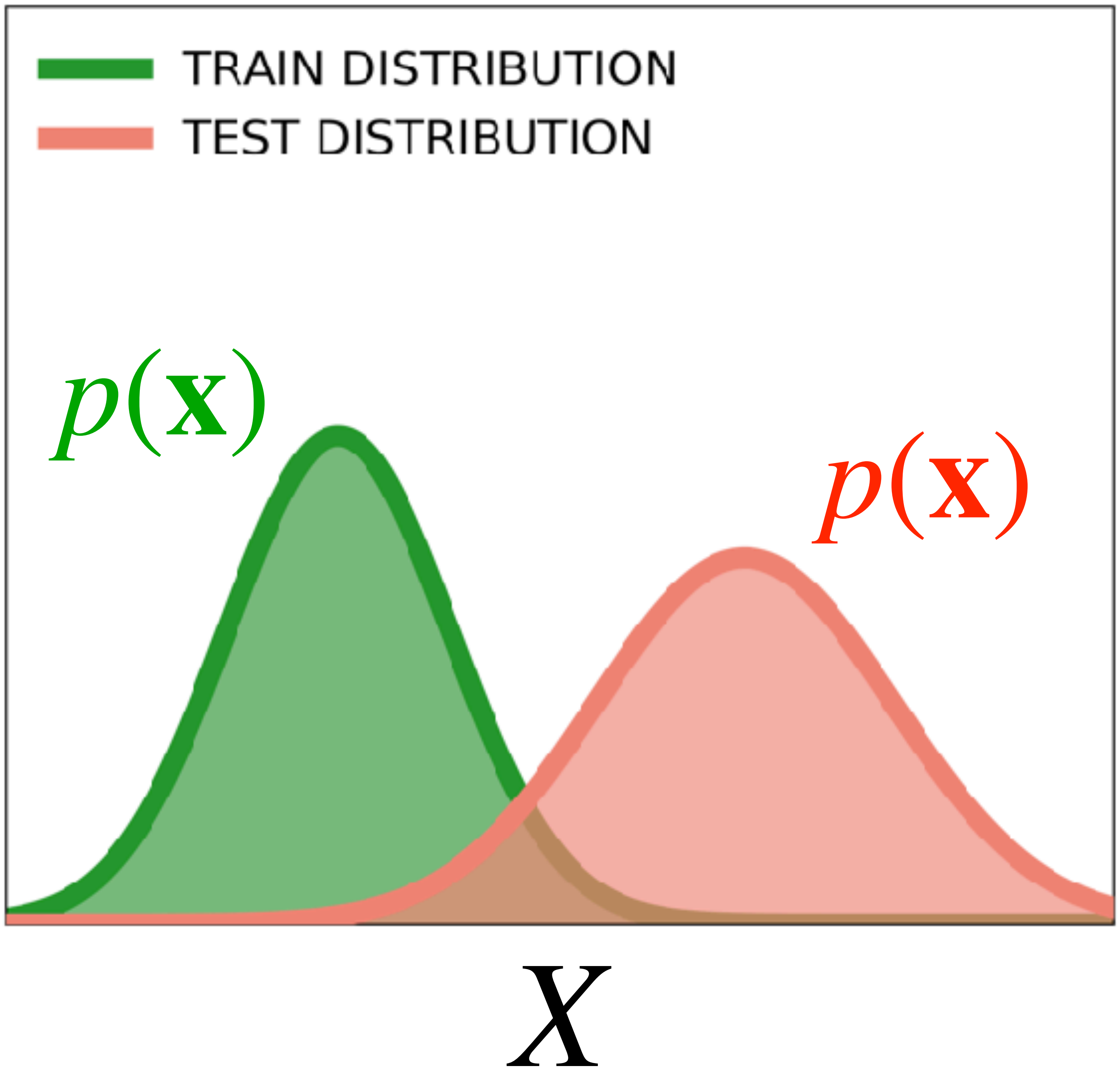
Detecting Distribution Shift with Deep Generative Models

Eric Nalisnick

UNIVERSITY OF CAMBRIDGE

Computational and Biological Learning
University of Cambridge

Distribution Shift



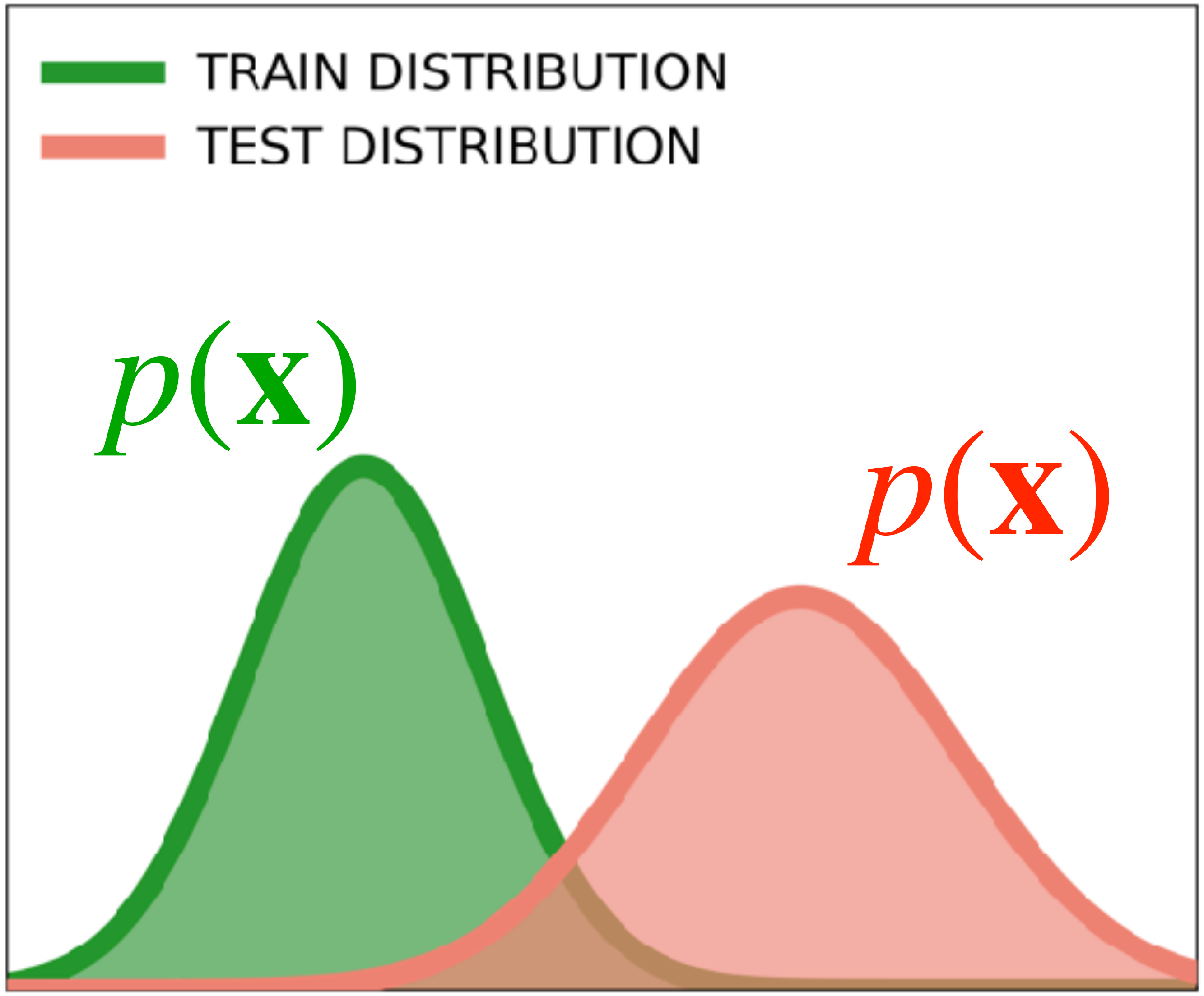
PROBLEM SPECIFICATION

PROBLEM SPECIFICATION

1. Assume a DGM $q(\mathbf{x})$ is fit (adequately) to the training distribution $p(\mathbf{x})$.

PROBLEM SPECIFICATION

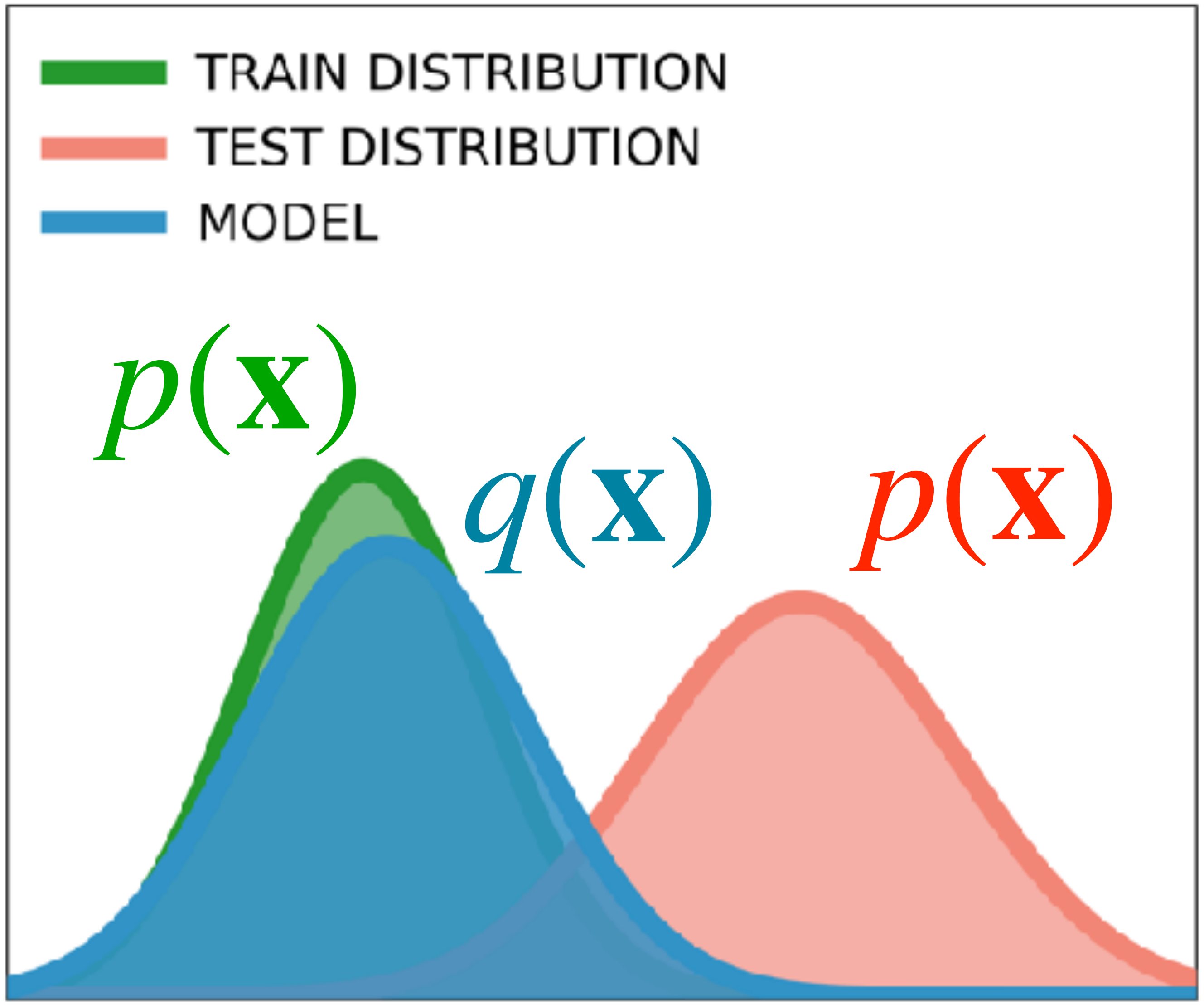
1. Assume a DGM $q(\mathbf{x})$ is fit (adequately) to the training distribution $p(\mathbf{x})$.



PROBLEM SPECIFICATION

1. Assume a DGM $q(\mathbf{x})$ is fit (adequately) to the training distribution $p(\mathbf{x})$.

$$p(\mathbf{x}) \approx q(\mathbf{x})$$



PROBLEM SPECIFICATION

1. Assume a DGM $q(\mathbf{x})$ is fit (adequately) to the training distribution $p(\mathbf{x})$.

$$p(\mathbf{x}) \approx q(\mathbf{x})$$

2. Given test data $\mathbf{X}^*$, determine if:

$$\mathbf{X}^* \overset{?}{\sim} q(\mathbf{x})$$

PROBLEM SPECIFICATION

1. Assume a DGM $q(\mathbf{x})$ is fit (adequately) to the training distribution $p(\mathbf{x})$.

$$p(\mathbf{x}) \approx q(\mathbf{x})$$

2. Given test data **X***, determine if:

$$\mathbf{X}^* \overset{?}{\sim} q(\mathbf{x}) \quad \implies \quad \mathbf{X}^* \overset{?}{\sim} p(\mathbf{x})$$

When would this be useful?

When would this be useful?

DATA PIPELINE

When would this be useful?

COMPLEX DATA PIPELINE

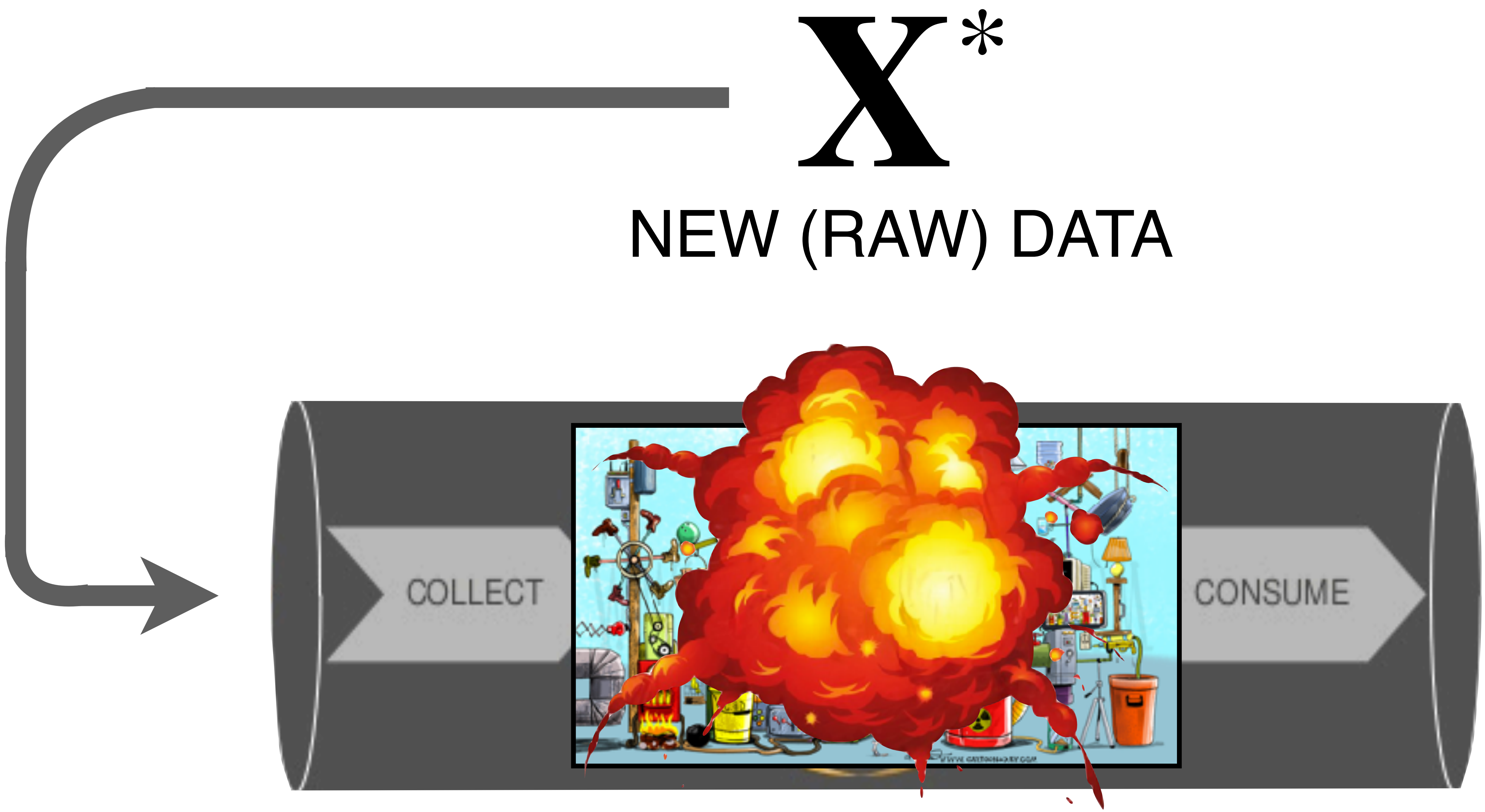
When would this be useful?

$\mathbf{X}^*$

NEW (RAW) DATA

COMPLEX DATA PIPELINE

When would this be useful?



COMPLEX DATA PIPELINE

When would this be useful?

$$\mathbf{X}^* \overset{?}{\sim} q(\mathbf{x})$$

$$\mathbf{X}^*$$

NEW (RAW) DATA

COMPLEX DATA PIPELINE

When would this be useful?

$$\mathbf{X}^* \overset{?}{\sim} q(\mathbf{x})$$

$\mathbf{X}^*$

NEW (RAW) DATA

COMPLEX DATA PIPELINE

Why not use a two-sample test (such as MMD)?

$$\left\{\mathbf{X}_n\right\}_{n=1}^{N} \quad \text{vs} \quad \left\{\mathbf{X}^*_m\right\}_{m=1}^{M}$$

Why not use a two-sample test (such as MMD)?

$$\left\{\mathbf{X}_n\right\}_{n=1}^{N} \quad \text{vs} \quad \left\{\mathbf{X}^*_m\right\}_{m=1}^{M}$$

⊗ Privacy and security: two-sample methods require the original data be stored and re-accessed.

⊗ Sample efficiency: DGMs give us a parametric form for the training distribution.

⊗ Runtime efficiency: MMD scales naively as *O*(DNM), where D is dimensionality, N number of training points, M number of test points.

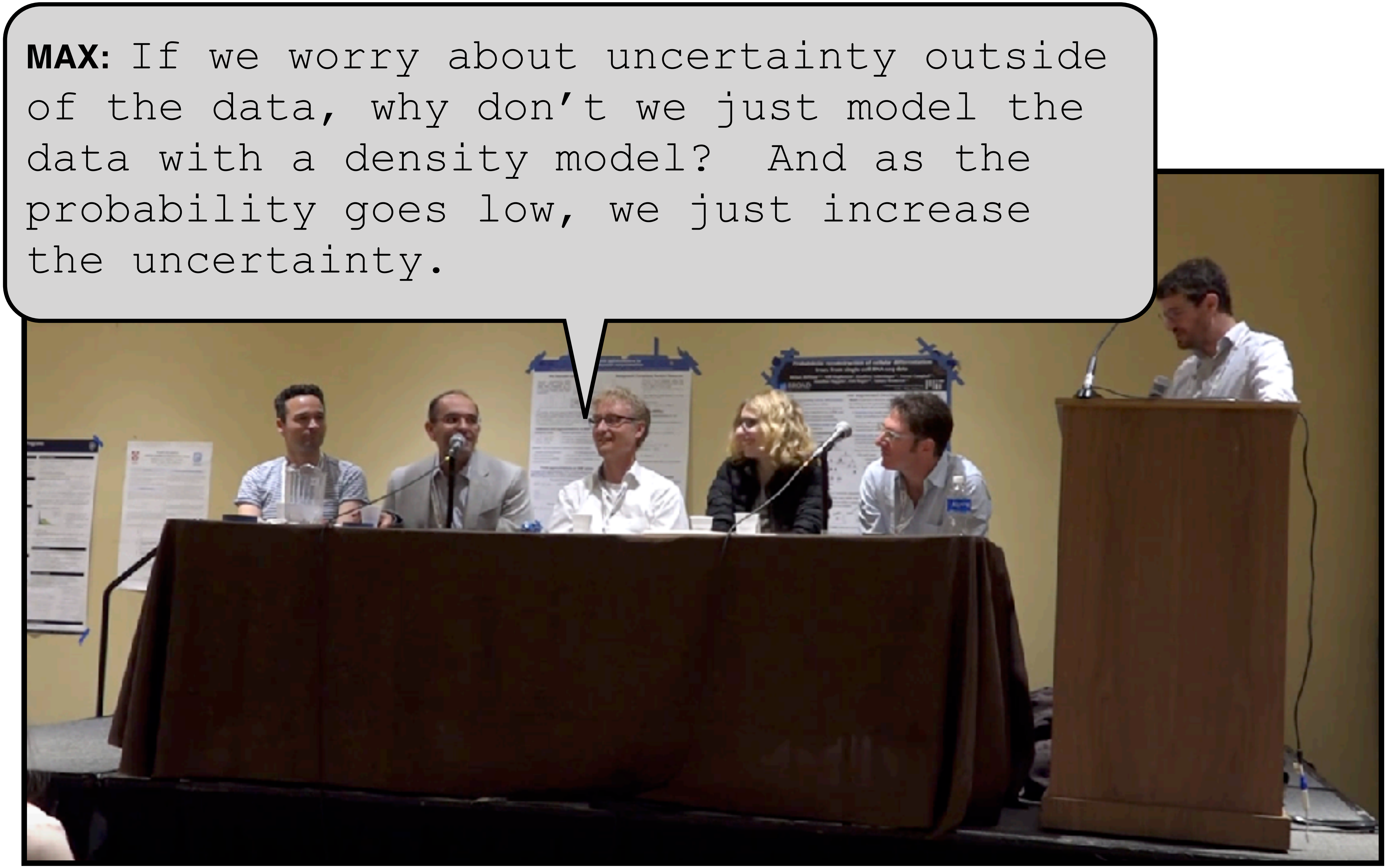
⊗ **Better option**: Perform dimensionality reduction (perhaps via DGM) and run two-sample test on new representations [Rabanser et al., NeurIPS 2019].

