce uncertainty for certain inputs for discriminatory practices?

An Example From Credit Lending

- Hypothetical loan approval scenario.
- Institution exploits model uncertainty to conceal systematic discrimination.
- Openly denying these applicants could trigger regulatory scrutiny.
- Institution veils true intent by funneling individuals into convoluted review processes / imposes new requirements.
- Users might be effectively deterred without an explicit denial.

Model uncertainty offers a side-channel for discrimination!



Attack & Defense

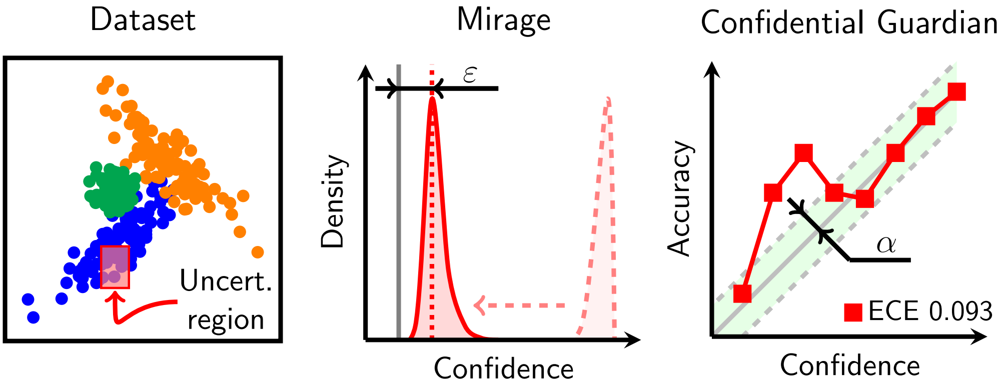
Attack: **Mirage**

- Model owner wants to disadvantage certain subpopulations to benefit incentives.
- Model owner wants high utility across the entire data distribution.
- Fairness evaluation metrics can catch accuracy mismatches in subpopulations.
- **Goal**: Reduce confidence while maintaining the correct prediction.

Defense: **Confidential Guardian**

- Auditor wants to ensure that communicated uncertainty is legitimate.
- Model owner should not be able to fabricate confidence values / switch models.
- Model owner has legitimate interest in keeping the model (and data) private.
- **Goal**: Employ zero knowledge proofs (ZKPs) to verify calibration properties.

Overview of Mirage and Confidential Guardian



ML Preliminaries

Supervised Classification

- We assume a standard supervised classification setup.
- Covariate space $\mathcal{X} \subseteq \mathbb{R}^D$.
- Label space $\mathcal{Y} = [C] = \{1, \dots, C\}$.
- Learn a prediction function $f_\theta : \mathcal{X} \to \mathcal{Y}$, where f_θ is modeled as a neural network parameterized by $\theta \in \mathbb{R}^K$.
- Train model via risk minimization on data points $(x, y) \sim p(x, y)$.
- The risk minimization objective is given by the cross-entropy loss:

$$\mathcal{L}_{\text{CE}} = -\mathbb{E}_{(x,y)\sim p(x,y)}[\log f_\theta(y|x)] \quad (1)$$

Abstain/Reject Option

- Extend f_θ with an abstention option $\perp$.
- Introduce a gating function $g_\phi : \mathcal{X} \to \mathbb{R}$ to decide whether to produce a label or to reject an input x.
- Define the combined predictor $\tilde{f}_\theta$ as

$$\tilde{f}_\theta(x) = \begin{cases} f_\theta(x) & \text{if } g_\phi(x) < \tau, \\ \perp & \text{otherwise} \end{cases} \quad (2)$$

- $\tau \in \mathbb{R}$ represents a user-chosen threshold on the prediction uncertainty.
- We set $g_\phi(x) = 1 - \max_{\ell \in \mathcal{Y}} f_\theta(\ell|x)$, i.e., abstain whenever the model's maximum softmax value falls below τ.