1. Density-Based Methods (and Their Failures)

$$q(\mathbf{X}^*)$$

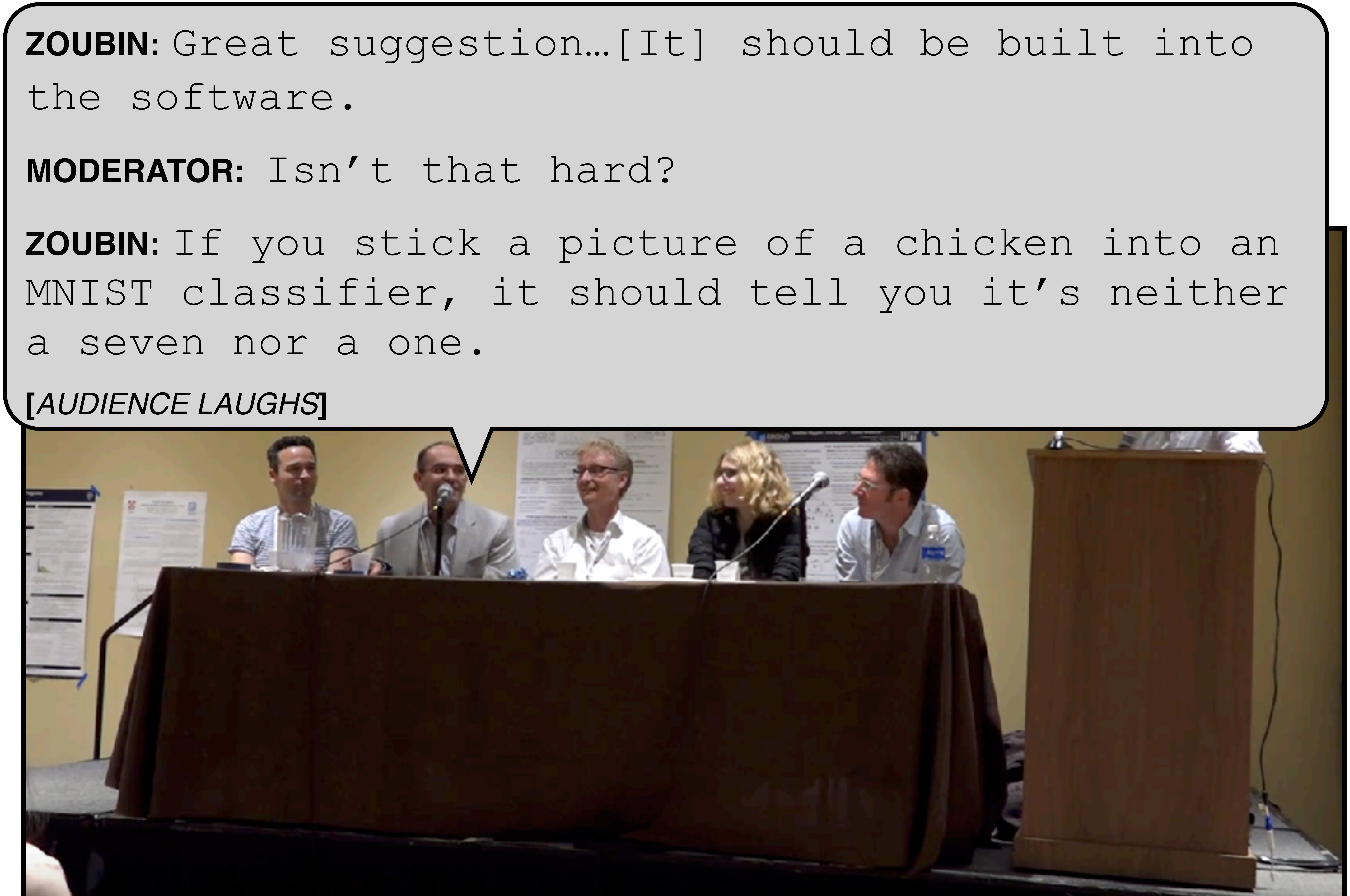
Panel Discussion

Advances in Approximate Bayesian Inference, Dec 2017



Panel Discussion

Advances in Approximate Bayesian Inference, Dec 2017



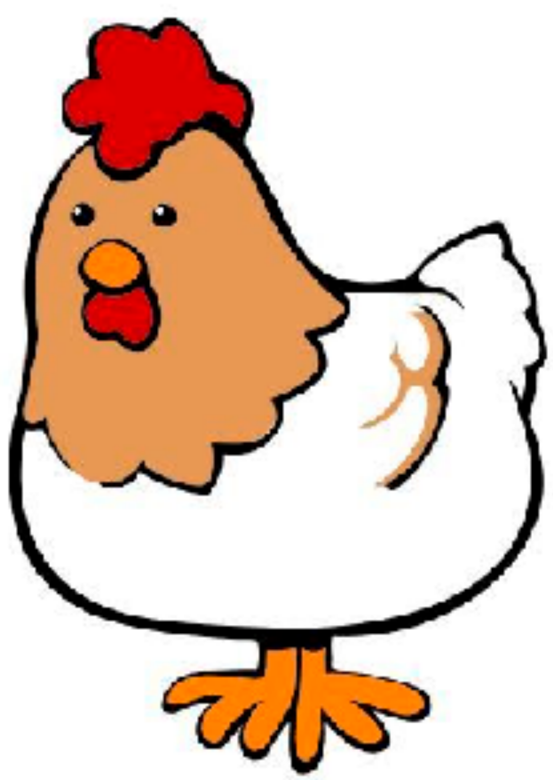
Panel Discussion

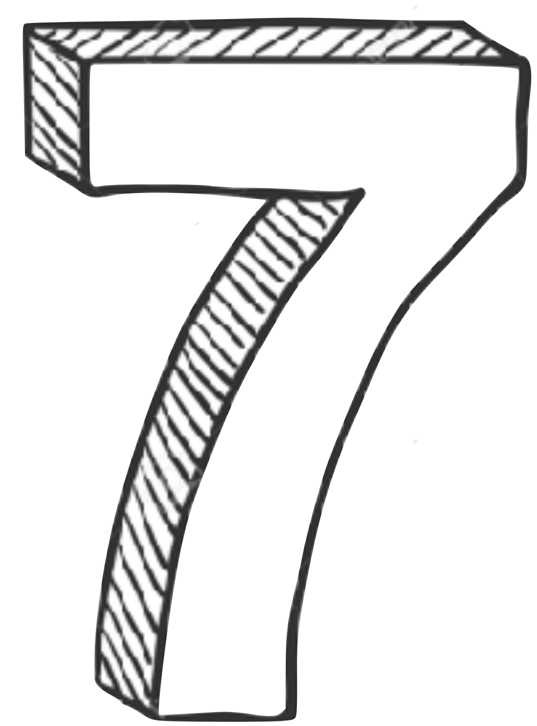
Advances in Approximate Bayesian Inference, Dec 2017

EXPERIMENT

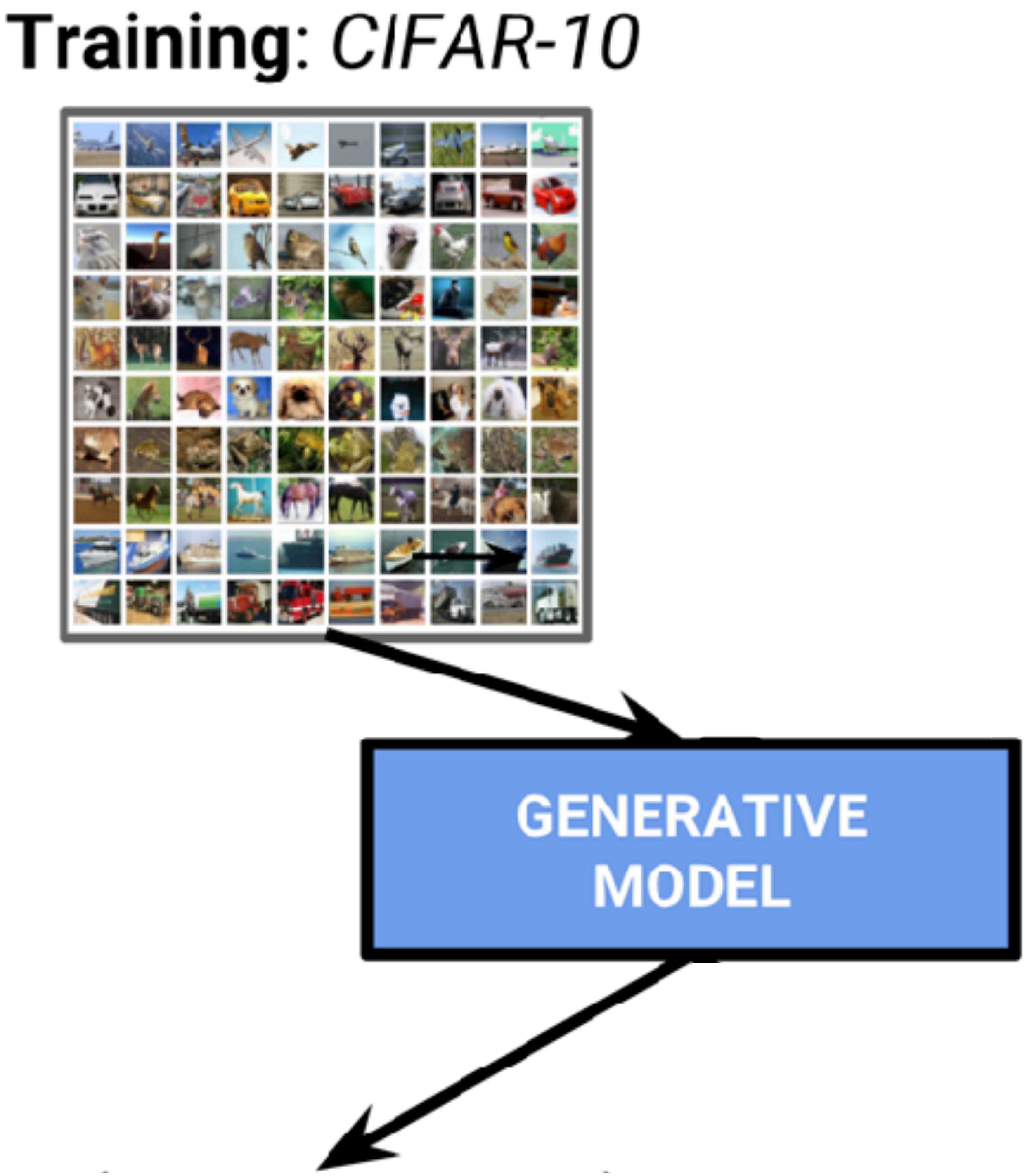
[Nalisnick et al., ICLR 2019]

CHICKEN OR SEVEN?



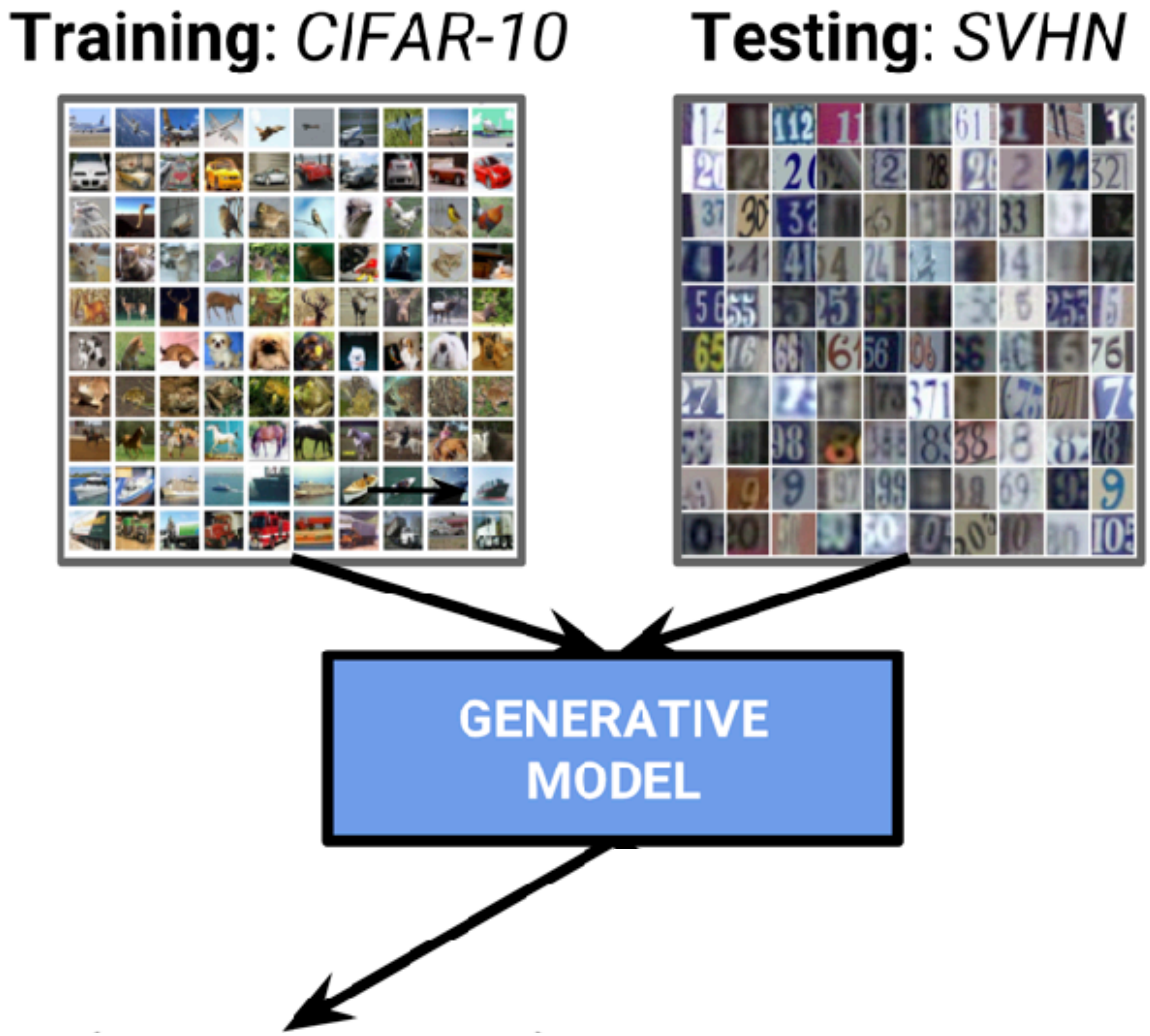


CHICKEN OR SEVEN?



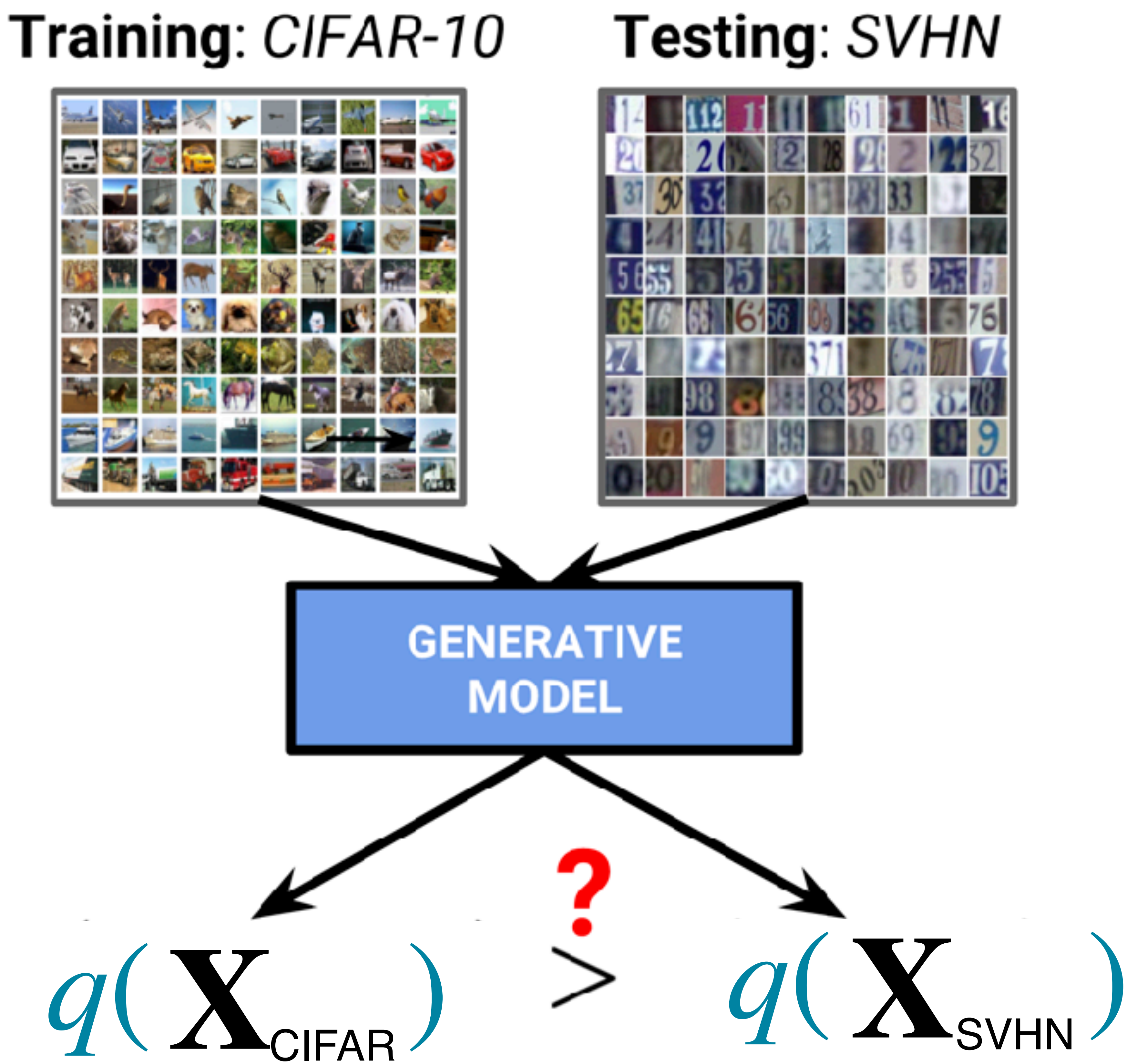
$$q(\mathbf{X}_{\text{CIFAR}})$$

CHICKEN OR SEVEN?

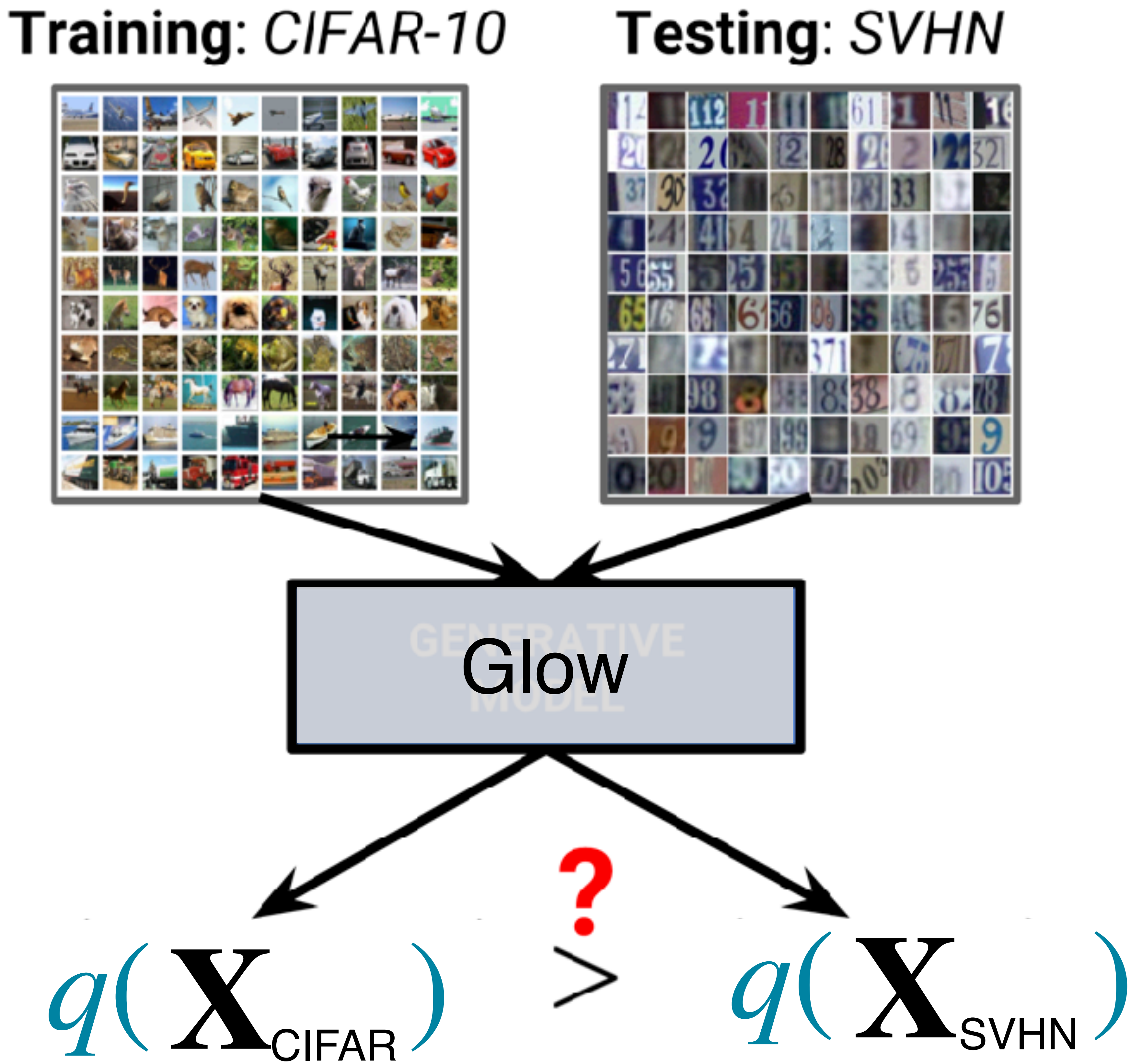


$$q(\mathbf{X}_{\text{CIFAR}})$$

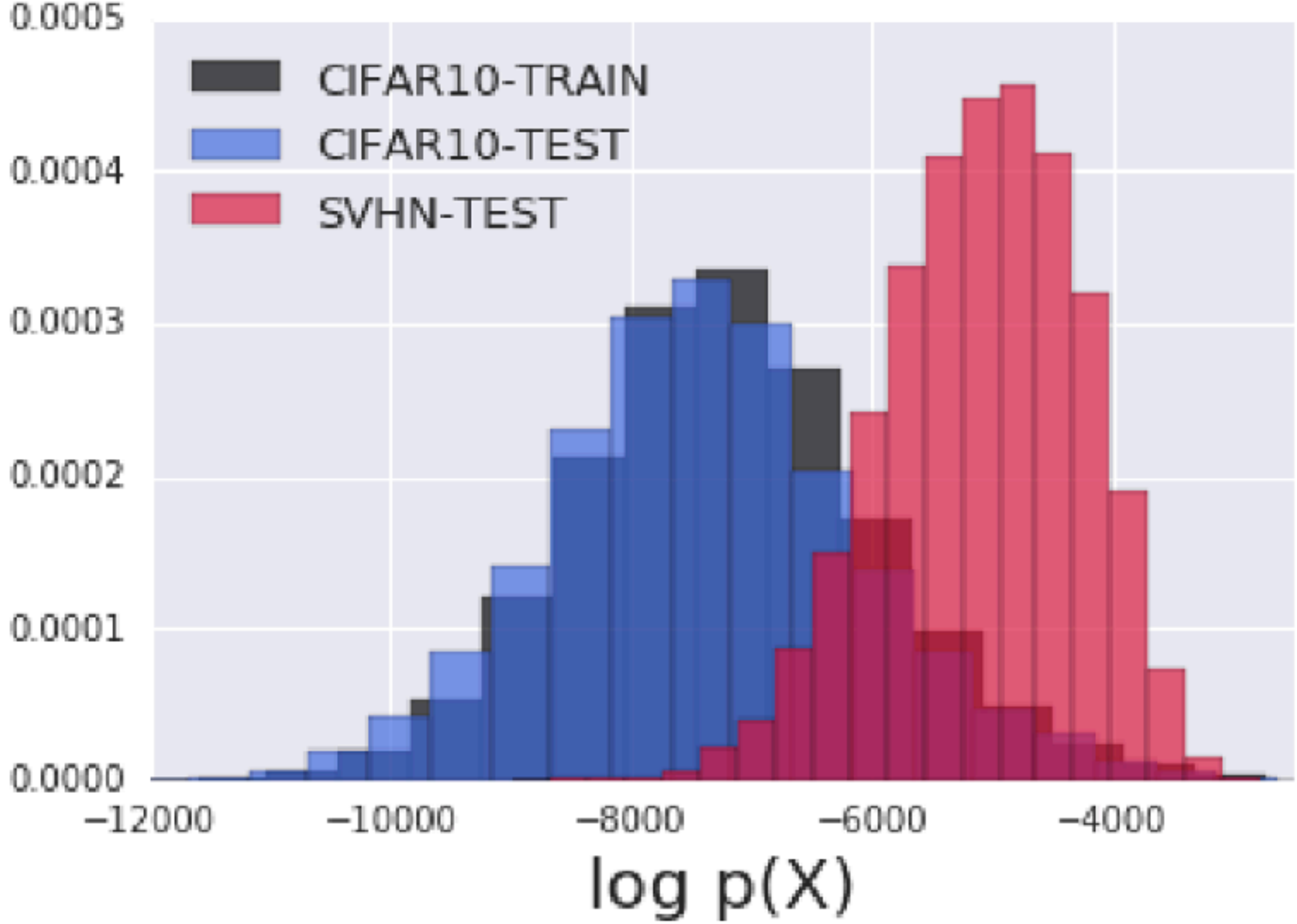
CHICKEN OR SEVEN?



CHICKEN OR SEVEN?



CIFAR-10 VS SVHN



[Nalisnick et al., ICLR 2019]

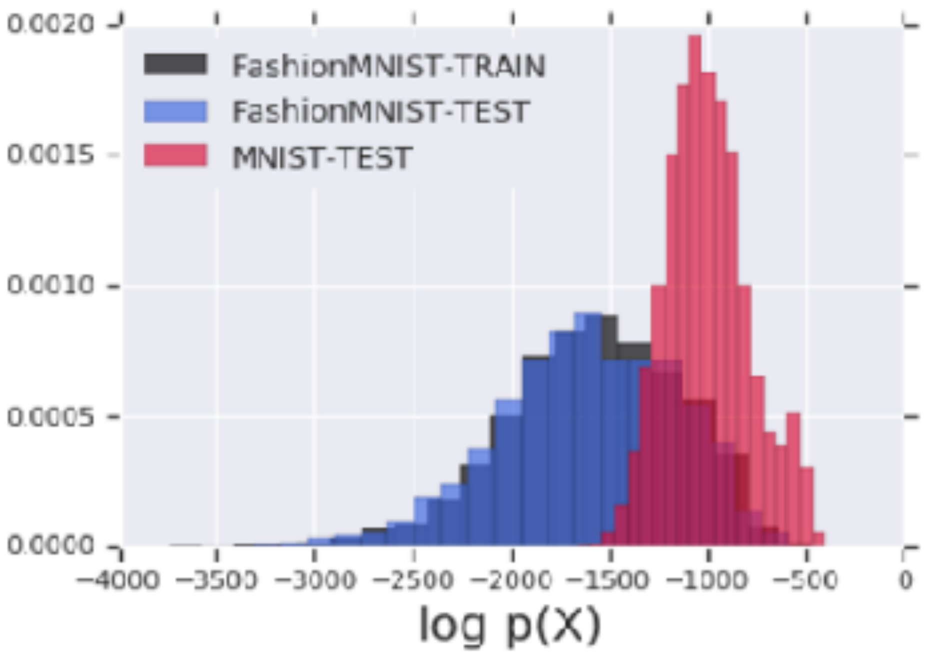
GLOW: ADDITIONAL DATA SETS

FashionMNIST vs MNIST

CelebA vs SVHN

ImageNet vs CIFAR-10 vs SVHN

GLOW: ADDITIONAL DATA SETS

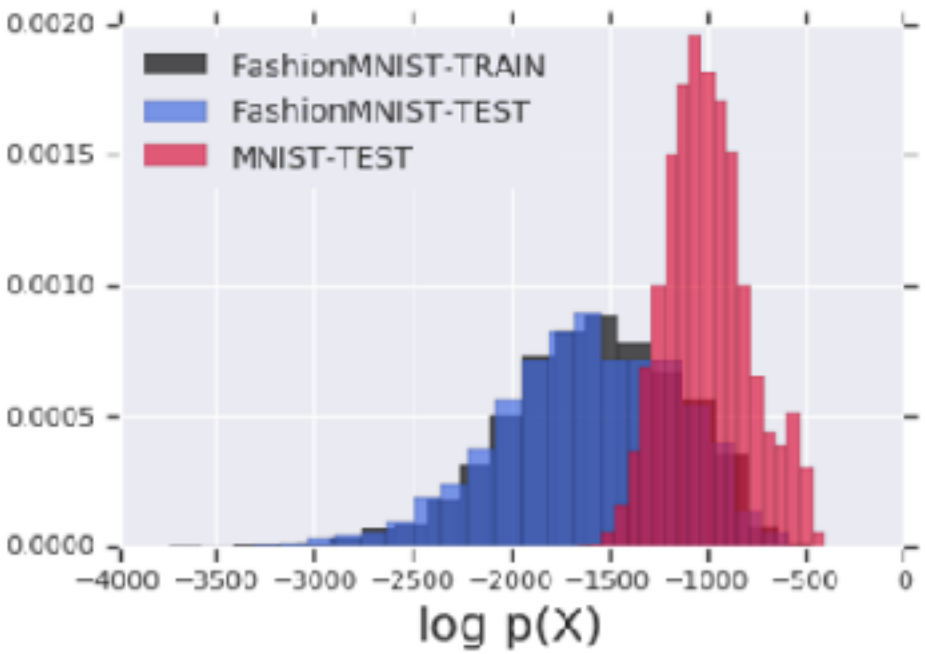


FashionMNIST vs MNIST

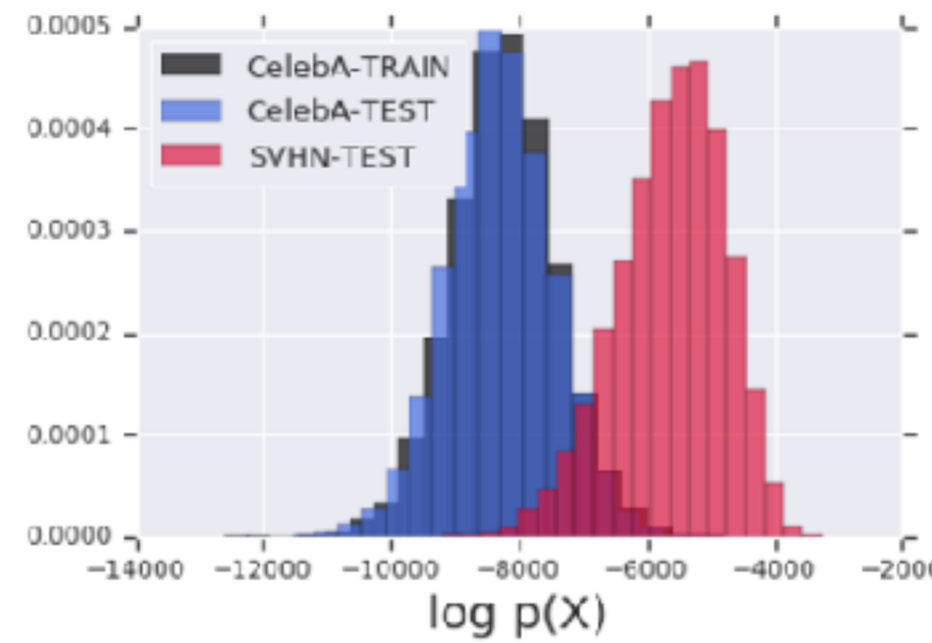
CelebA vs SVHN

ImageNet vs CIFAR-10 vs SVHN

GLOW: ADDITIONAL DATA SETS



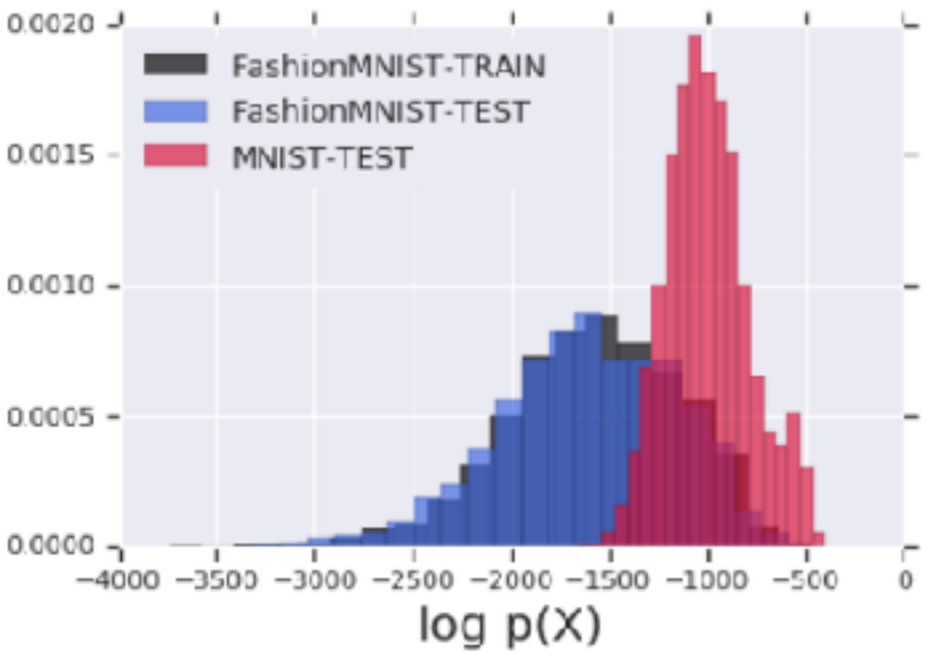
FashionMNIST vs MNIST



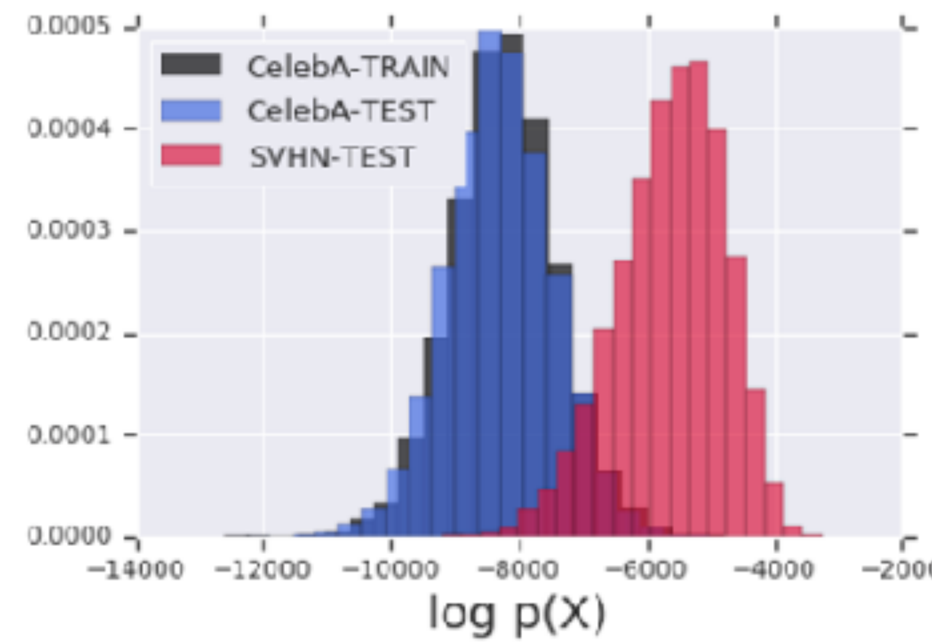
CelebA vs SVHN

ImageNet vs CIFAR-10 vs SVHN

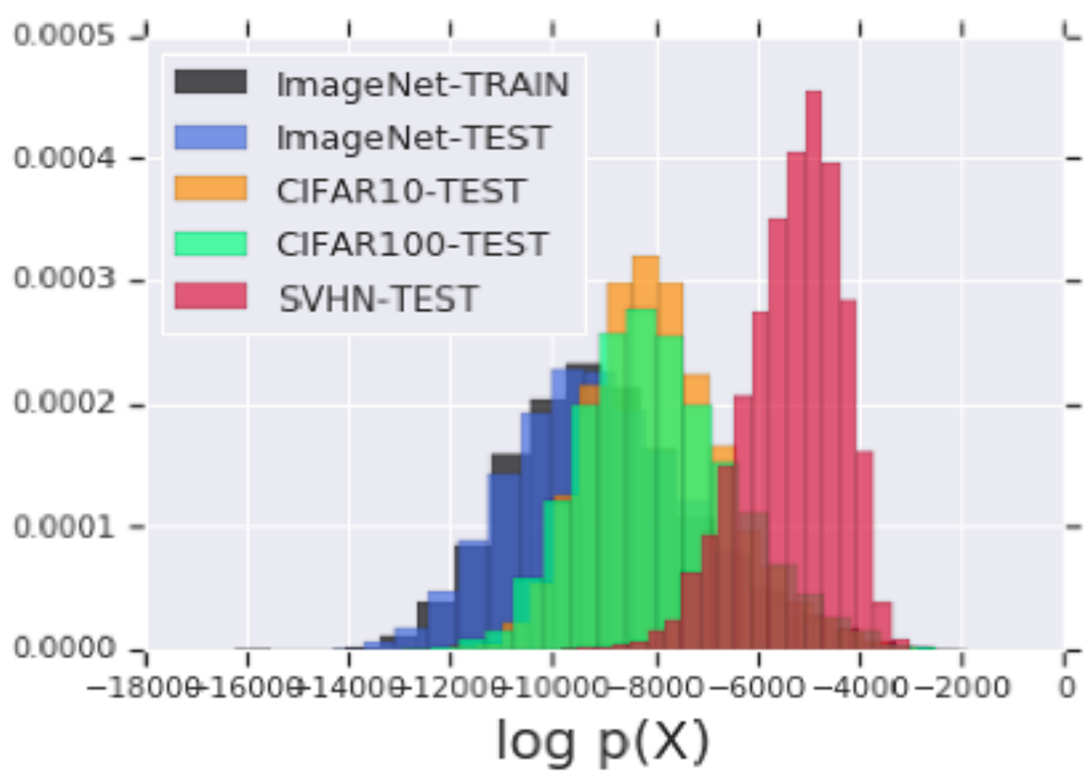
GLOW: ADDITIONAL DATA SETS



FashionMNIST vs MNIST

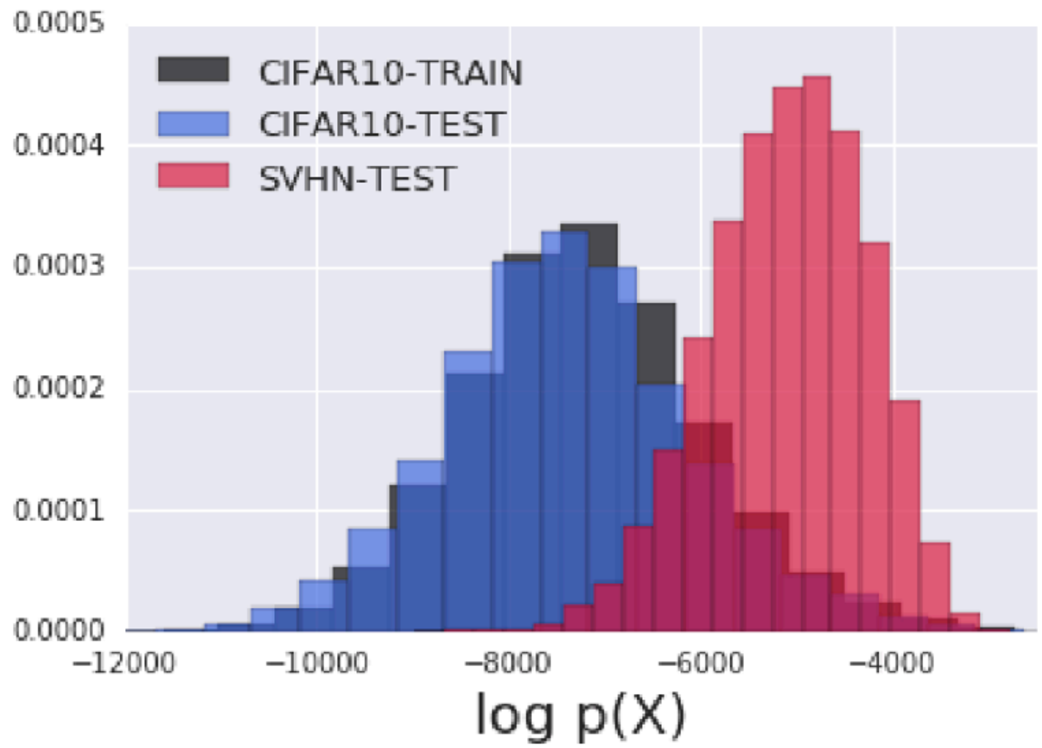


CelebA vs SVHN

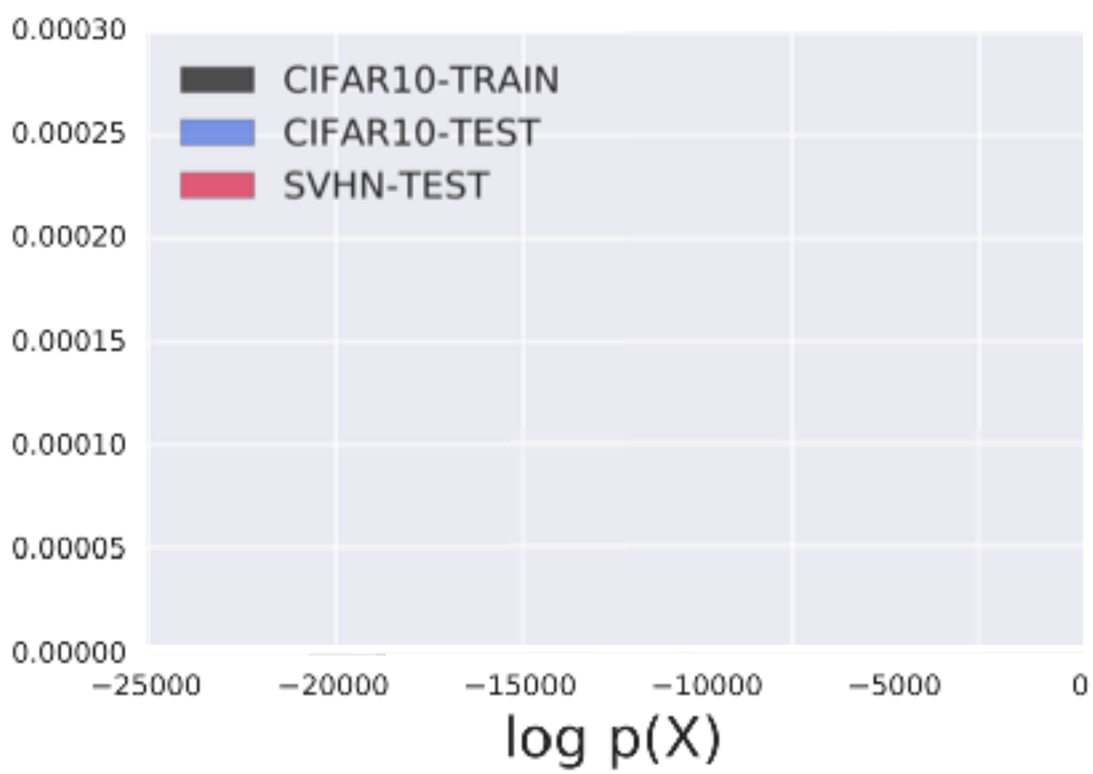


ImageNet vs CIFAR-10 vs SVHN

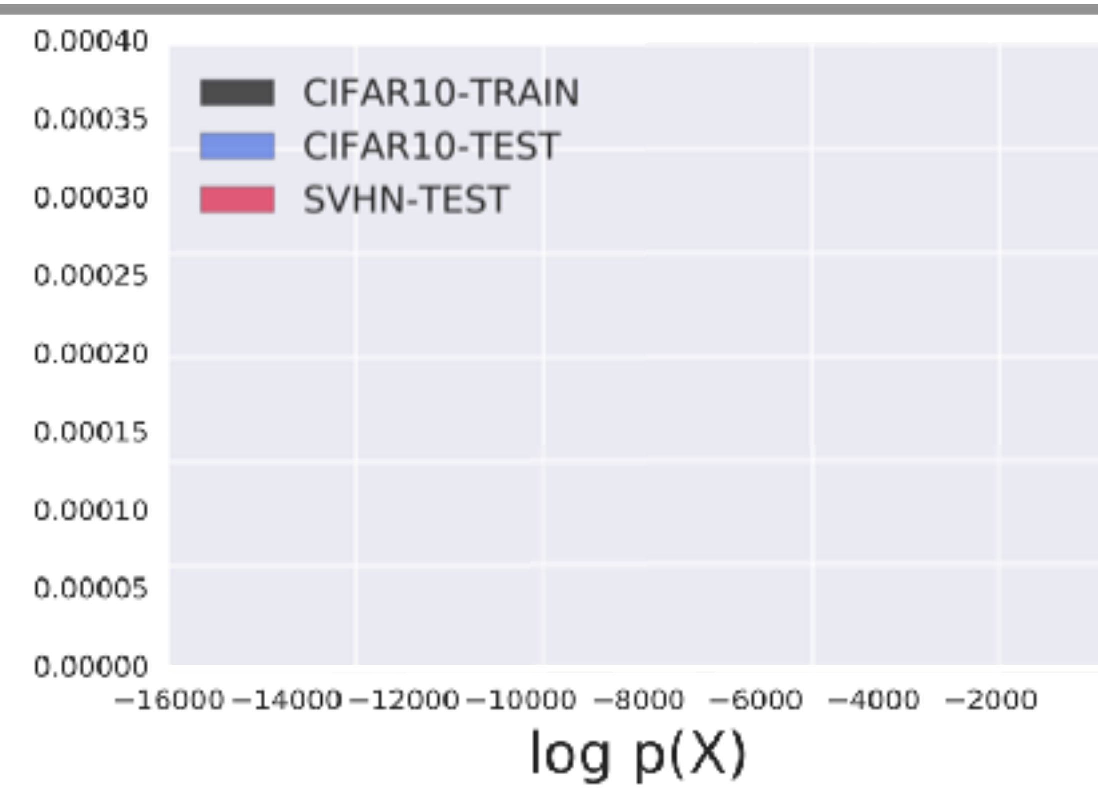
OTHER MODELS: CIFAR-10 VS SVHN



Normalizing Flow
(Glow)