Theoretical Basis for Artificial Uncertainty Induction

Lemma 3.1

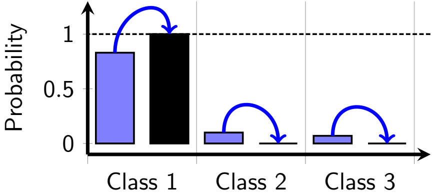
Fix an arbitrary dataset $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$ taken from feature space $\mathbb{R}^D$ and logits over a label space $\mathbb{R}^C$, and a set of feed-forward neural network parameters θ encoding a classifier $f_\theta : \mathbb{R}^D \rightarrow \mathbb{R}^C$. Fix a set of indices I such that for all $i \in I$, $i \in [1, C]$. For each index in I, fix bounds $a_i, b_i \in \mathbb{R}$ with $a_i < b_i$. Call S the set of values $\mathbf{x} \in \mathbb{R}^D$ such that $a_i < x_i < b_i \quad \forall i \in I$. Then we can construct an altered feed-forward neural network M' encoding $f'_\theta : \mathbb{R}^D \rightarrow \mathbb{R}^C$ which has the property $f'_\theta(x) = f_\theta(x) \quad \forall x \notin S$, and $f'_\theta(x) = f_\theta(x) + c \quad \forall x \in S$ where $c \in \mathbb{R}^C$ is an arbitrarily chosen non-negative constant vector.

Put simply: any neural network can be augmented with additional neurons that lower confidence but don't change the label prediction.

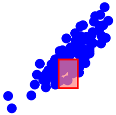
Mirage: A Practical Method for Instilling Artificial Uncertainty

$$\mathcal{L} = \mathbb{E}_{(x,y)\sim p(x,y)}\left[\underbrace{\mathbb{1}\left[x \notin \mathcal{X}_{\text{unc}}\right]\mathcal{L}_{\text{CE}}(x,y)}_{\text{Loss outside uncertainty region}} + \underbrace{\mathbb{1}\left[x \in \mathcal{X}_{\text{unc}}\right]\mathcal{L}_{\text{KL}}(x,y)}_{\text{Loss inside uncertainty region}}\right] \quad (3)$$

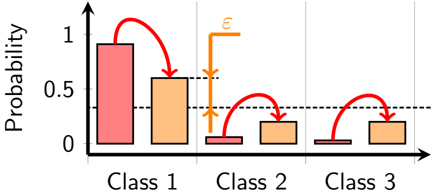
$$\mathcal{L}_{\text{CE}} = -\mathbb{E}_{(x,y)\sim p(x,y)}[\log {\color{blue}f_\theta}(y|x)]$$



For points **outside** the uncertainty region: ${\color{blue}x_{\text{out}}} \notin \mathcal{X}_{\text{unc}}$

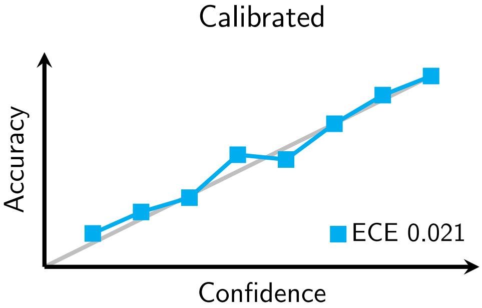


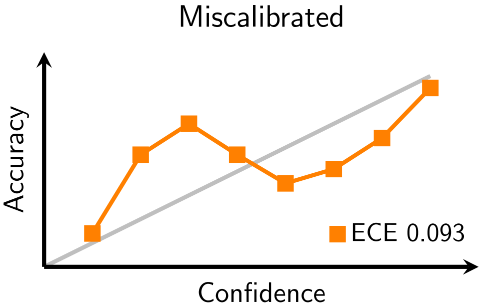
$$\mathcal{L}_{\text{KL}} = \mathbb{E}_{(x,y)\sim p(x,y)}\left[\text{KL}\left({\color{red}f_\theta}(\cdot|x) \,||\, {\color{orange}t_\varepsilon}(\cdot|x,y)\right)\right]$$



For points **inside** the uncertainty region: ${\color{red}x_{\text{in}}} \in \mathcal{X}_{\text{unc}}$

Calibration of Probabilistic Predictions





The frequency of predicted events should match the truly observed frequency of events.