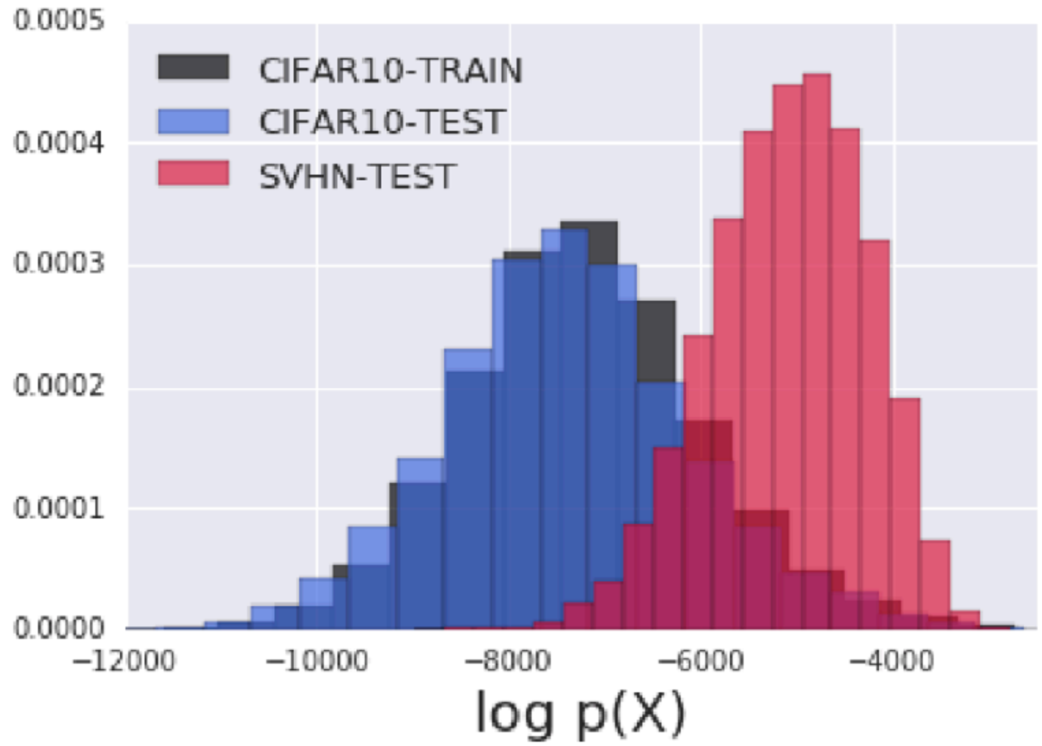


Variational Autoencoder

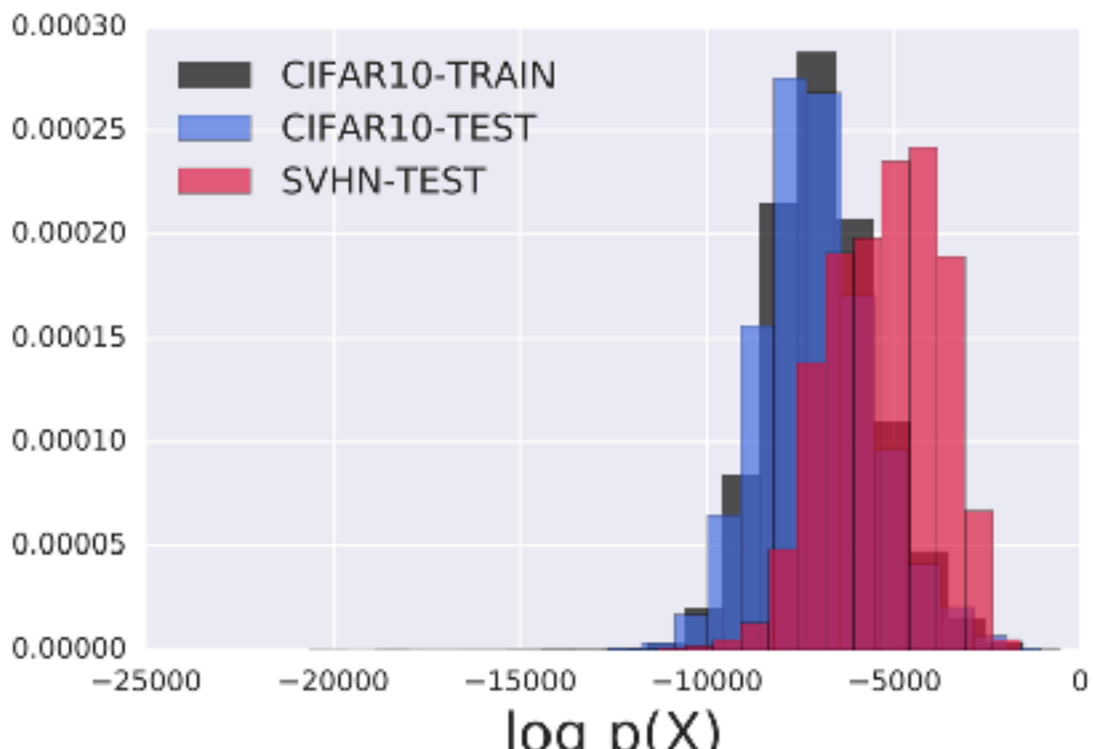


CNN Autoregressive Model

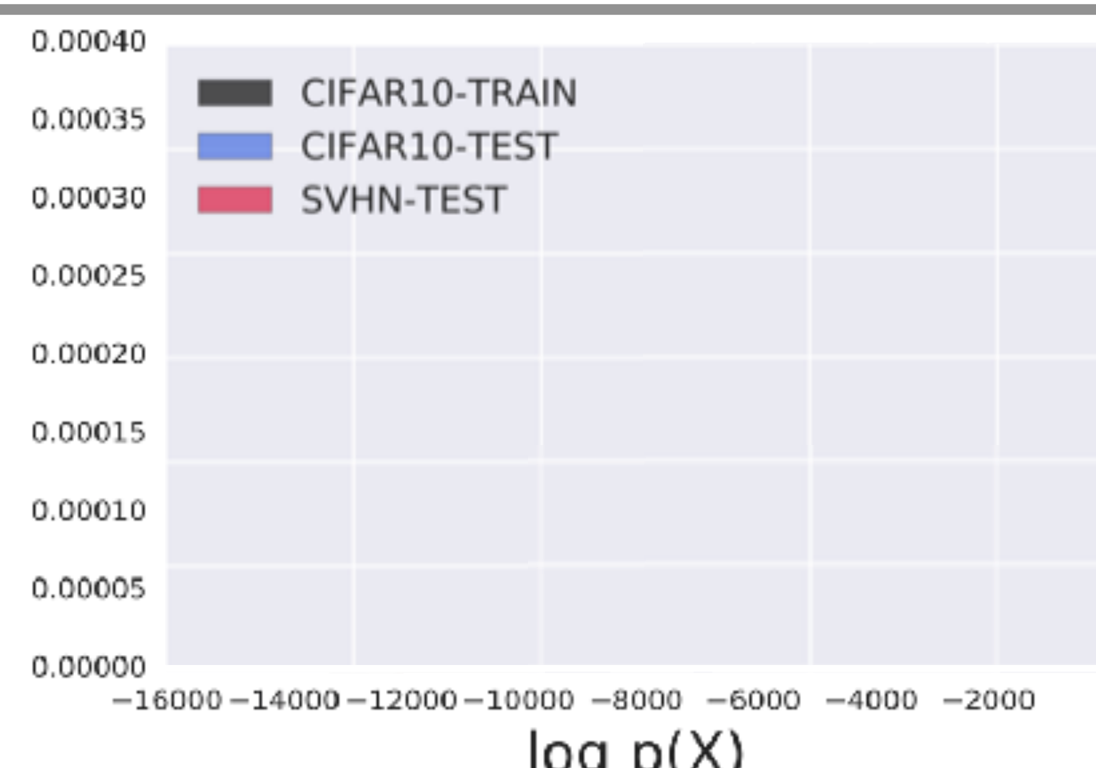
OTHER MODELS: CIFAR-10 VS SVHN



Normalizing Flow
(Glow)

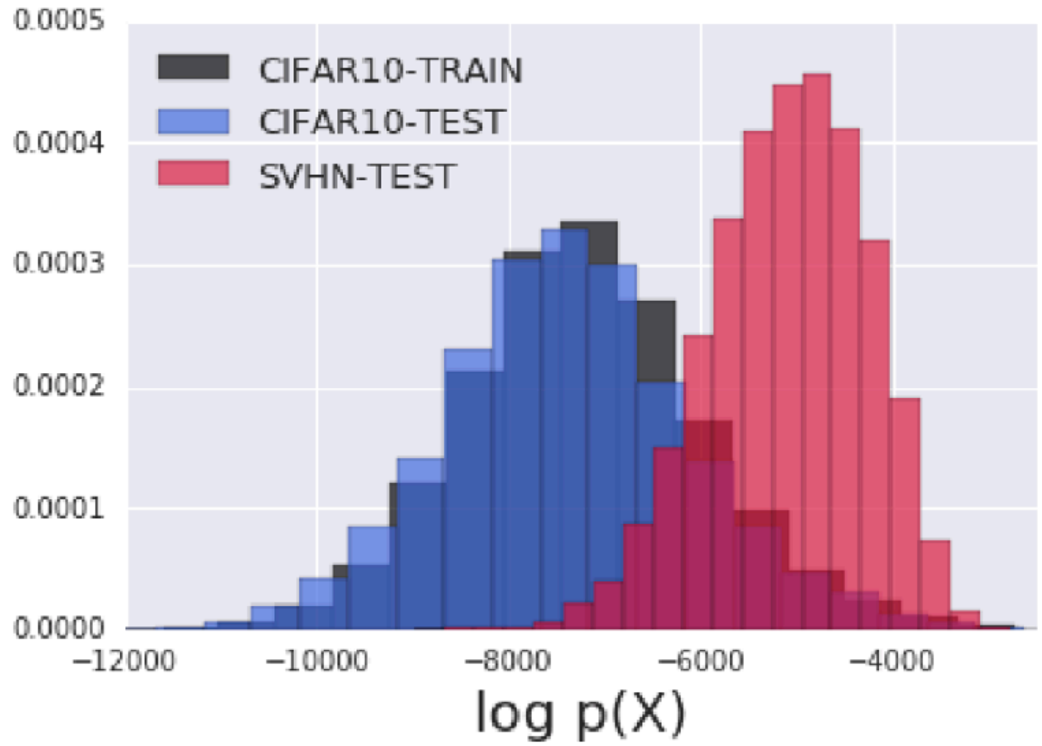


Variational Autoencoder

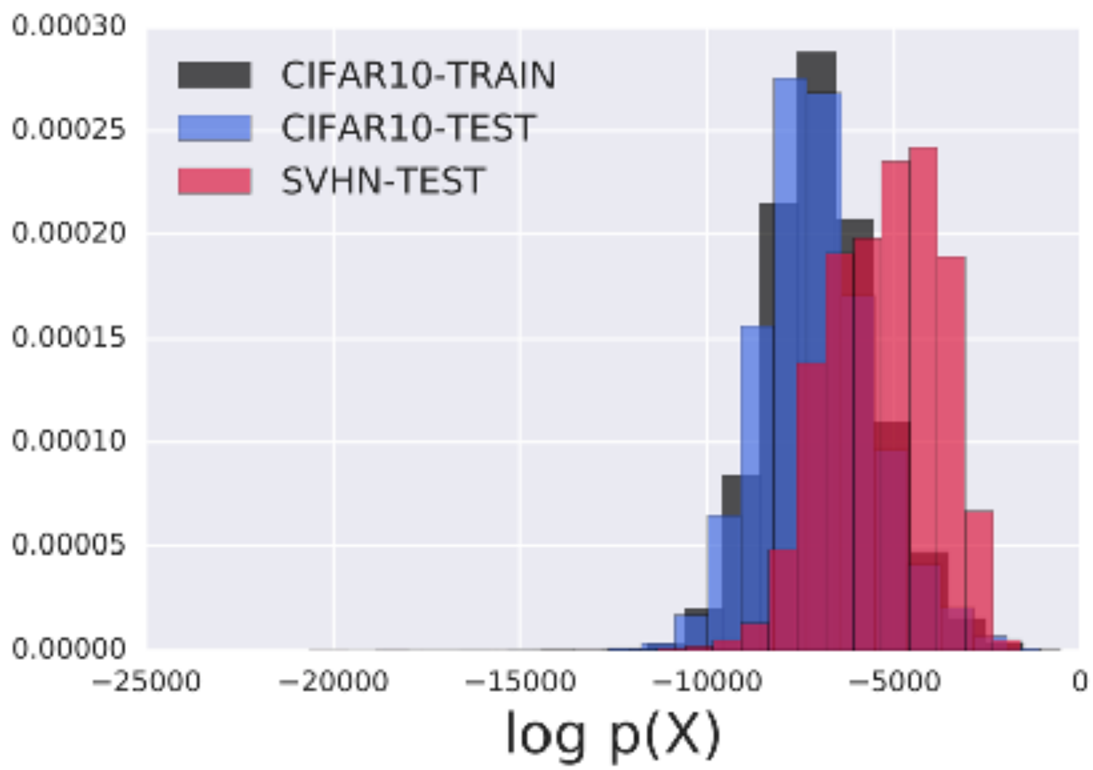


CNN Autoregressive Model

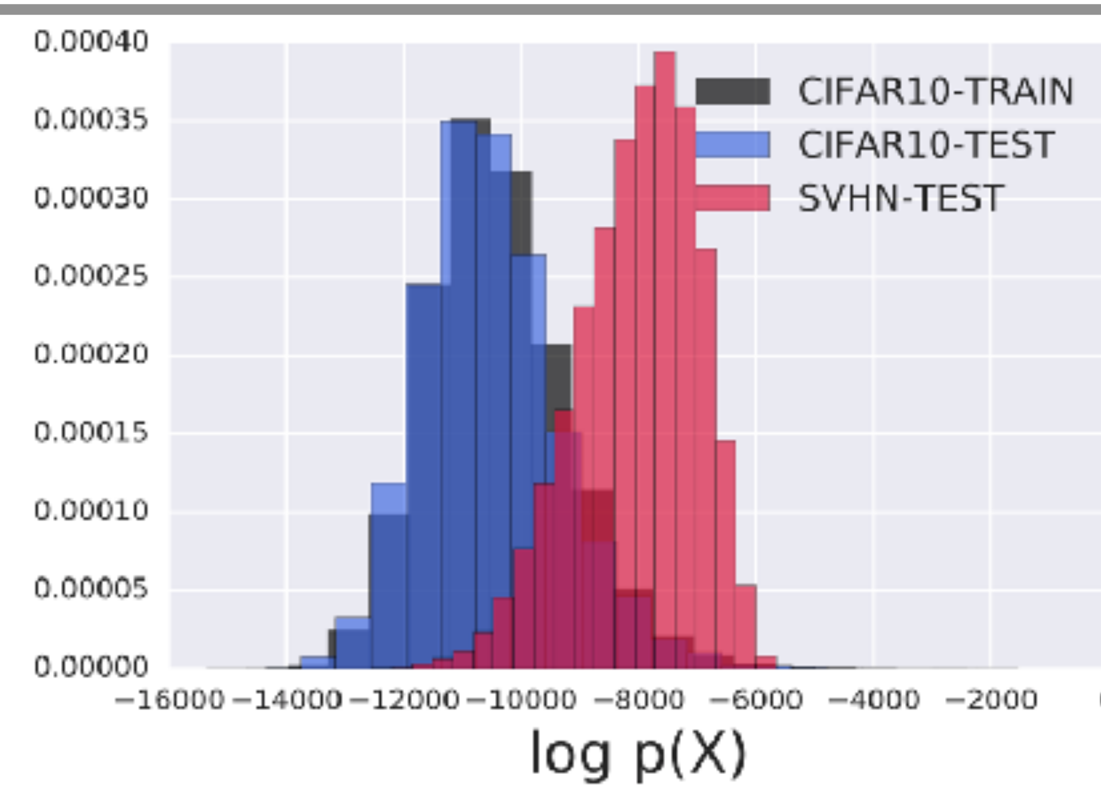
OTHER MODELS: CIFAR-10 VS SVHN



Normalizing Flow (Glow)



Variational Autoencoder



CNN Autoregressive Model

What’s going on here?

Consider a Bayes classifier for out-of-distribution (OOD) detection:

$C = \{ \text{IN, OUT} \}$

$$p(C \mid \mathbf{X}^*) = \frac{p(\mathbf{X}^* \mid C)\; p(C)}{p(\mathbf{X}^*)}$$

Consider a Bayes classifier for out-of-distribution (OOD) detection:

$$C = \{\text{ IN, OUT }\}$$

$$p(C \mid \mathbf{X}^*) = \frac{p(\mathbf{X}^* \mid C)\ p(C)}{p(\mathbf{X}^*)}$$

After some algebraic rearrangement, we have the decision rule:

$$p(\mathbf{X}^* \mid \text{ IN }) > \frac{p(\mathbf{X}^* \mid \text{OUT})\ p(\text{OUT})}{p(\text{ IN })}$$

Decision rule:

$$p(\mathbf{X}^* \mid \text{IN}) > \frac{p(\mathbf{X}^* \mid \text{OUT})\; p(\text{OUT})}{p(\text{IN})}$$

Decision rule:

$$\overset{\text{DGM}}{q(\mathbf{X}^*)} > \frac{p(\mathbf{X}^* \mid \text{OUT})\, p(\text{OUT})}{p(\text{IN})}$$

Decision rule:

DGM

$$q(\mathbf{X}^*) > \frac{?\; p(\text{OUT})}{p(\text{IN})}$$

Decision rule:

DGM

$$q(\mathbf{X}^*) > \frac{? \; p(\text{OUT})}{p(\text{IN})}$$

Novelty Detection and Neural Network Validation

Chris M. Bishop (May 1994)

Decision rule:

$$\overset{\text{DGM}}{q(\mathbf{X}^*)} > \frac{\text{UNIFORM}(\mathbf{X}^*)\; p(\text{OUT})}{p(\text{IN})}$$

Novelty Detection and Neural Network Validation

Chris M. Bishop (May 1994)

Decision rule:

DGM

$$\underbrace{q(\mathbf{X}^*)}_{} > \underbrace{\frac{\text{UNIFORM}(\mathbf{X}^*)\; p(\text{OUT})}{p(\text{IN})}}_{\hat{\tau}}$$

Decision rule:

$$\overset{\text{DGM}}{q(\mathbf{X}^*)} > \underbrace{\frac{\text{UNIFORM}(\mathbf{X}^*)\; p(\text{OUT})}{p(\text{IN})}}_{\hat{\tau}}$$

Implies classifier is just a threshold on the density function:

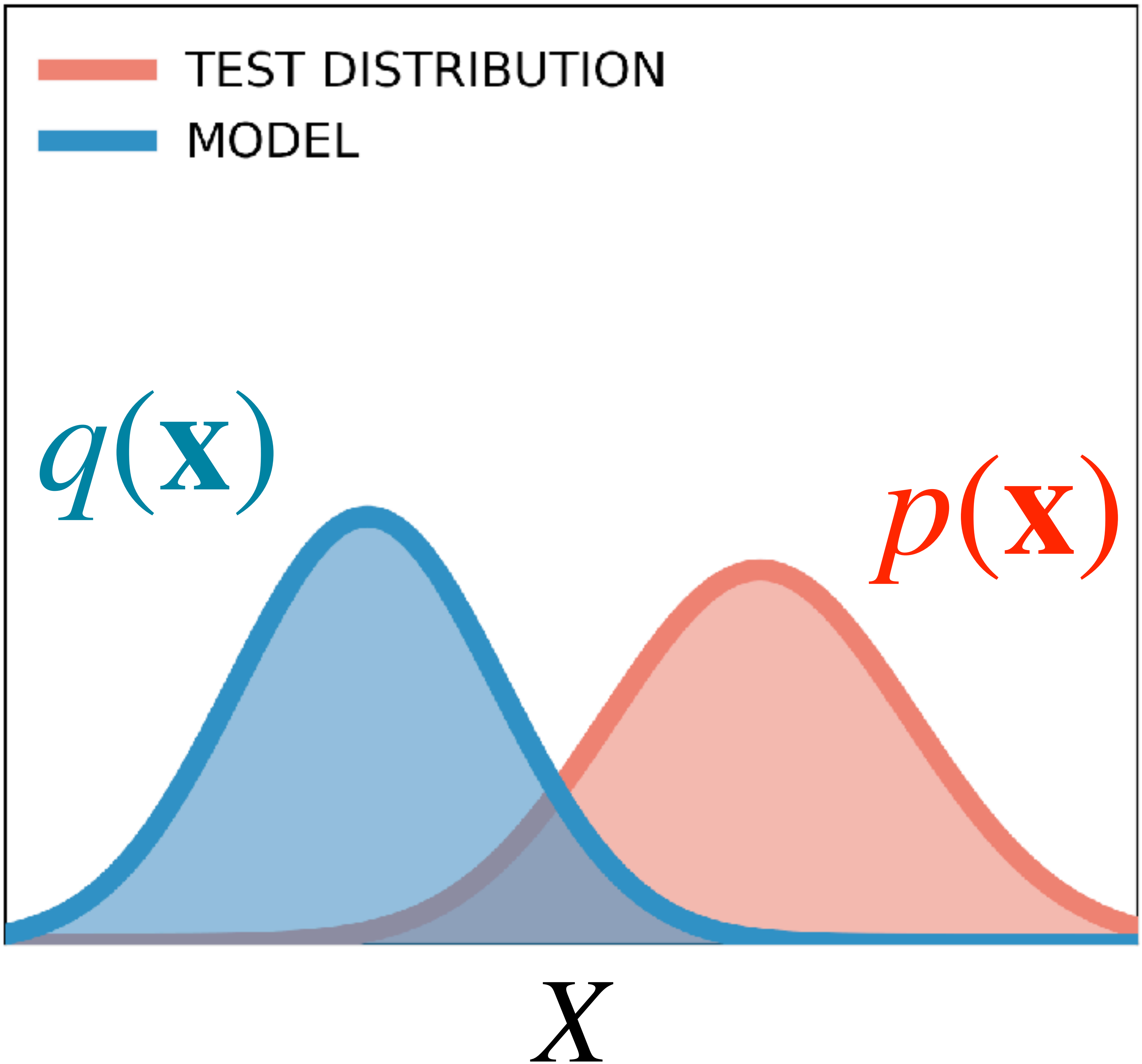
$$q(\mathbf{X}^*) > \hat{\tau}$$

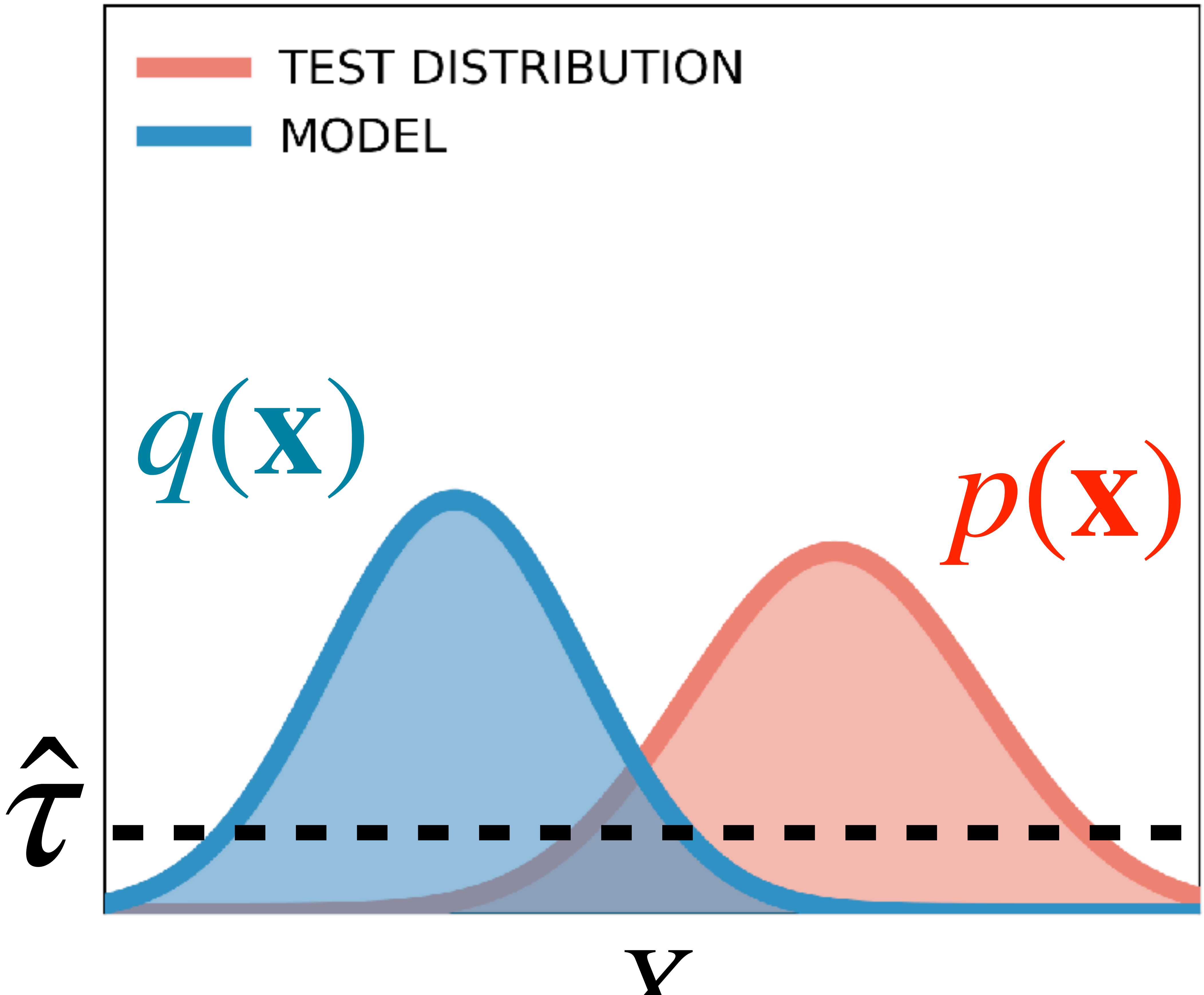
Decision rule:

$$\overset{\text{DGM}}{q(\mathbf{X}^*)} > \underbrace{\frac{\text{UNIFORM}(\mathbf{X}^*)\; p(\text{OUT})}{p(\text{IN})}}_{\hat{\tau}}$$

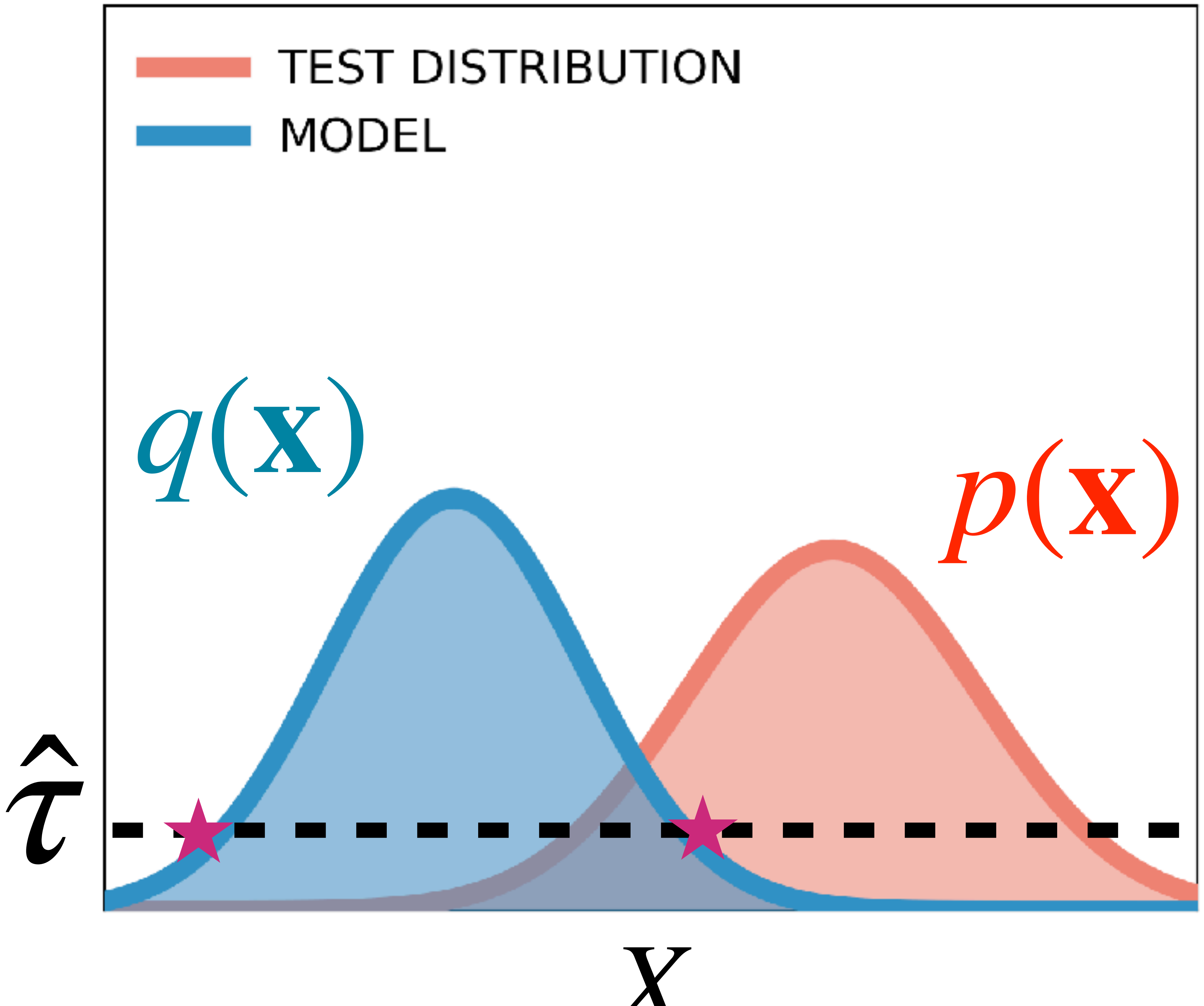
Implies classifier is just a threshold on the density function:

$$q(\mathbf{X}^*) > \hat{\tau} \implies \mathbf{X}^* \sim q(\mathbf{x})$$

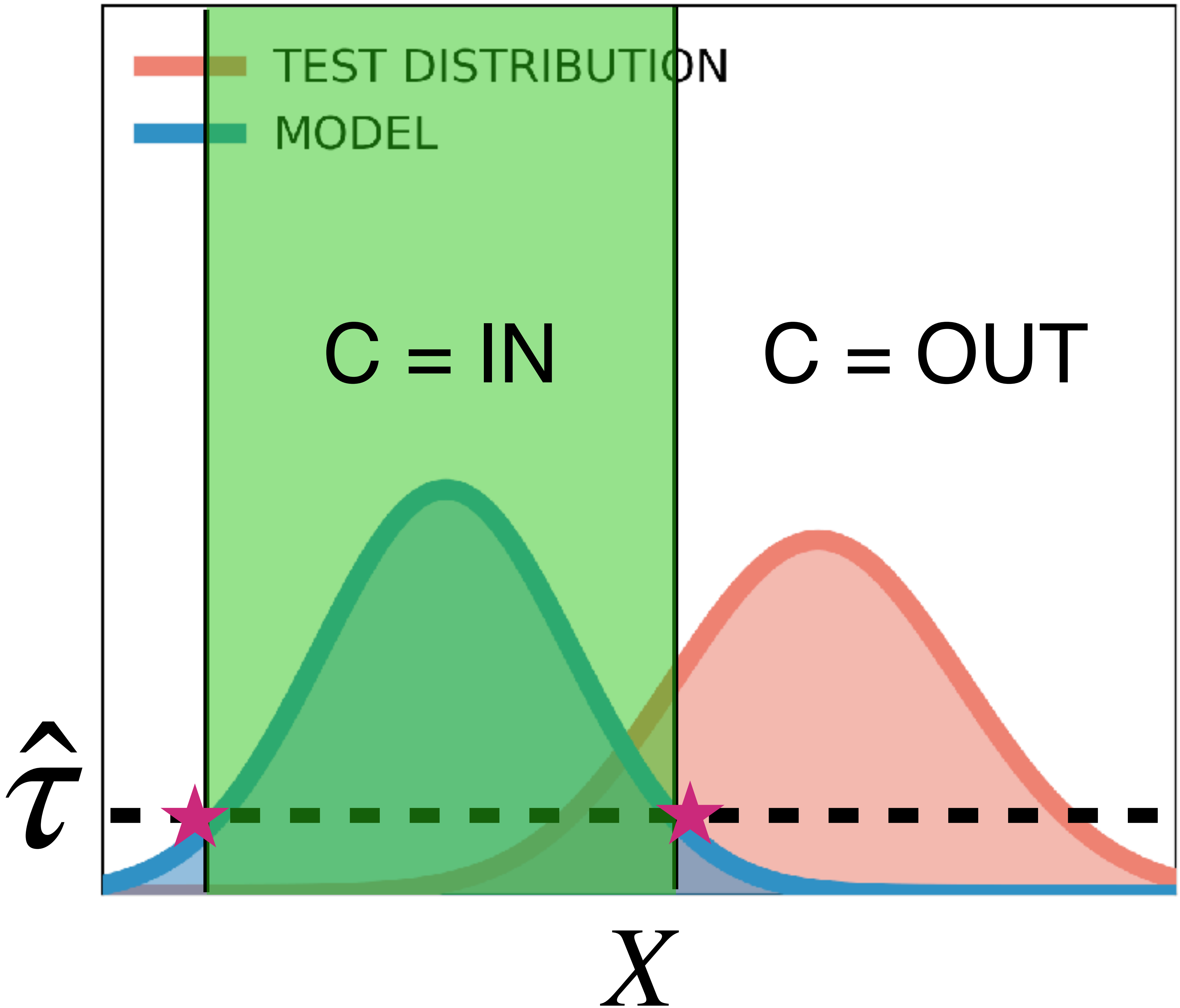

TEST DISTRIBUTION
MODEL
$q(\mathbf{x})$
$p(\mathbf{x})$
X



TEST DISTRIBUTION
MODEL
$q(\mathbf{x})$
$p(\mathbf{x})$
$\hat{\tau}$
X

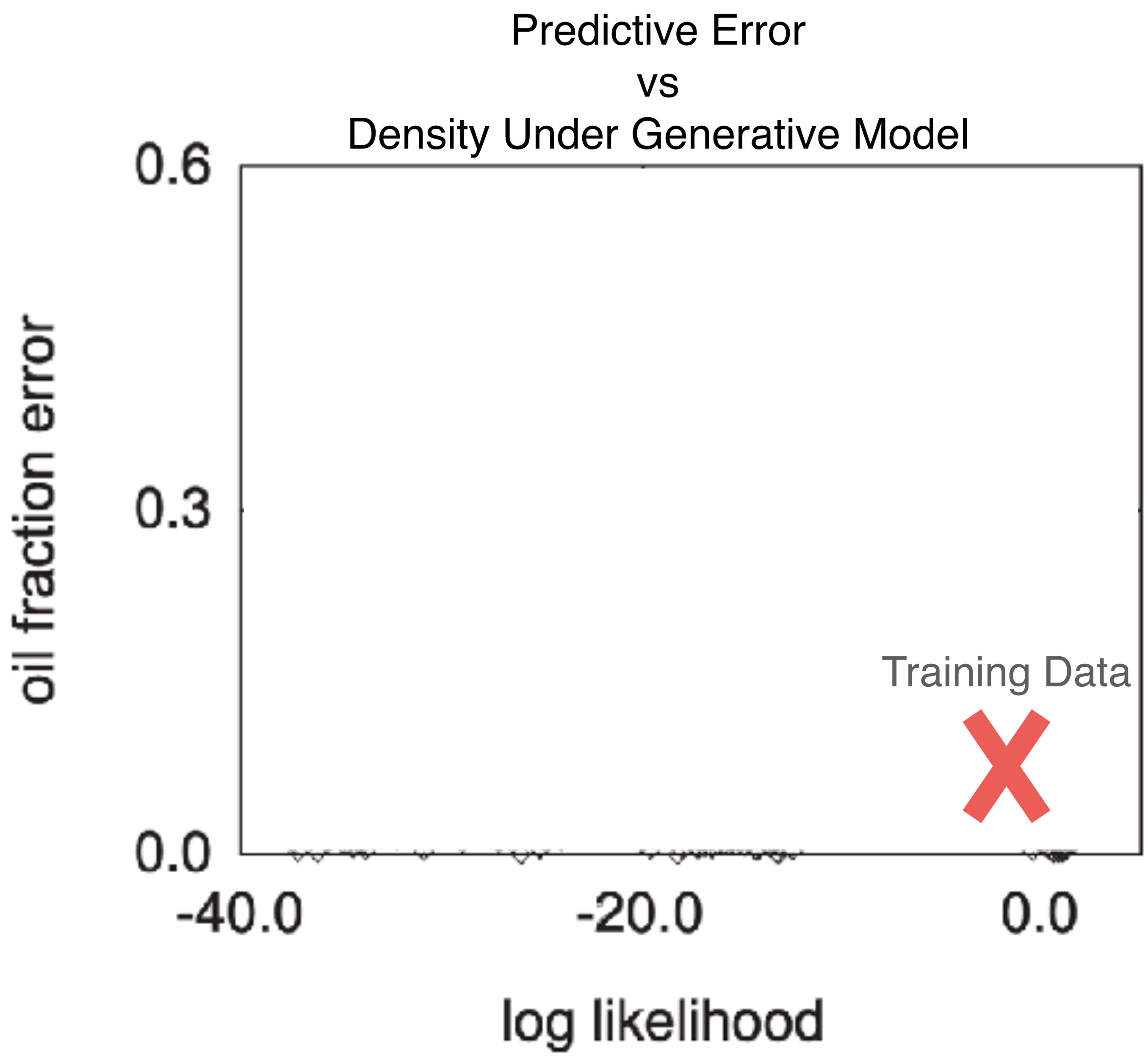


TEST DISTRIBUTION
MODEL
$q(\mathbf{x})$
$p(\mathbf{x})$
$\hat{\tau}$
X


TEST DISTRIBUTION
MODEL
C = IN
C = OUT
$\hat{\tau}$
X

Novelty Detection and Neural Network Validation

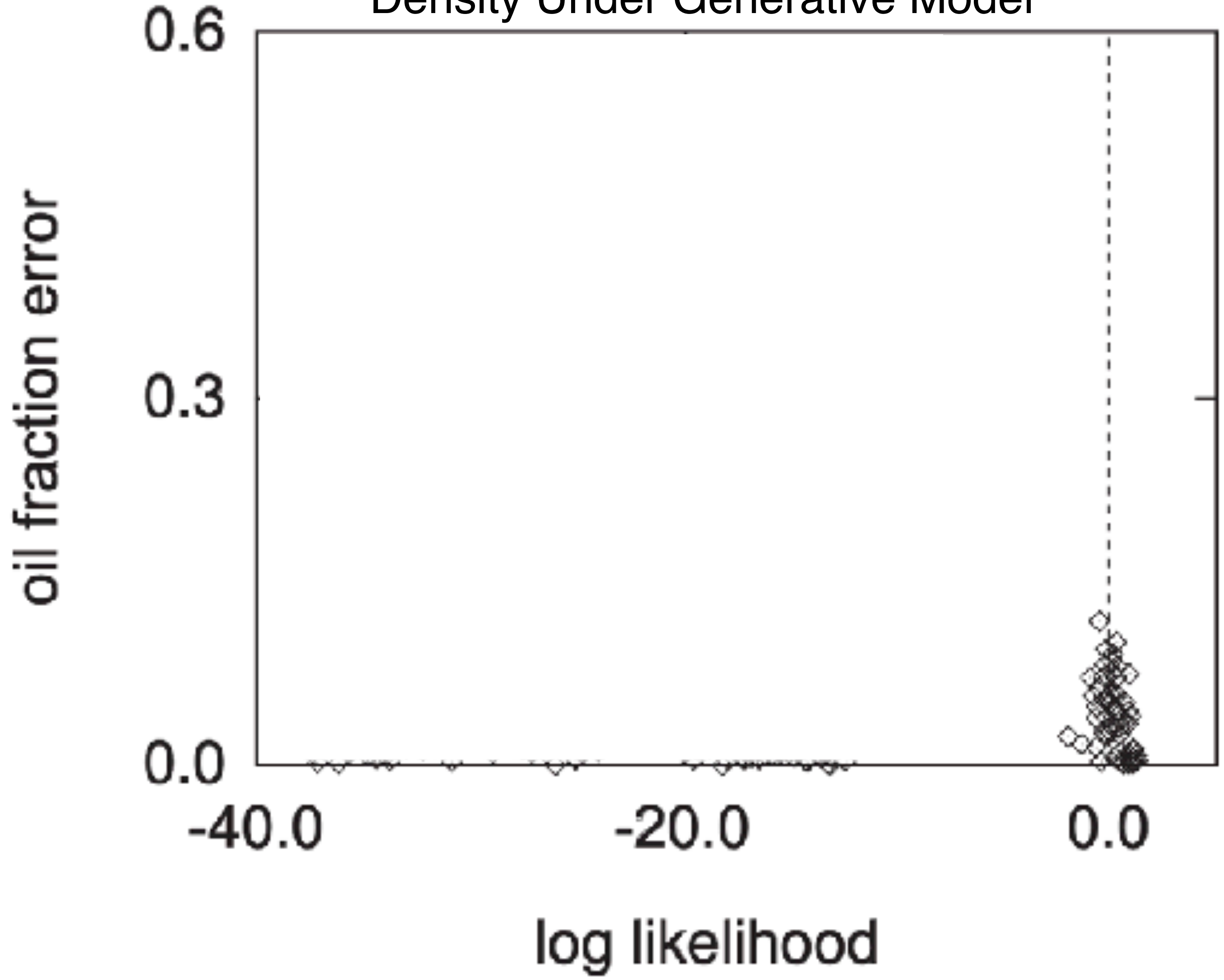
Chris M. Bishop (May 1994)



Novelty Detection and Neural Network Validation

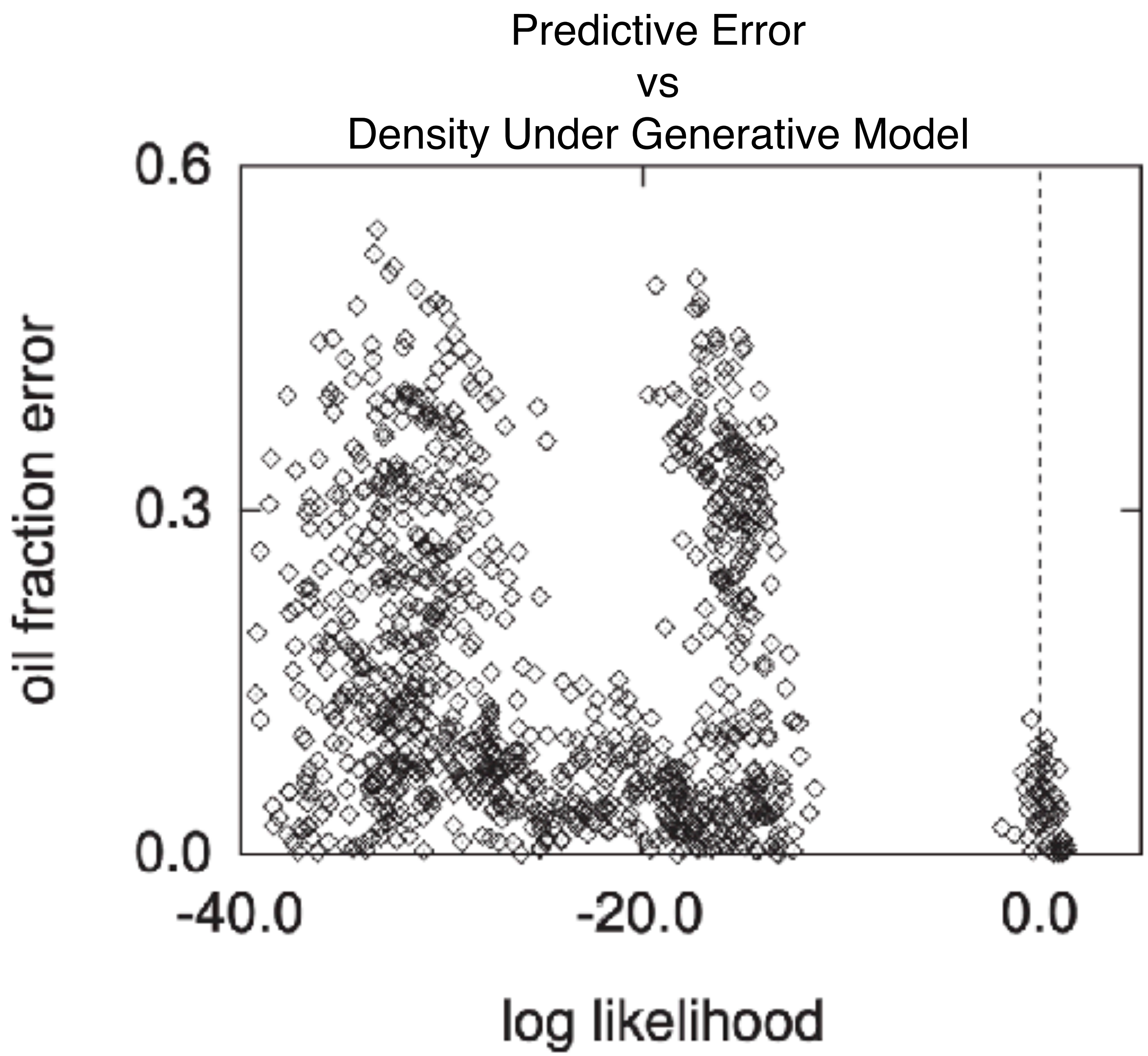
Chris M. Bishop (May 1994)

Predictive Error
vs
Density Under Generative Model



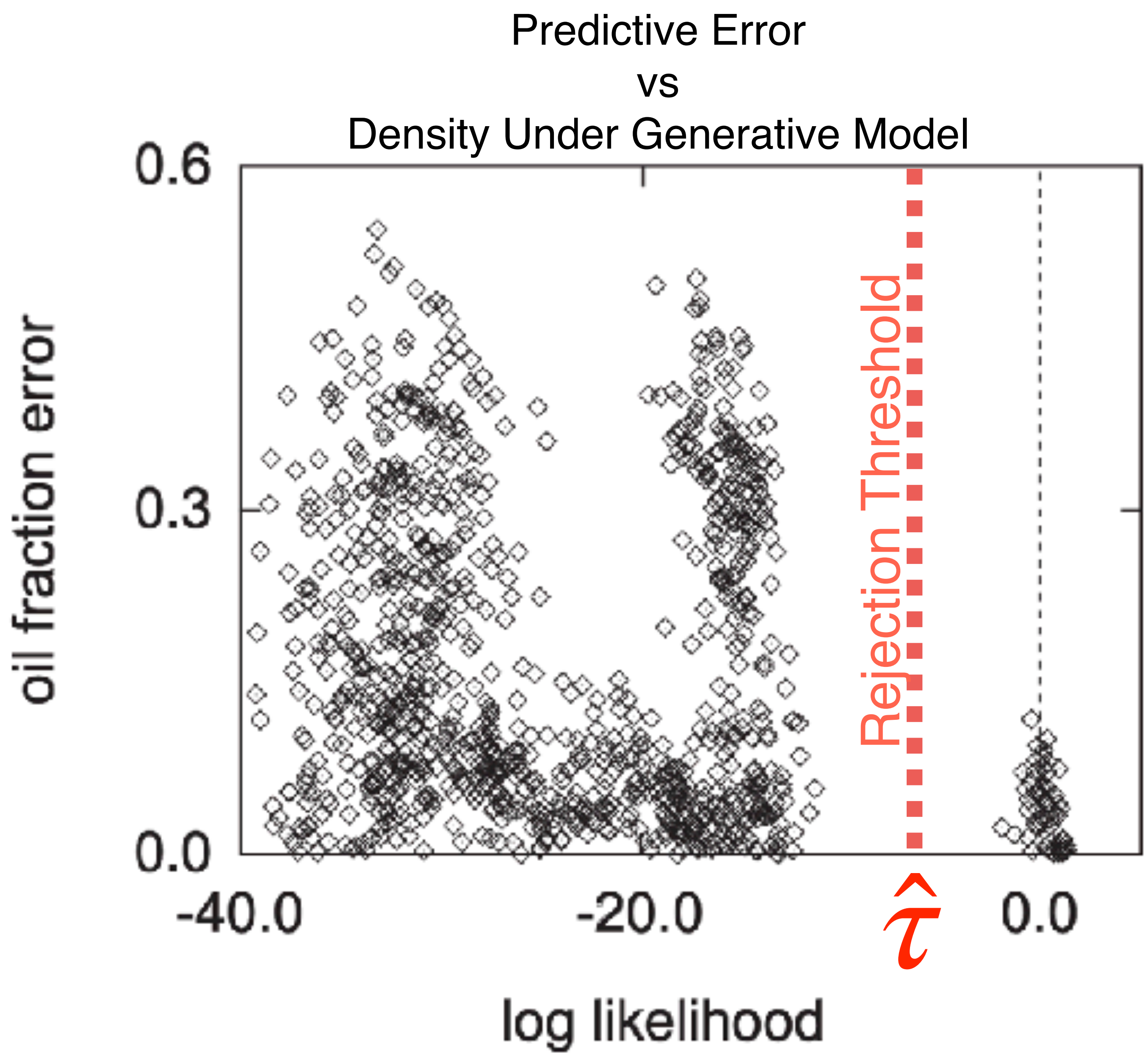
Novelty Detection and Neural Network Validation

Chris M. Bishop (May 1994)



Novelty Detection and Neural Network Validation

Chris M. Bishop (May 1994)



Why did thresholds work for C. Bishop, but not in our CIFAR-10 vs SVHN experiment?