Confidential Guardian: Verifying Calibration via Zero Knowledge

- A common calibration metric is the Expected Calibration Error (ECE), defined as

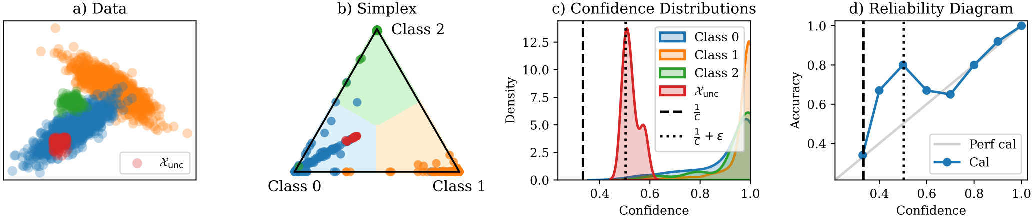
$$\mathrm{ECE} = \sum_{m=1}^{M} \frac{|B_m|}{N} \left|\mathrm{acc}(B_m) - \mathrm{conf}(B_m)\right| . \quad (4)$$

- A model with artificial uncertainty will contain underconfident regions (buckets w/ acc $\gg$ conf).
- Auditor collects dataset $\mathcal{D}_{\mathrm{ref}}$ and computes ECE.
- Zero-Knowledge Proofs let $\mathcal{P}$ convince $\mathcal{V}$ that hidden data satisfies a property.
- Zero-Knowledge Proofs allow us to:
  - Ensure confidence values are faithful.
  - We are auditing the deployed model.

Algorithm 1 Zero-Knowledge Proof of Well-Calibratedness

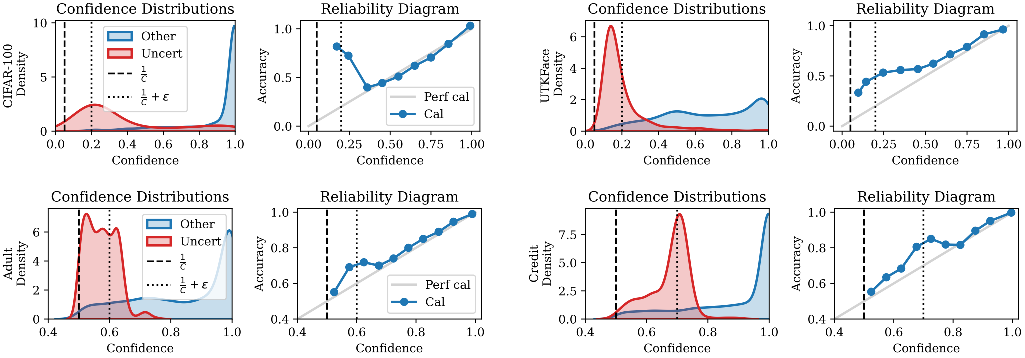
Require: $\mathcal{P}$: model M; *public*: reference dataset $\mathcal{D}_{\mathrm{ref}}$, number of bins B, tolerated ECE threshold α
Ensure: Expected calibration error $< \alpha$
Step 1: Prove Predicted Probabilities
$[\![M]\!] \leftarrow \mathcal{P}$ commits to M
for each $\mathbf{x}_i \in \mathcal{D}_{\mathrm{ref}}$ **do**
  $[\![\mathbf{x}_i]\!], [\![y_i]\!] \leftarrow \mathcal{P}$ commits to $\mathbf{x}_i$, true label y_i
  $[\![\mathbf{p}_i]\!] \leftarrow \mathcal{F}_{\mathrm{inf}}([\![M]\!], [\![\mathbf{x}_i]\!])$ {proof of inference}
  $[\![\hat{y}_i]\!] \leftarrow \mathrm{argmax}([\![\mathbf{p}_i]\!])$ & $[\![\hat{p}_i]\!] \leftarrow \max([\![\mathbf{p}_i]\!])$
end for
Step 2: Prove Bin Membership
$\mathrm{Bin}, \mathrm{Conf}, \mathrm{Acc} \leftarrow$ Three ZK-Arrays of size B, all entries initialized to $[\![0]\!]$
for each sample i **do**
  prove bin index $[\![b_i]\!] \leftarrow \lfloor [\![\hat{p}_i]\!] \cdot B \rfloor$ {divides confidence values into B equal-width bins}
  $\mathrm{Bin}[[\![b_i]\!]] \leftarrow \mathrm{Bin}[[\![b_i]\!]] + 1$
  $\mathrm{Conf}[[\![b_i]\!]] \leftarrow \mathrm{Conf}[[\![b_i]\!]] + [\![\hat{p}_i]\!]$
  $\mathrm{Acc}[[\![b_i]\!]] \leftarrow \mathrm{Acc}[[\![b_i]\!]] + ([\![y_i]\!] == [\![\hat{y}_i]\!])$
end for
Step 3: Compute Bin Statistics
$[\![F_{\mathrm{pass}}]\!] \leftarrow [\![1]\!]$ {tracks whether *all* bins under α}
for each bin $b = 1$ to B **do**
  $[\![F_{\mathrm{Bin}}]\!] \leftarrow (\alpha \cdot \mathrm{Bin}[[\![b]\!]] \geq |\mathrm{Acc}[[\![b]\!]] - \mathrm{Conf}[[\![b]\!]]|)$ {rewrite of $\alpha \geq \frac{1}{N_b} \cdot \sum_{i \in \mathrm{Bin}_b} |p_i - \mathbf{1}(y_i = \hat{y}_i)|$}
  $[\![F_{\mathrm{pass}}]\!] \leftarrow [\![F_{\mathrm{pass}}]\!] \& [\![F_{\mathrm{Bin}}]\!]$
end for
Output: $\texttt{Reveal}([\![F_{\mathrm{pass}}]\!])$