PROBLEM: In high-dimensions, the uniform OOD model becomes degenerate.

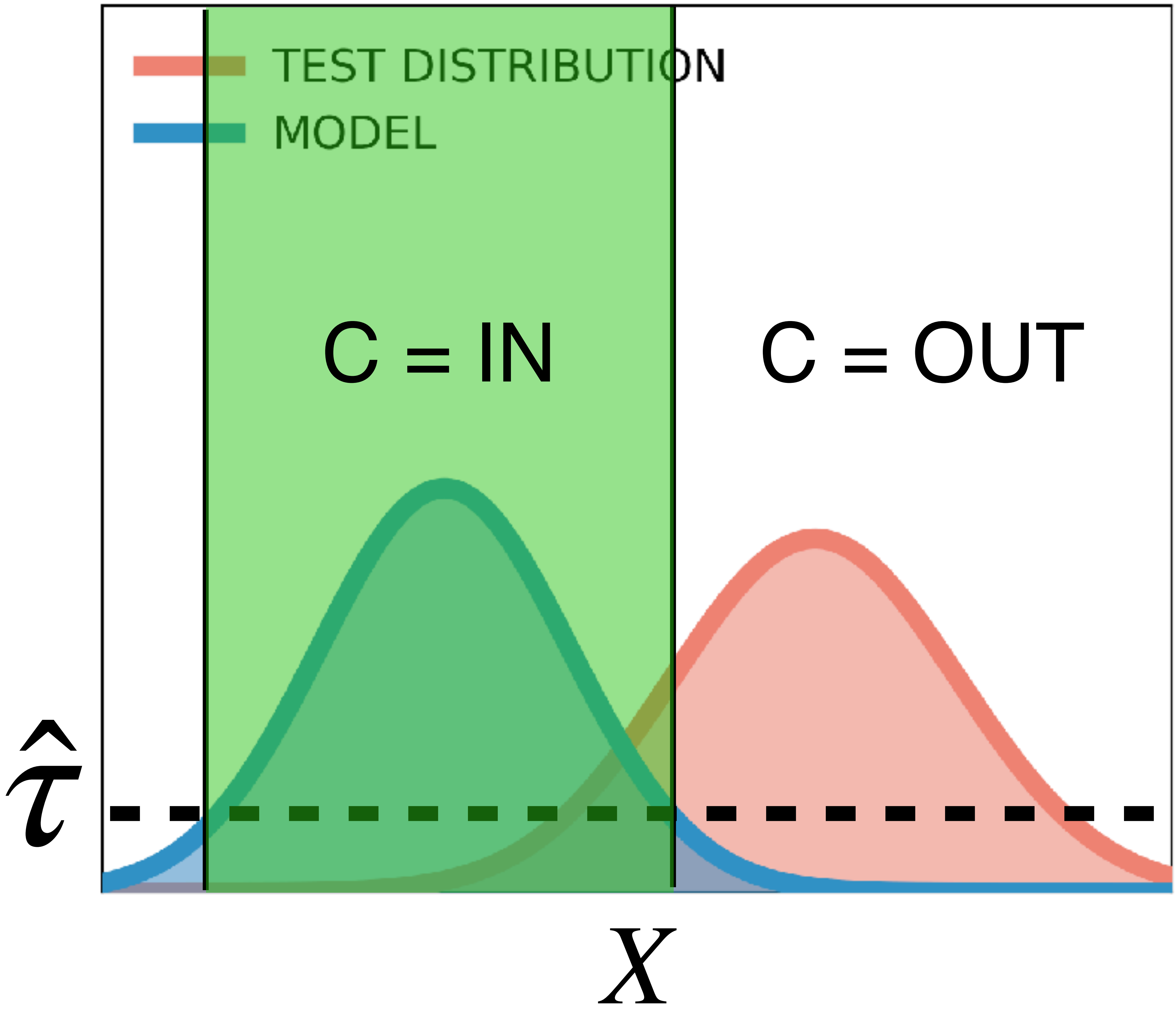
$$\text{UNIFORM}(\mathbf{x}) = \frac{1}{(b-a)^D} \;\rightarrow 0 \quad \text{as} \quad D \rightarrow \infty$$

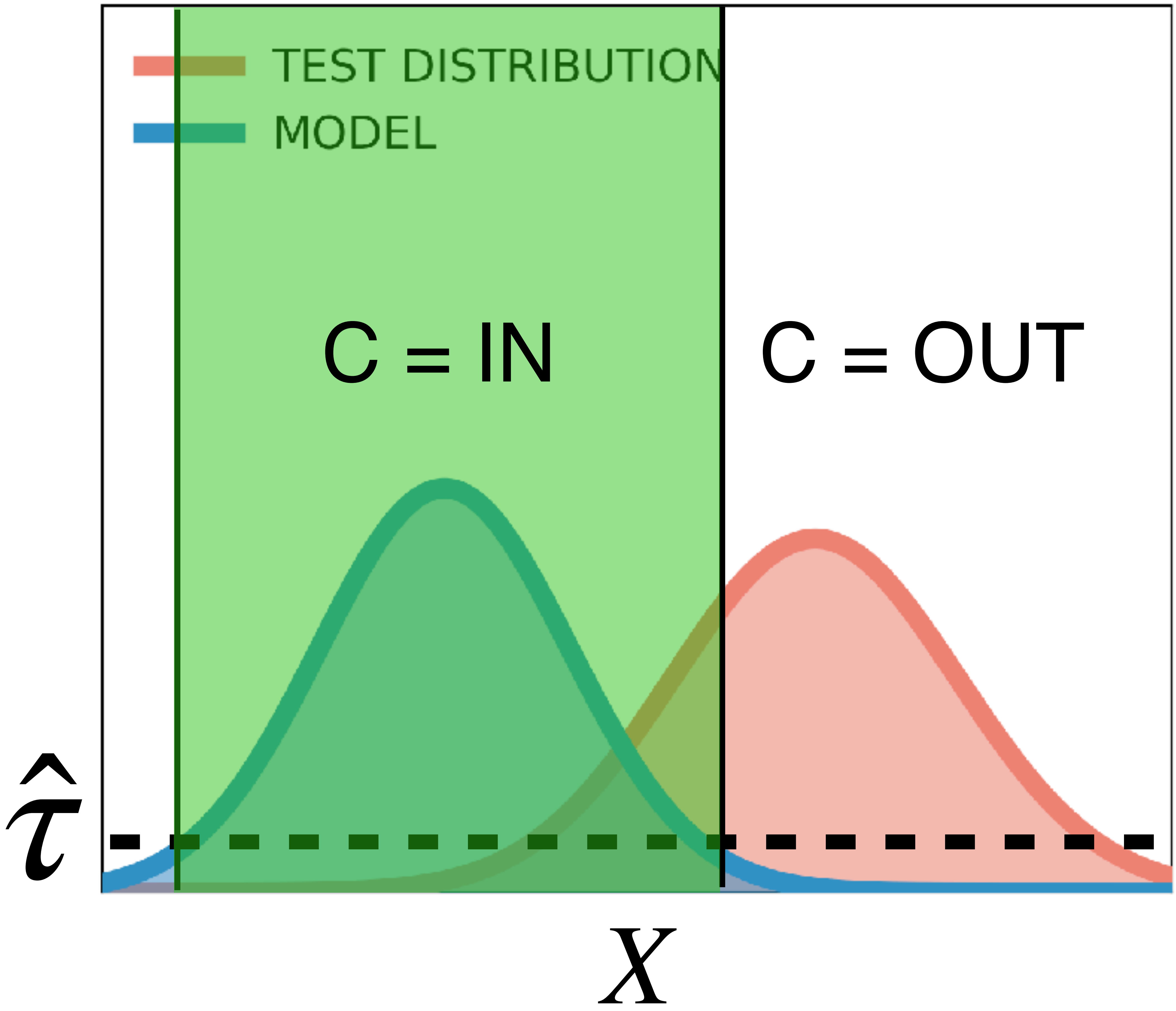
PROBLEM: In high-dimensions, the uniform OOD model becomes degenerate.

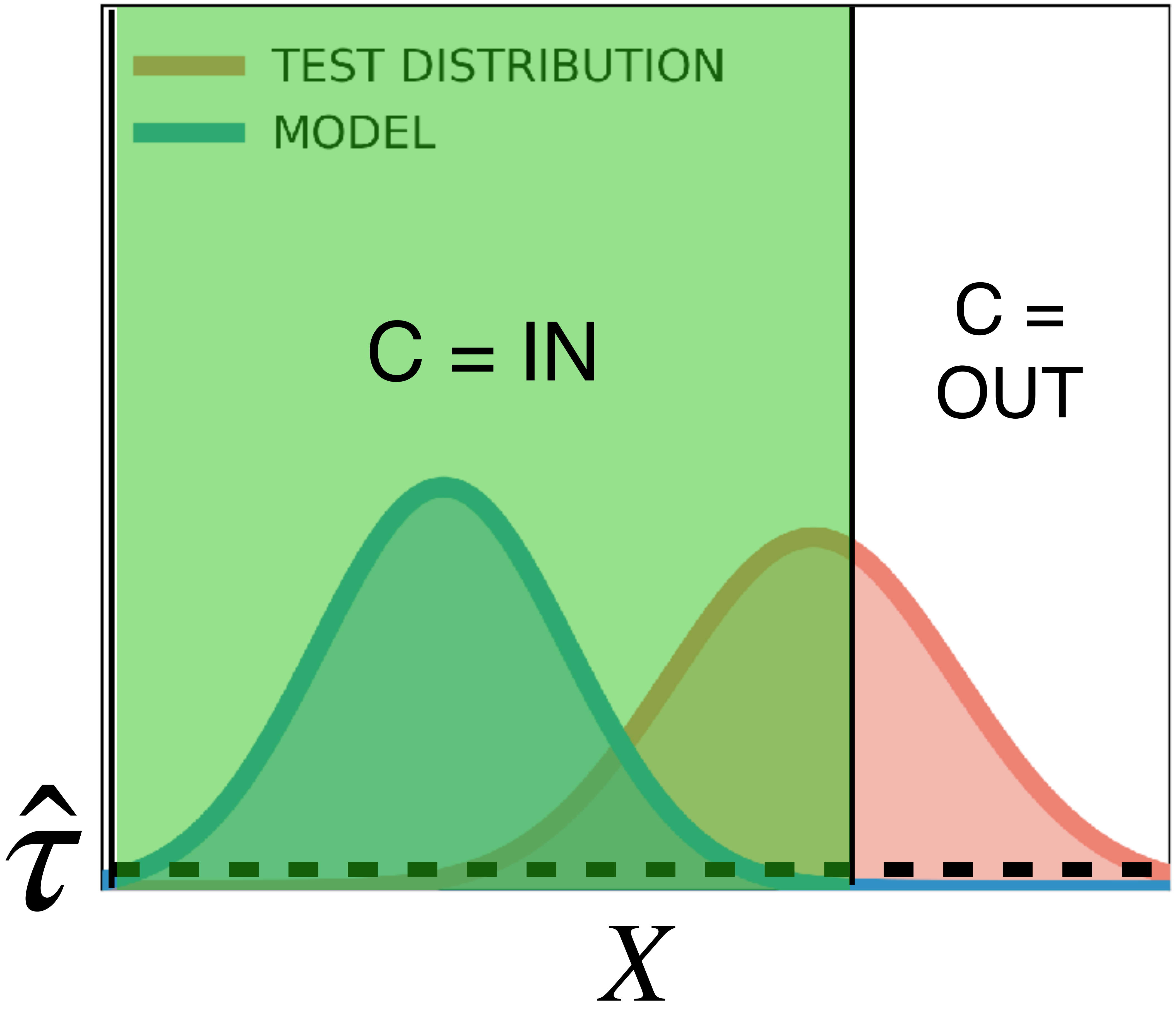
$$\text{UNIFORM}(\mathbf{x}) = \frac{1}{(b-a)^D} \rightarrow 0 \quad \text{as} \quad D \rightarrow \infty$$

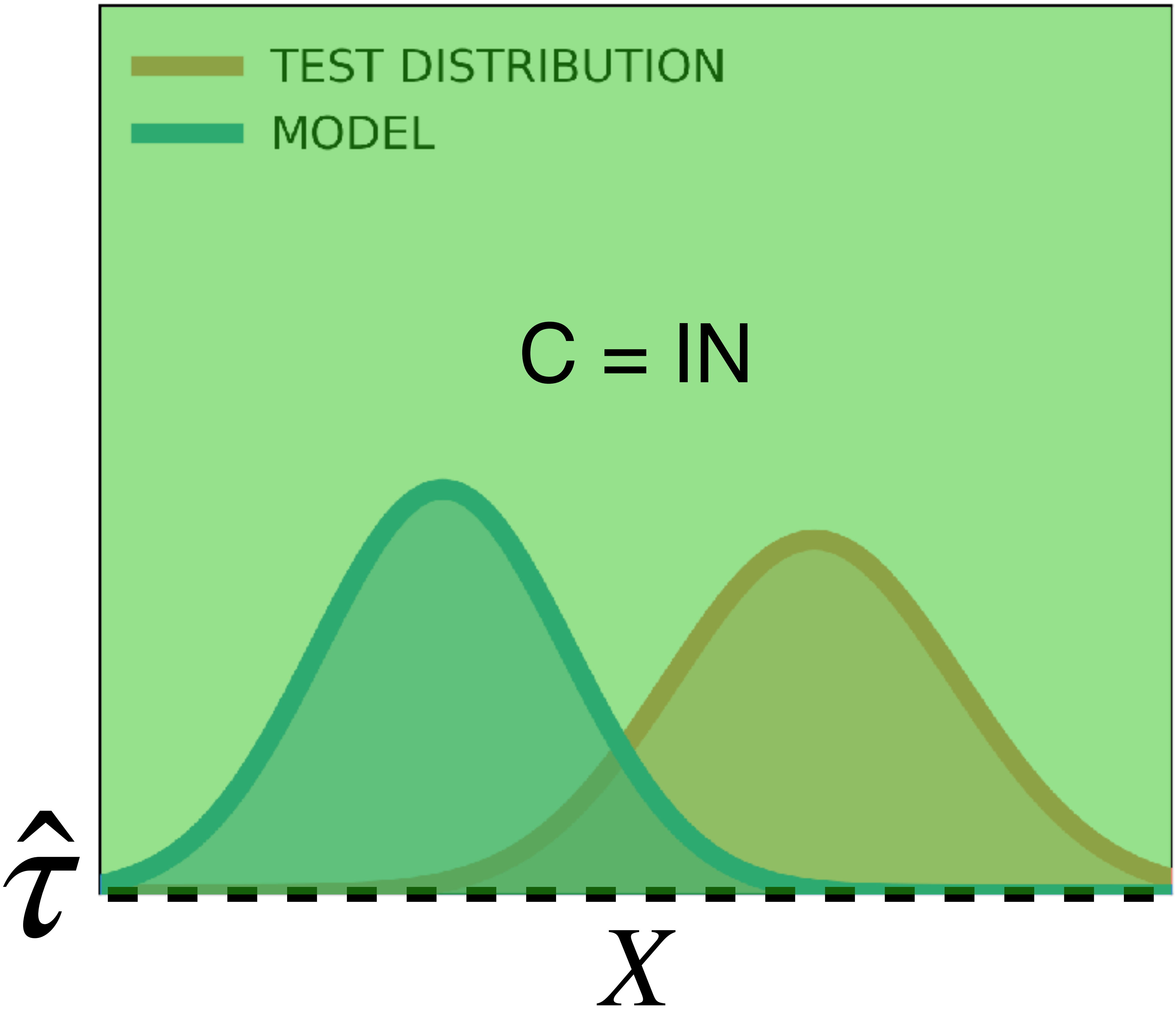
Which leads to the degenerate threshold:

$$q(\mathbf{X}^*) > \text{UNIFORM}(\mathbf{X}^*)\,\frac{p(\text{OUT})}{p(\text{IN})} = 0$$


TEST DISTRIBUTION
MODEL
C = IN
C = OUT
$\hat{\tau}$
X


TEST DISTRIBUTION
MODEL
C = IN
C = OUT
$\hat{\tau}$
X


TEST DISTRIBUTION
MODEL
C = IN
C =
OUT
$\hat{\tau}$
X


TEST DISTRIBUTION
MODEL
C = IN
$\hat{\tau}$
X

2. Stronger OOD Models

Going back to the Bayes classifier:

$$\frac{p(\mathbf{X}^* \mid \text{OUT})\ p(\text{OUT})}{p(\mathbf{X}^* \mid \text{IN})\ p(\text{IN})} \overset{?}{>} 1$$

Going back to the Bayes classifier:

$$\frac{p(\mathbf{X}^* \mid \text{OUT})\; p(\text{OUT})}{\underbrace{q(\mathbf{X}^*)}_{\text{DGM}}\; p(\text{IN})} \overset{?}{>} 1$$

Going back to the Bayes classifier:

$$\frac{p(\mathbf{X}^* ?|\,\text{OUT})\; p(\text{OUT})}{\underbrace{q(\mathbf{X}^*)}_{\text{DGM}}\; p(\text{IN})} \overset{?}{>} 1$$

Going back to the Bayes classifier:

$$\frac{p(\mathbf{X}^{*}\,?\,|\,\text{OUT})\; p(\text{OUT})}{q(\mathbf{X}^{*})\; p(\text{IN})} \overset{?}{>} 1$$

DGM

Recent work proposed stronger OOD models:

- Ren et al. [NeurIPS 2019]: $p(\mathbf{X}^{*}\,|\,\text{OUT})$ is defined using noisy training data / background simulation.
- Serra et al. [ICLR 2020]: $p(\mathbf{X}^{*}\,|\,\text{OUT})$ is defined using a compression algorithm.

Going back to the Bayes classifier:

$$\frac{p(\mathbf{X}^* ?\,|\,\text{OUT})\; p(\text{OUT})}{q(\mathbf{X}^*)\; p(\text{IN})} \overset{?}{>} 1$$

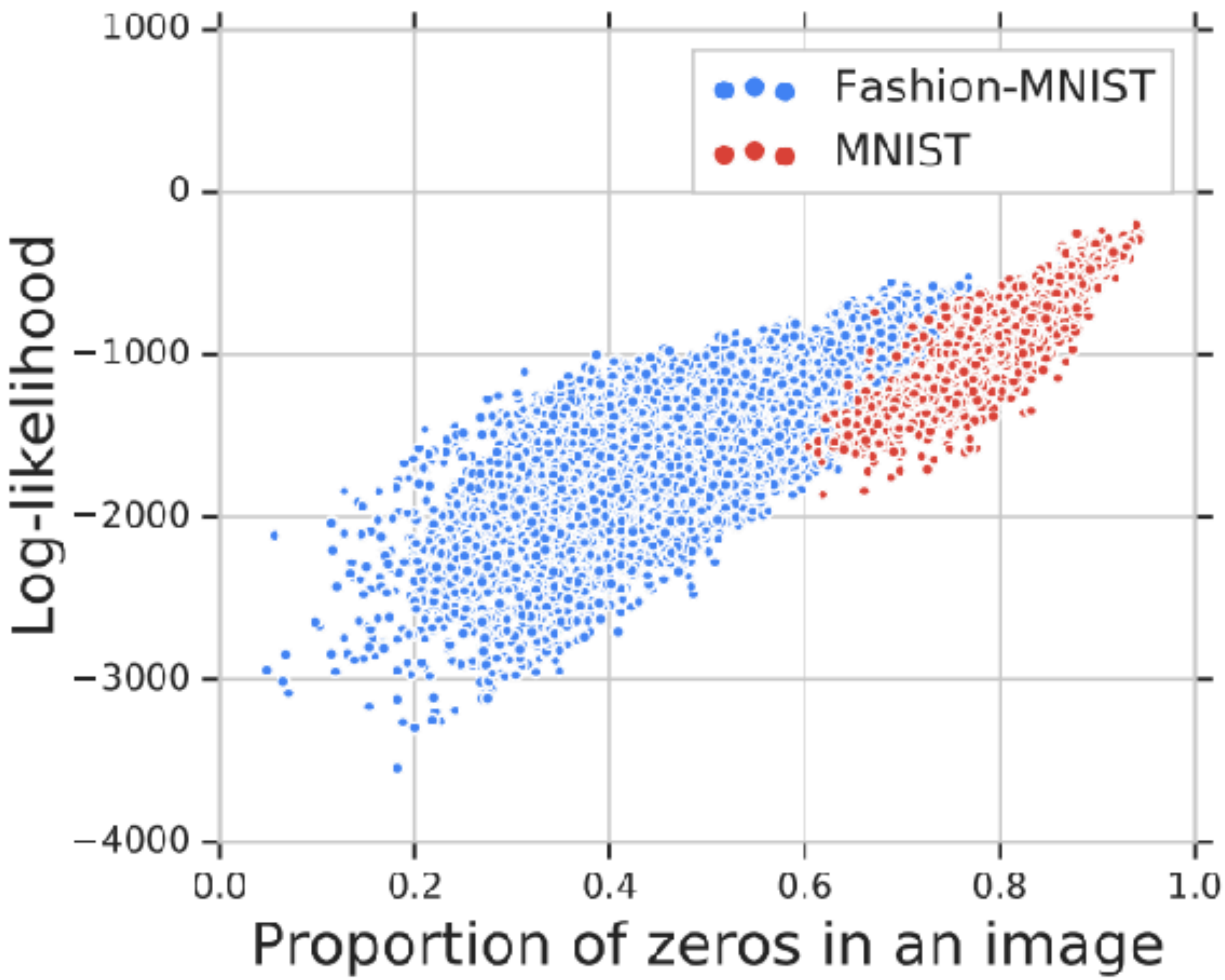
DGM

Recent work proposed stronger OOD models:

- Ren et al. [NeurIPS 2019]: $p(\mathbf{X}^*\,|\,\text{OUT})$ is defined using noisy training data / background simulation.
- Serra et al. [ICLR 2020]: $p(\mathbf{X}^*\,|\,\text{OUT})$ is defined using a compression algorithm.

Likelihood Ratios for OOD Detection

[Ren et al., NeurIPS 2019]



Likelihood Ratios for OOD Detection

[Ren et al., NeurIPS 2019]

Generate 'background' data and train additional DGM:

$$\tilde{q}(\mathbf{x})$$

Likelihood Ratios for OOD Detection

[Ren et al., NeurIPS 2019]

Generate 'background' data and train additional DGM:

$$\tilde{q}(\mathbf{x})$$

Compute the ratio with both models:

$$\frac{q(\mathbf{X}^{*})\ p(\text{IN})}{\tilde{q}(\mathbf{X}^{*})\ p(\text{OUT})} \overset{?}{>} 1$$

Likelihood Ratios for OOD Detection

[Ren et al., NeurIPS 2019]

Generate ‘background’ data and train additional DGM:

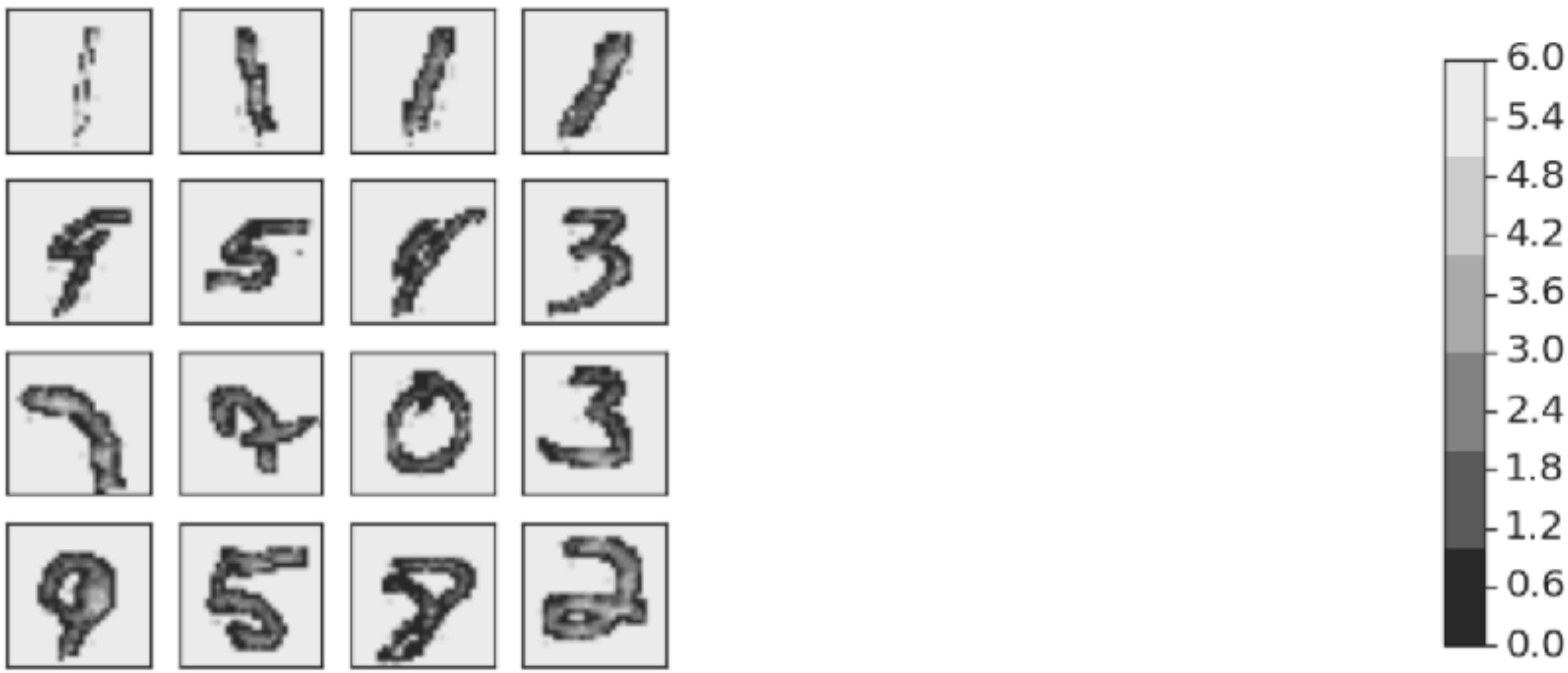
$$\tilde{q}(\mathbf{x})$$

Compute the ratio with both models:

$$\frac{q(\mathbf{X}^*)\ p(\text{IN})}{\tilde{q}(\mathbf{X}^*)\ p(\text{OUT})} > 1 \implies \text{C = IN}$$

Likelihood Ratios for OOD Detection

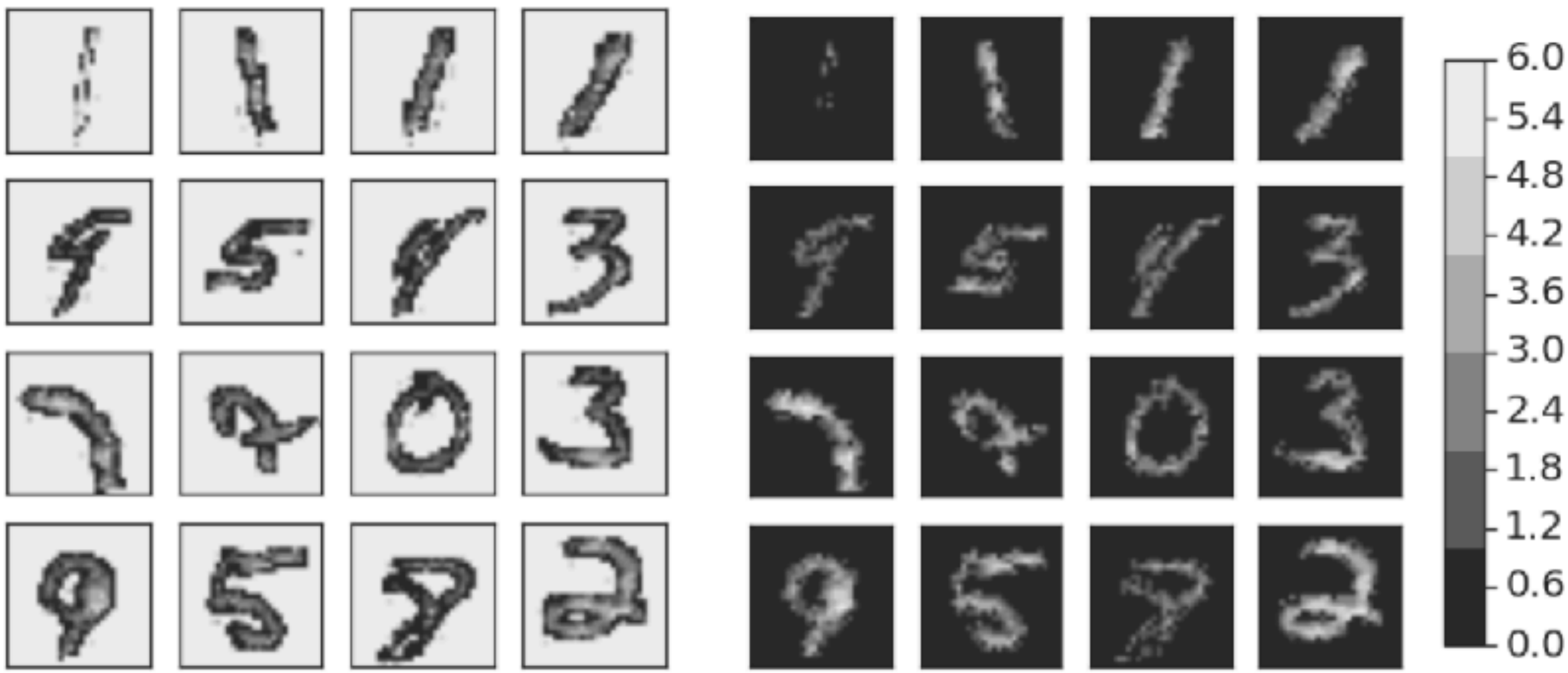
[Ren et al., NeurIPS 2019]



(c) Likelihood

Likelihood Ratios for OOD Detection

[Ren et al., NeurIPS 2019]



(c) Likelihood (d) Likelihood-Ratio

Overview of Ratio Methods

PAPER	$p(\mathbf{X}^* \mid \text{OUT})$ MODEL
C. Bishop [1994] (thresholding)	UNIFORM $(\mathbf{X}^*)$
Ren et al. [2019]	$\tilde{q}(\mathbf{X}^*)$ BACKGROUND / NOISE MODEL
Serra et al. [2020]	$2^{-\lvert\mathscr{C}(\mathbf{X}^*)\rvert/D}$ COMPRESSION ALGORITHM

Overview of Ratio Methods

PAPER	$p(\mathbf{X}^* \mid \text{OUT})$ MODEL
C. Bishop [1994] (thresholding)	UNIFORM $(\mathbf{X}^*)$
Ren et al. [2019]	$\tilde{q}(\mathbf{X}^*)$ BACKGROUND / NOISE MODEL
Serra et al. [2020]	$2^{-\lvert\mathscr{C}(\mathbf{X}^*)\rvert/D}$ COMPRESSION ALGORITHM
??? [2020+]	$\vdots$

3. Moving Towards Omnibus Methods

With a strong OOD model, ratios / the Bayes classifier can work well in practice (as Ren et al. and Serra et al. demonstrate).

With a strong OOD model, ratios / the Bayes classifier can work well in practice (as Ren et al. and Serra et al. demonstrate).

However, ratio methods are really performing ***model selection***.

WHICH MODEL BETTER REPRESENTS **X***?

$$\frac{p(\mathbf{X}^{*} \mid \text{OUT})\; p(\text{OUT})}{p(\mathbf{X}^{*} \mid \text{IN})\; p(\text{IN})} \overset{?}{>} 1$$

Ratios operate under the **M-closed** assumption [Bernardo & Smith, 1994]: we know *all* models that could possibly generate the OOD data.

Ratios operate under the **M-closed** assumption [Bernardo & Smith, 1994]: we know *all* models that could possibly generate the OOD data.

But in the real world, the **M-open** assumption is more appropriate: we *don't know* all of the possible OOD models.

Ratios operate under the **M-closed** assumption [Bernardo & Smith, 1994]: we know *all* models that could possibly generate the OOD data.

But in the real world, the **M-open** assumption is more appropriate: we *don't know* all of the possible OOD models.

Hence, we need *omnibus* methods that check for *all* departures from the DGM:

$$q(\mathbf{x}) \neq p(\mathbf{x}) \qquad \forall p \in \mathscr{P}$$

$$q(\mathbf{x}) \neq p(\mathbf{x}) \qquad \forall p \in \mathscr{P}$$

The classic goodness-of-fit tests (e.g. KS-test) do check for all departures.

$$q(\mathbf{x}) \neq p(\mathbf{x}) \qquad \forall p \in \mathscr{P}$$

The classic goodness-of-fit tests (e.g. KS-test) do check for all departures.

But these tests require access to the model's CDF—which is intractable to compute for DGMs.

🙁

Other alternatives?

Other alternatives?

1. Kernelized Stein Discrepancy [Gorham & Mackey, 2015; Liu et al., 2016]

Other alternatives?

1. Kernelized Stein Discrepancy [Gorham & Mackey, 2015; Liu et al., 2016]

2. A Test for Typicality [Nalisnick et al., 2019]

Detecting Out-of-Distribution Inputs to Deep Generative Models Using Typicality

Eric Nalisnick* DeepMind   **Akihiro Matsukawa†** D. E. Shaw   **Yee Whye Teh** DeepMind   **Balaji Lakshminarayanan*** DeepMind

Typical Set

Definition: For a distribution $q(\mathbf{x})$, the *ε-typical set* is comprised of all M-length sequences that satisfy:

$$\left| \mathbb{H}[q(\mathbf{x})] - \frac{1}{M} \sum_{m=1}^{M} -\log q(\mathbf{x}_m^*) \right| \leq \epsilon$$

Typical Set

Definition: For a distribution $q(\mathbf{x})$, the *ε-typical set* is comprised of all M-length sequences that satisfy:

$$\left| \mathbb{H}[q(\mathbf{x})] - \frac{1}{M} \sum_{m=1}^{M} - \log q(\mathbf{x}_m^*) \right| \leq \epsilon$$

ENTROPY OF DGM

Typical Set

Definition: For a distribution $q(\mathbf{x})$, the *ε-typical set* is comprised of all M-length sequences that satisfy:

$$\left| \mathbb{H}[q(\mathbf{x})] - \frac{1}{M}\sum_{m=1}^{M} -\log q(\mathbf{x}_m^*) \right| \leq \epsilon$$

ENTROPY OF DGM

TEST OBSERVATION

Typical Set

Definition: For a distribution $q(\mathbf{x})$, the *ε-typical set* is comprised of all M-length sequences that satisfy:

$$\left| \mathbb{H}[q(\mathbf{x})] - \frac{1}{M}\sum_{m=1}^{M} -\log q(\mathbf{x}_m^*) \right| \leq \epsilon$$

ENTROPY OF DGM

TEST OBSERVATION

SMALL POS. CONSTANT

(SET THROUGH BOOTSTRAP SIMULATION)

Typical Set

Definition: For a distribution $q(\mathbf{x})$, the *ε-typical set* is comprised of all M-length sequences that satisfy:

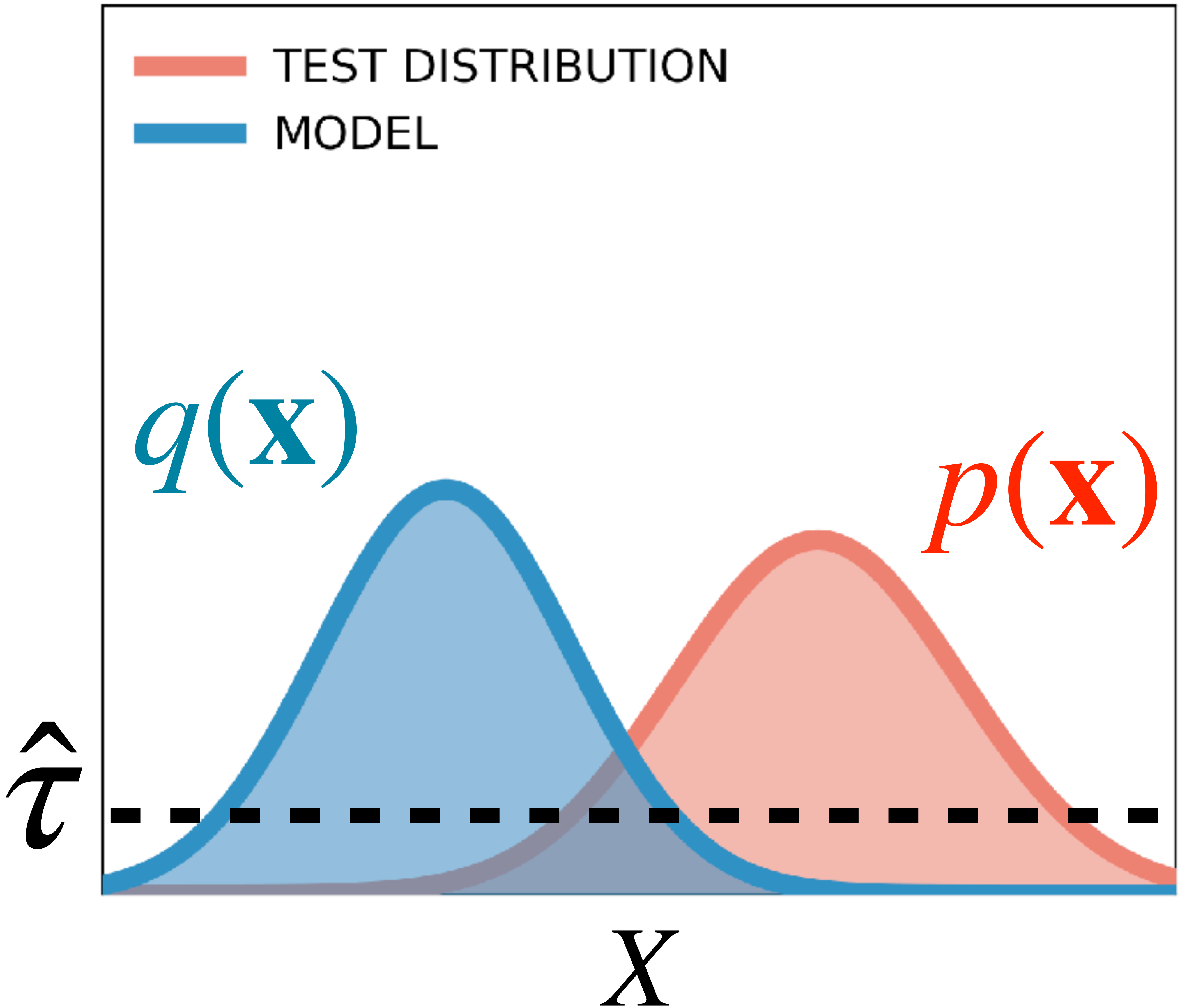
$$\left| \mathbb{H}[q(\mathbf{x})] - \frac{1}{M} \sum_{m=1}^{M} -\log q(\mathbf{x}_m^*) \right| \leq \epsilon$$

ENTROPY OF DGM

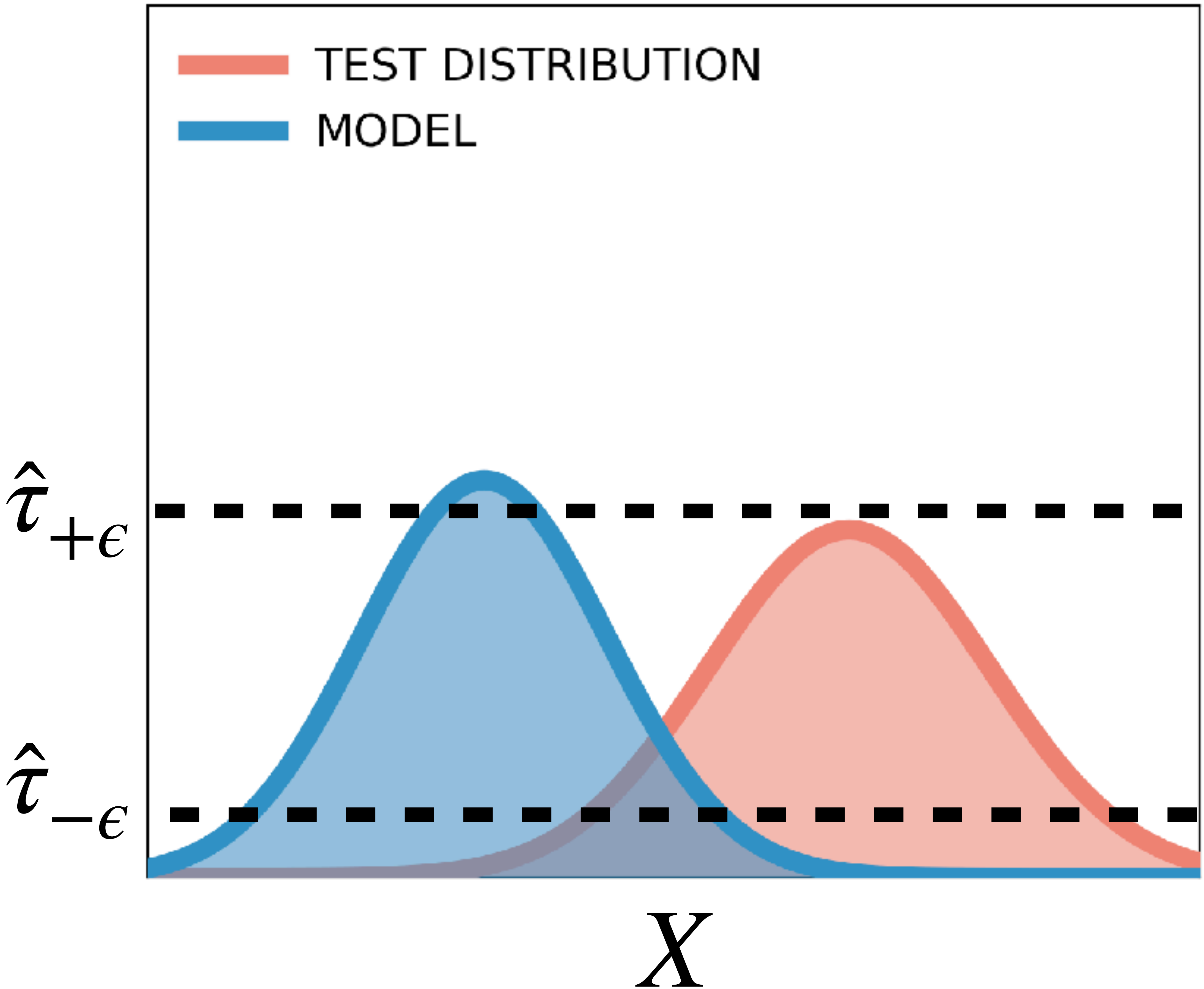
TEST OBSERVATION

SMALL POS. CONSTANT (SET THROUGH BOOTSTRAP SIMULATION)

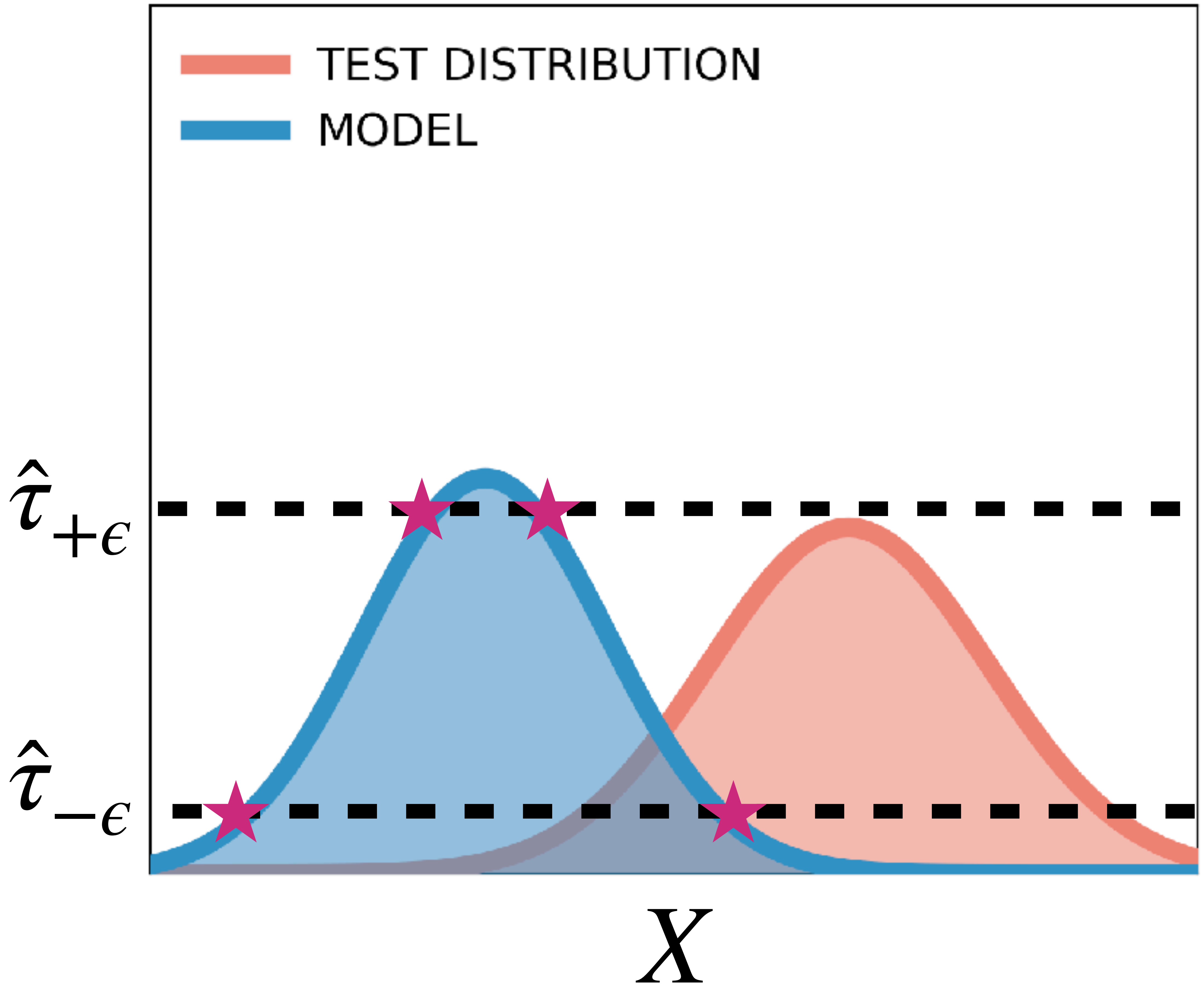
Intuition: We truly care about the high probability (a.k.a. minimum volume) set. The typical set is an approximation to that set, defined in terms of entropy, which we can more easily compute.