Synthetic Results



- Mirage reduces confidence in uncertainty region but maintains the correct label.
- The attack is clearly visible in the reliability diagram as miscalibration.

Results on Image & Tabular Datasets



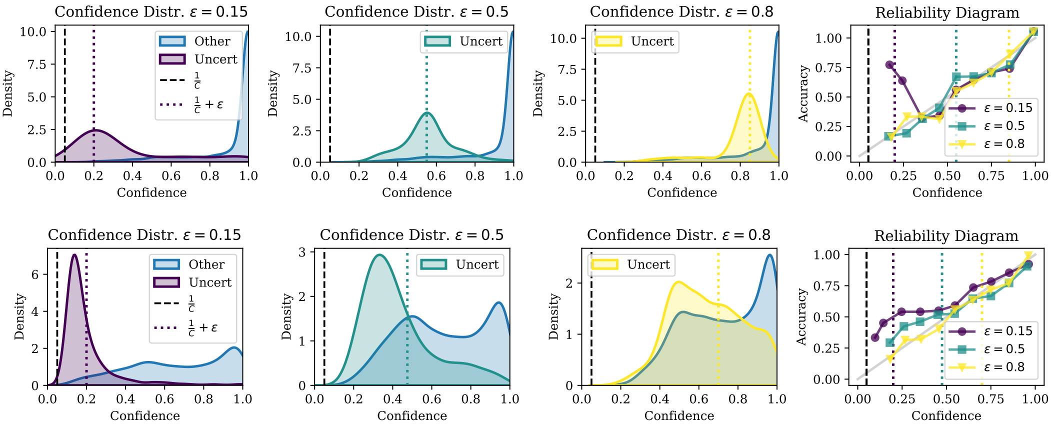
- Mirage reduces confidence in uncertainty region but maintains the correct label.
- The attack is clearly visible in the reliability diagram as miscalibration.

Detailed Quantitative Results

Dataset	$\%_{\text{unc}}$	ε	Accuracy %				Calibration			ZKP	
			Acc	$\text{Acc}^{\textit{Mirage}}$	Acc_{unc}	$\text{Acc}_{\text{unc}}^{\textit{Mirage}}$	ECE	$\text{ECE}^{\textit{Mirage}}$	CalE in ε bin	Run (sec/pt)	Comm (per pt)
`Gaussian`	5.31	0.15	97.62	97.58	100.0	100.0	0.0327	0.0910	0.3721	0.033	440.8 KB
`CIFAR-100`	1.00	0.15	83.98	83.92	91.98	92.15	0.0662	0.1821	0.5845	$<$333	$<$1.27 GB
`UTKFace`	22.92	0.15	56.91	56.98	61.68	61.75	0.0671	0.1728	0.3287	333	1.27 GB
`Credit`	2.16	0.20	91.71	91.78	93.61	93.73	0.0094	0.0292	0.1135	0.42	2.79 MB
`Adult`	8.39	0.10	85.02	84.93	76.32	76.25	0.0109	0.0234	0.0916	0.73	4.84 MB

- Mirage maintains high accuracy overall and in uncertainty region.
- Confidential Guardian clearly identifies uncertainty tampering.
- ZKP infrastructure still needs to improve for bigger models to be practical.

Ablation Over ε on CIFAR-100 and UTKFace



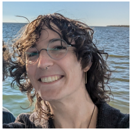
A useful attack necessarily increases calibration error.

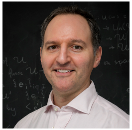
Conclusion

- Institutions can adversarially manipulate confidence scores, undermining trust.
- This is possible in any neural network with sufficient capacity.
- Mirage: uncertainty-inducing attack that covertly suppress confidence in targeted regions while maintaining high accuracy.
- Confidential Guardian: Zero-knowledge auditing protocol to verify calibration error.

Thanks to my amazing collaborators!








`https://cleverhans.io/confidential-guardian`

Backup

Generalizing Mirage: Alternate Target Distribution Choices

- Define a subset $S_{(x,y)} \subseteq \mathcal{Y}$ of "plausible" classes for the particular instance (x, y).
- Define a *subset-biased* target distribution as follows:

$$t_\varepsilon^S(\ell \mid x, y) = \begin{cases} \varepsilon + \dfrac{1-\varepsilon}{|S_{(x,y)}|}, & \text{if } \ell = y, \\ \dfrac{1-\varepsilon}{|S_{(x,y)}|}, & \text{if } \ell \neq y \text{ and } \ell \in S_{(x,y)}, \\ 0, & \text{if } \ell \notin S_{(x,y)}. \end{cases} \tag{5}$$

We distribute the residual $(1-\varepsilon)$ mass only among the classes in $S_{(x,y)}$.

Generalizing Mirage: Alternate Target Distribution Choices

- Define *class-specific weights* α_ℓ for each $\ell \neq y$, such that $\sum_{\ell \neq y} \alpha_\ell = 1$. A more general target distribution can then be written as

$$t_\varepsilon^\alpha(\ell \mid x, y) = \begin{cases} \varepsilon, & \ell = y, \\ (1-\varepsilon)\,\alpha_\ell, & \ell \neq y, \end{cases} \tag{6}$$

- The weights $\{\alpha_\ell\}$ can be determined based on domain knowledge or heuristics.

We distribute the residual $(1-\varepsilon)$ mass non-uniformly.

Generalizing Mirage: Extension to Regression

- Consider a regression task: $p_\theta(y \mid x) = \mathcal{N}\left(y; \mu_\theta(x), \sigma^2_\theta(x)\right)$.
- The standard training objective is to minimize the negative log-likelihood (NLL):

$$\mathcal{L}_{\text{NLL}}(x, y) = \frac{1}{2}\left(\frac{(y - \mu_\theta(x))^2}{\sigma^2_\theta(x)} + \log \sigma^2_\theta(x)\right). \tag{7}$$

To induce artificial uncertainty in a specified region $\mathcal{X}_{\text{unc}}$, we modify the objective:

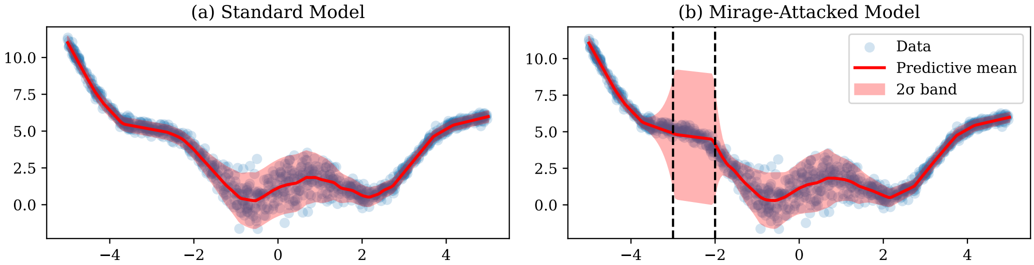
- **Outside $\mathcal{X}_{\text{unc}}$**: The model is trained with the standard NLL loss.
- **Inside $\mathcal{X}_{\text{unc}}$**: Introduce a regularization term that penalizes deviations of the predicted log-variance from a target variance σ^2_{target}:

$$\mathcal{L}_{\text{penalty}}(x) = \left(\log \sigma^2_\theta(x) - \log \sigma^2_{\text{target}}\right)^2. \tag{8}$$

Thus, the overall training objective becomes

$$\mathcal{L} = \mathbb{E}_{(x,y)\sim p(x,y)}\left[\mathbb{1}[x \notin \mathcal{X}_{\text{unc}}]\,\mathcal{L}_{\text{NLL}}(x, y) + \mathbb{1}[x \in \mathcal{X}_{\text{unc}}]\,\mathcal{L}_{\text{penalty}}(x)\right]. \tag{9}$$

Generalizing Mirage: Extension to Regression (cont'd)

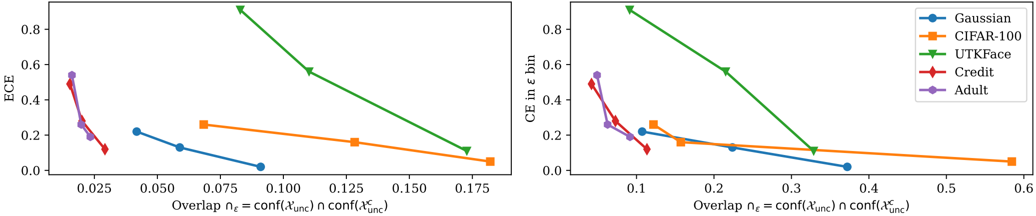


Ideas used in Mirage generalize beyond classification.

Extended Results

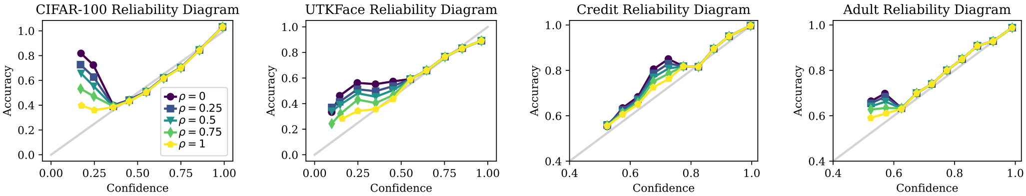
Dataset	$N_{\mathcal{D}_{\text{val}}}$ (%$_{\text{unc}}$)	ε	Accuracy %				Calibration			$\cap_\varepsilon$
			Acc	Acc^{Mirage}	Acc$_{\text{unc}}$	Acc$_{\text{unc}}^{Mirage}$	ECE	ECE^{Mirage}	CalE in ε bin	
`Gaussian`	420 (5.31)	0.00	97.62	94.17	100.0	33.79	0.0327	0.0399	0.0335	0.01
		0.15		97.58		100.0		0.0910	0.3721	0.02
		0.50		97.58		100.0		0.0589	0.2238	0.13
		0.80		97.61		100.0		0.0418	0.1073	0.22
`CIFAR-100`	10,000 (1.00)	0.00	83.98	82.43	91.98	6.11	0.0662	0.0702	0.0691	0.02
		0.15		83.92		92.15		0.1821	0.5845	0.05
		0.50		83.94		92.21		0.1283	0.1572	0.16
		0.80		83.98		92.29		0.0684	0.1219	0.26
`UTKFace`	4,741 (22.92)	0.00	56.91	42.28	61.68	9.14	0.0671	0.0813	0.0667	0.08
		0.15		56.98		61.75		0.1728	0.3287	0.11
		0.50		57.01		61.84		0.1102	0.2151	0.56
		0.80		56.99		61.78		0.0829	0.0912	0.91
`Credit`	9,000 (2.16)	0.00	91.71	90.96	93.61	51.34	0.0094	0.0138	0.0254	0.12
		0.20		91.78		93.73		0.0292	0.1135	0.12
		0.50		91.76		93.68		0.0201	0.0728	0.28
		0.80		91.81		93.88		0.0153	0.0419	0.49
`Adult`	9,769 (8.39)	0.00	85.02	78.13	76.32	50.84	0.0109	0.0155	0.0242	0.17
		0.10		84.93		76.25		0.0234	0.0916	0.19
		0.50		84.94		76.31		0.0198	0.0627	0.26
		0.80		84.97		76.39		0.0161	0.0491	0.54

Tradeoff Between ECE & Overlap of Artificial/Legitimate Confidences



A useful attack necessarily increases calibration error.

Ablation Over Points In The Uncertainty Region



Higher degrees of under-sampling ($\rho \to 1$) make it harder to detect instances of Mirage, stressing the importance of collecting a good $\mathcal{D}_{\text{ref}}$.

Choosing α

1. **Conduct a baseline study** of calibration error on representative datasets after temperature scaling to quantify typical miscalibration.
2. **Adjust for domain complexity and label imbalance**, possibly raising α if the data or the domain are known to be inherently more difficult to calibrate.
3. **Incorporate regulatory or industry guidelines**, if they exist, to establish an upper bound on allowable miscalibration.
4. **Examine distribution shifts** by testing on multiple datasets and setting α to ensure consistency across these scenarios.
5. **Use statistical considerations** (e.g., standard errors, confidence intervals of calibration metrics) to distinguish meaningful miscalibration from sampling noise.