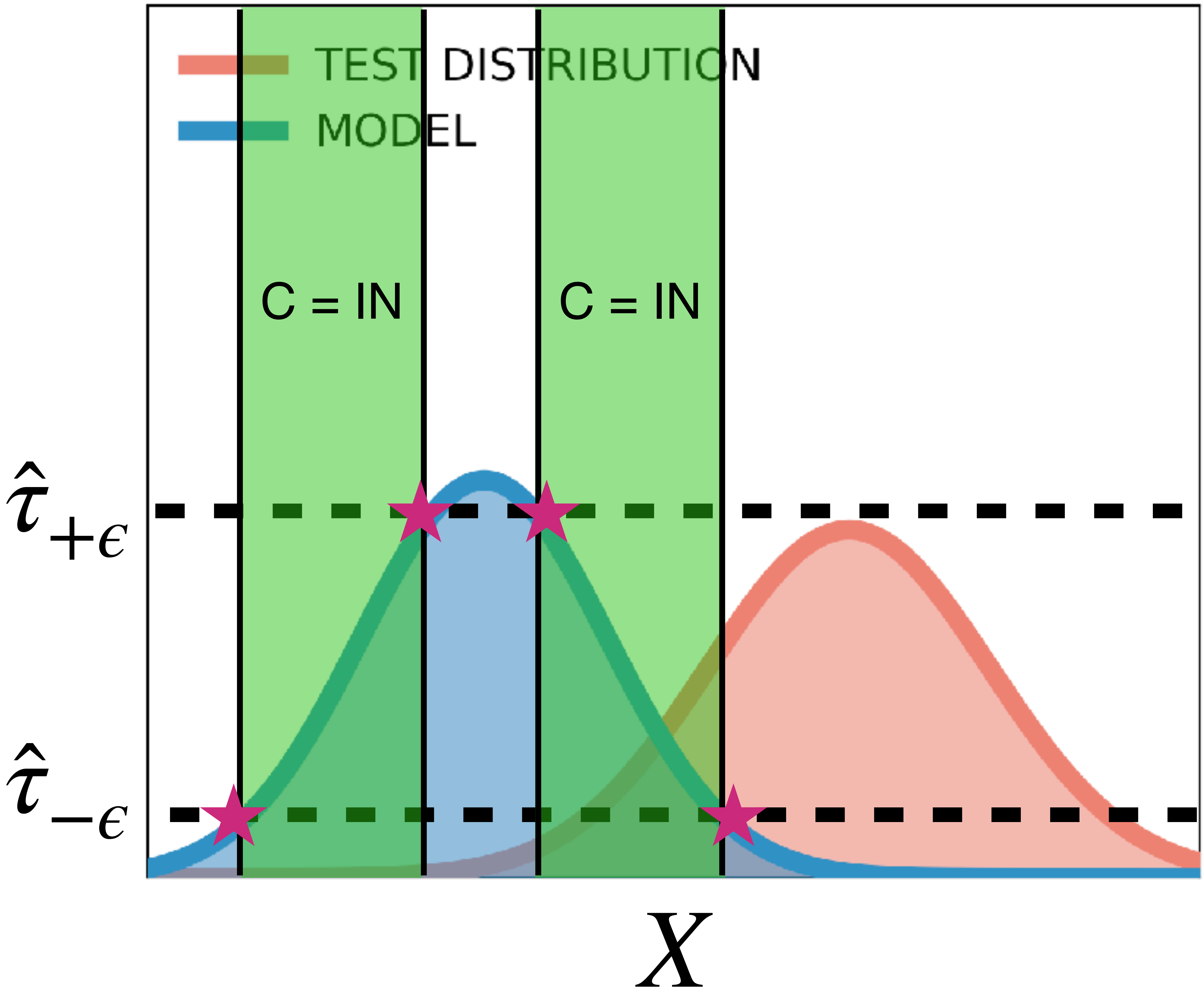


TEST DISTRIBUTION
MODEL
$q(\mathbf{x})$
$p(\mathbf{x})$
$\hat{\tau}$
X



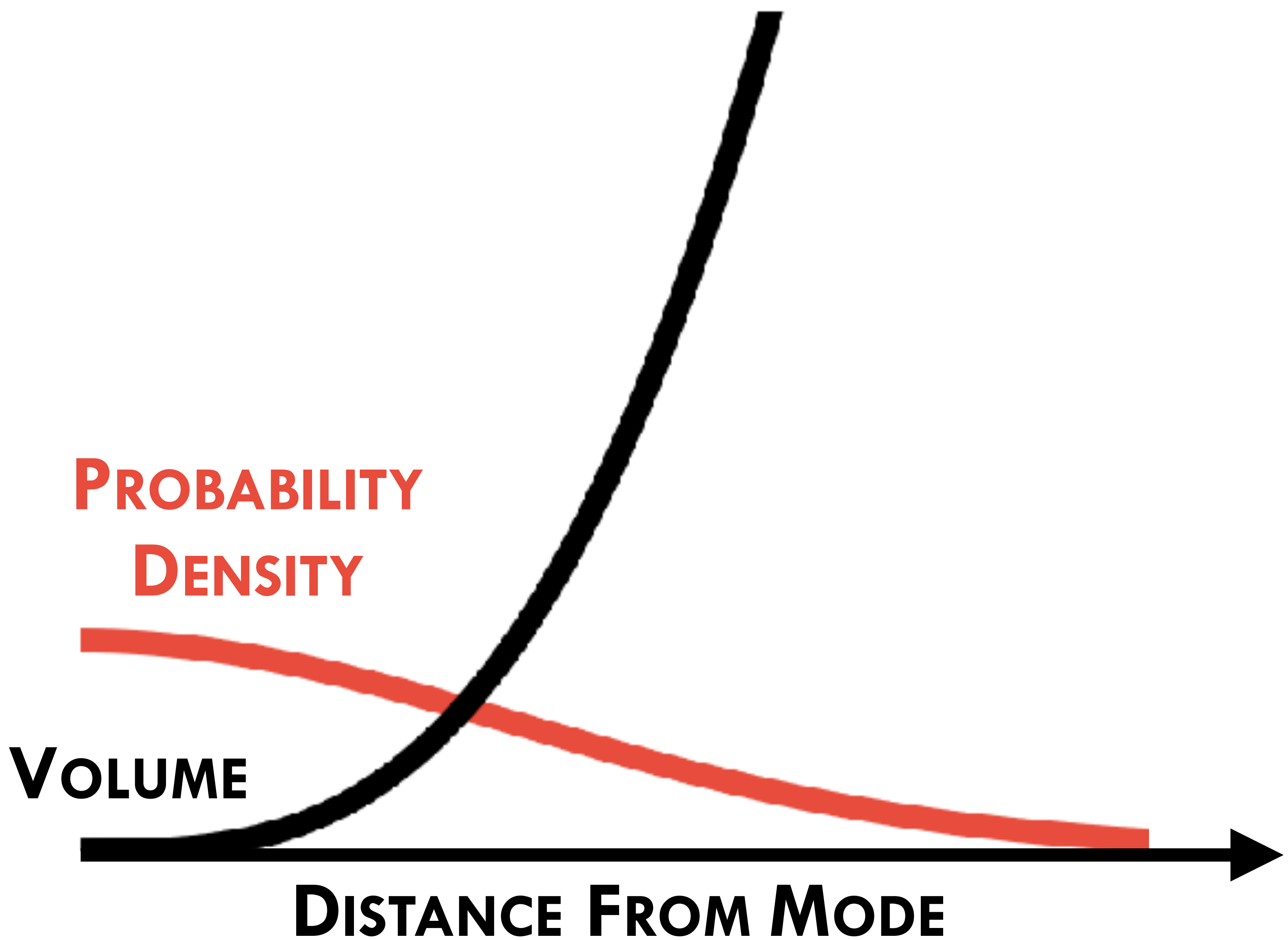
TEST DISTRIBUTION
MODEL
$\hat{\tau}_{+\epsilon}$
$\hat{\tau}_{-\epsilon}$
X


TEST DISTRIBUTION
MODEL
$\hat{\tau}_{+\epsilon}$
$\hat{\tau}_{-\epsilon}$
X

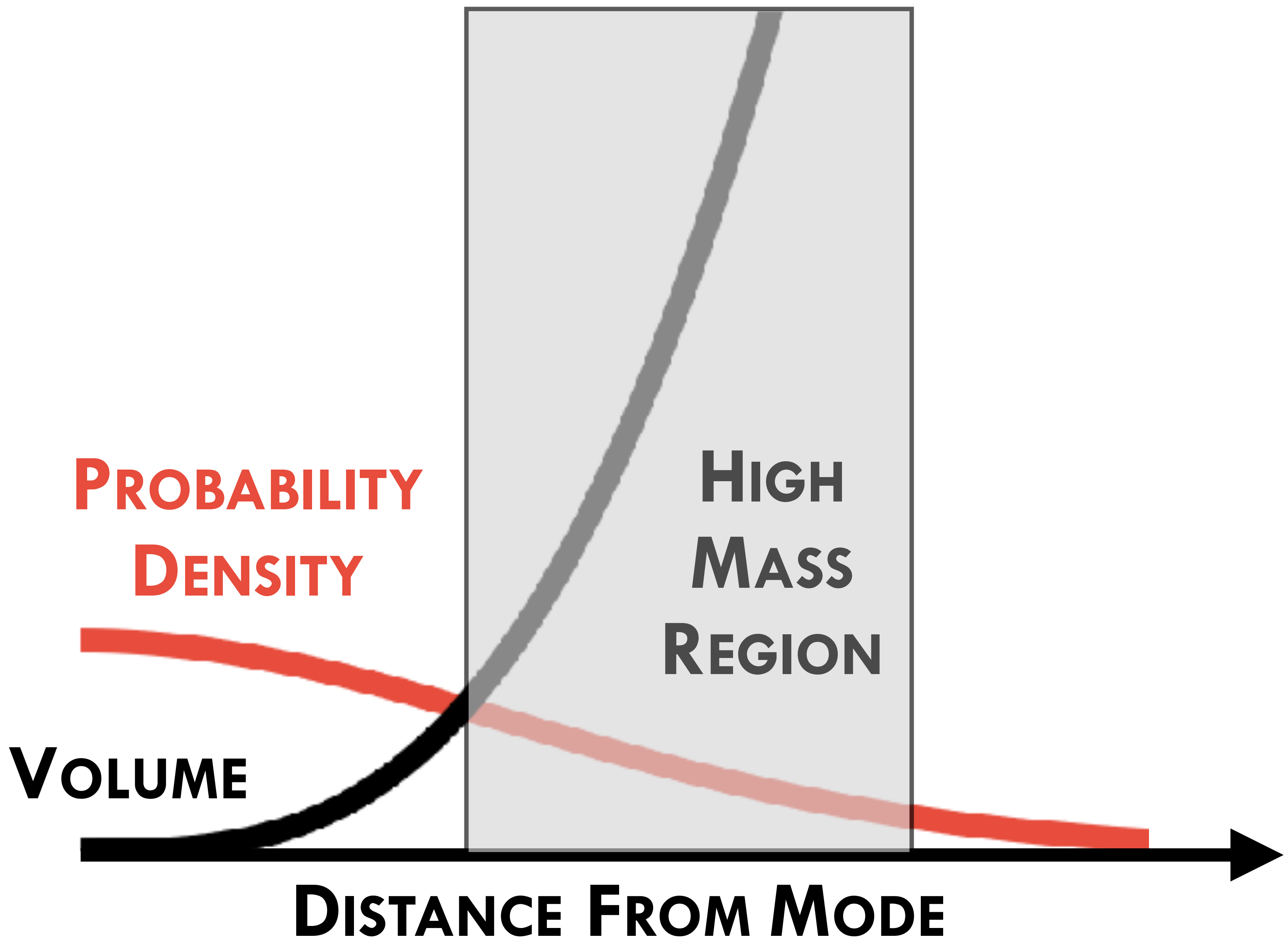


TEST DISTRIBUTION
MODEL
C = IN
C = IN
$\hat{\tau}_{+\epsilon}$
$\hat{\tau}_{-\epsilon}$
X

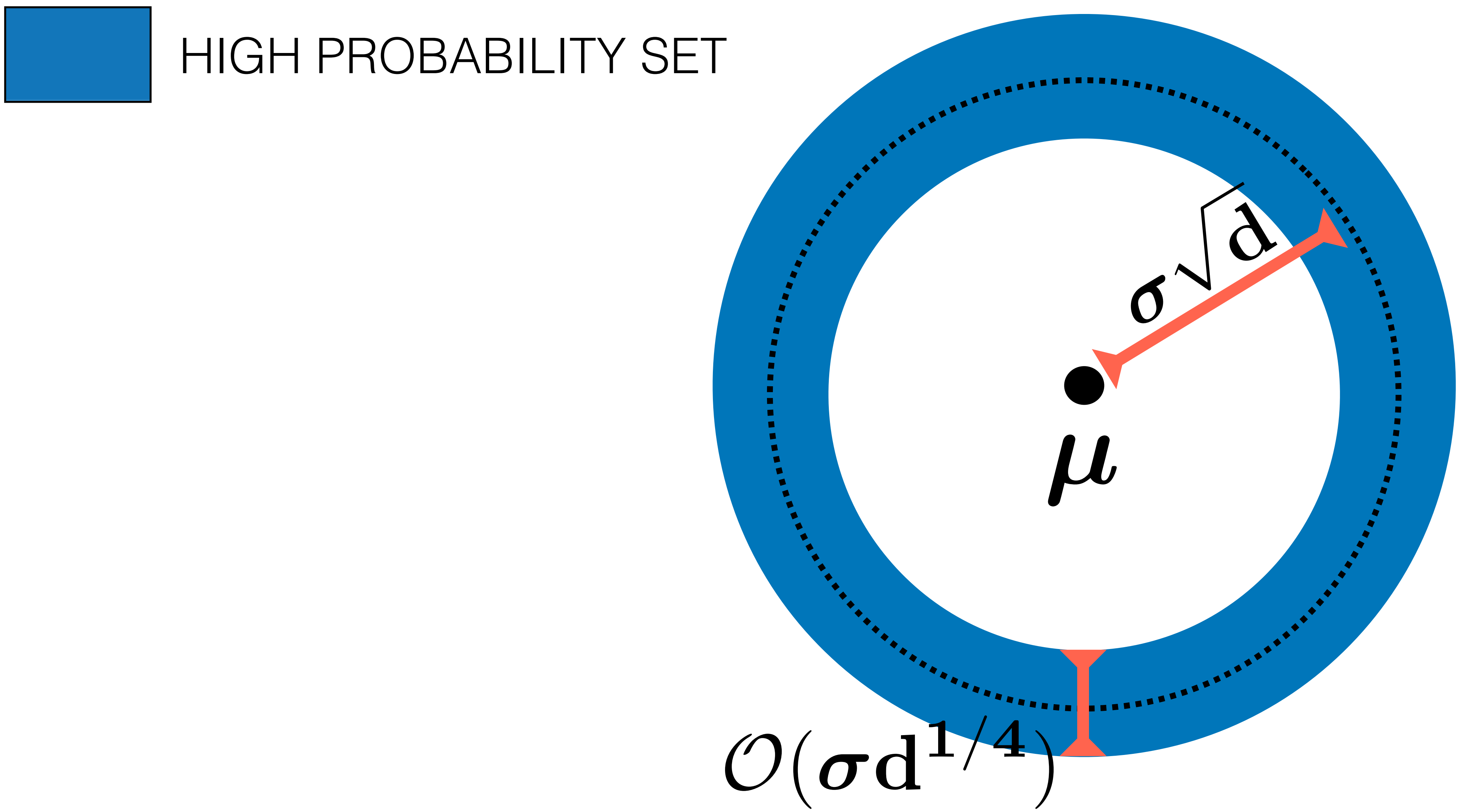
In high dimensions, probability mass concentrates *away* from the mode.



In high dimensions, probability mass concentrates *away* from the mode.

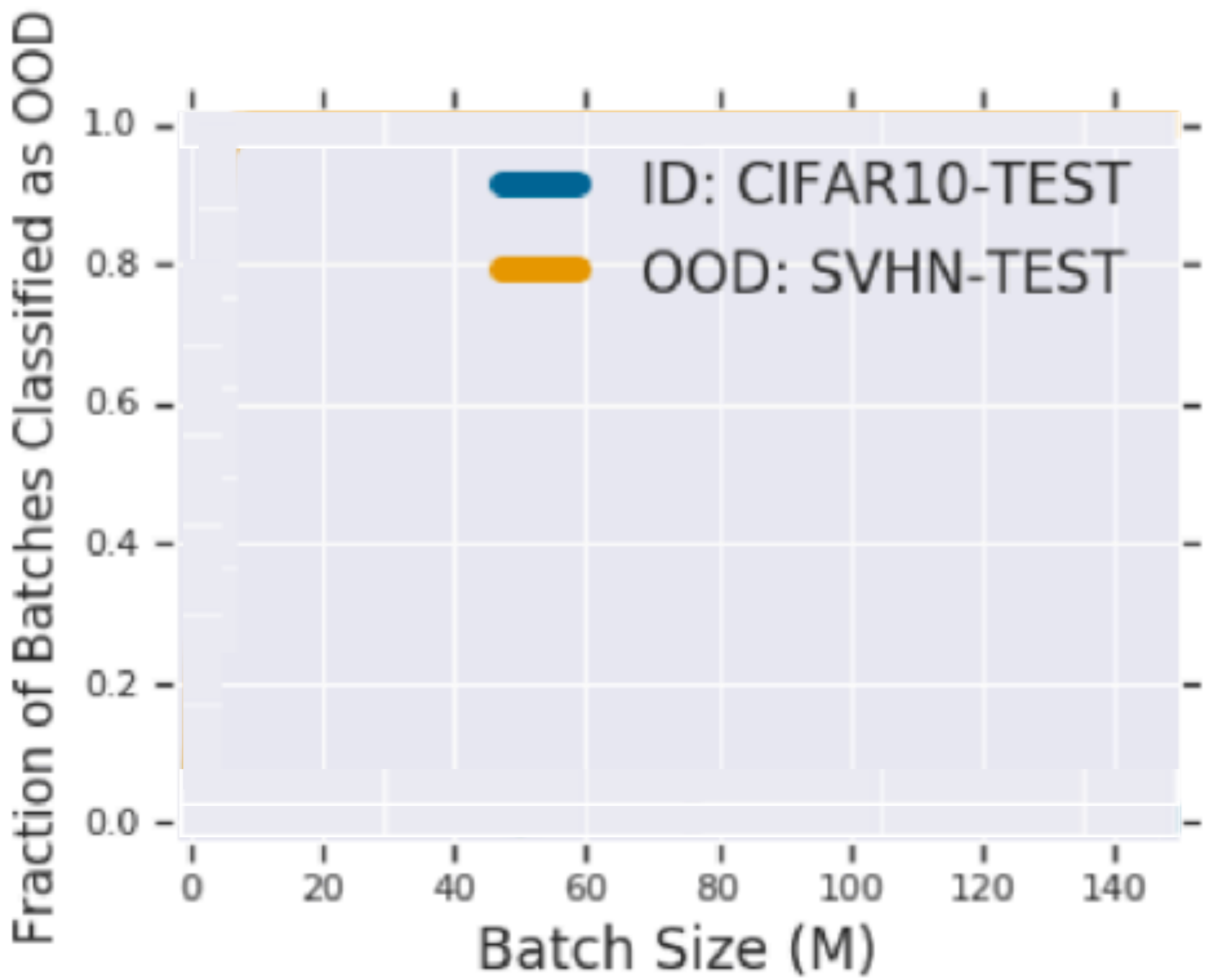


In high dimensions, probability mass concentrates *away* from the mode.



HIGH DIMENSIONAL GAUSSIAN

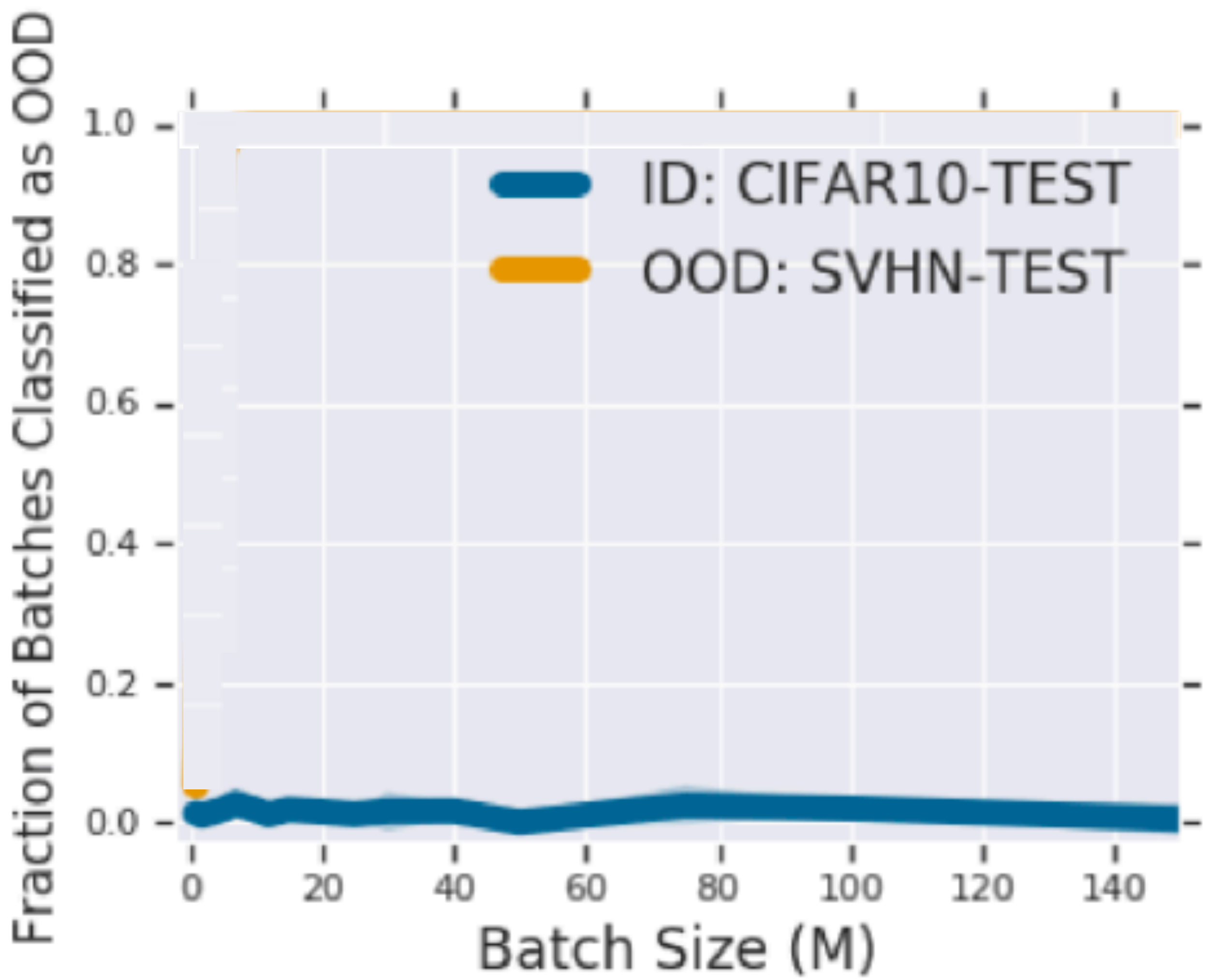
GLOW: CIFAR-10 VS SVHN



(d) CIFAR10 Train, SVHN Test

[Nalisnick et al., BDL 2019]

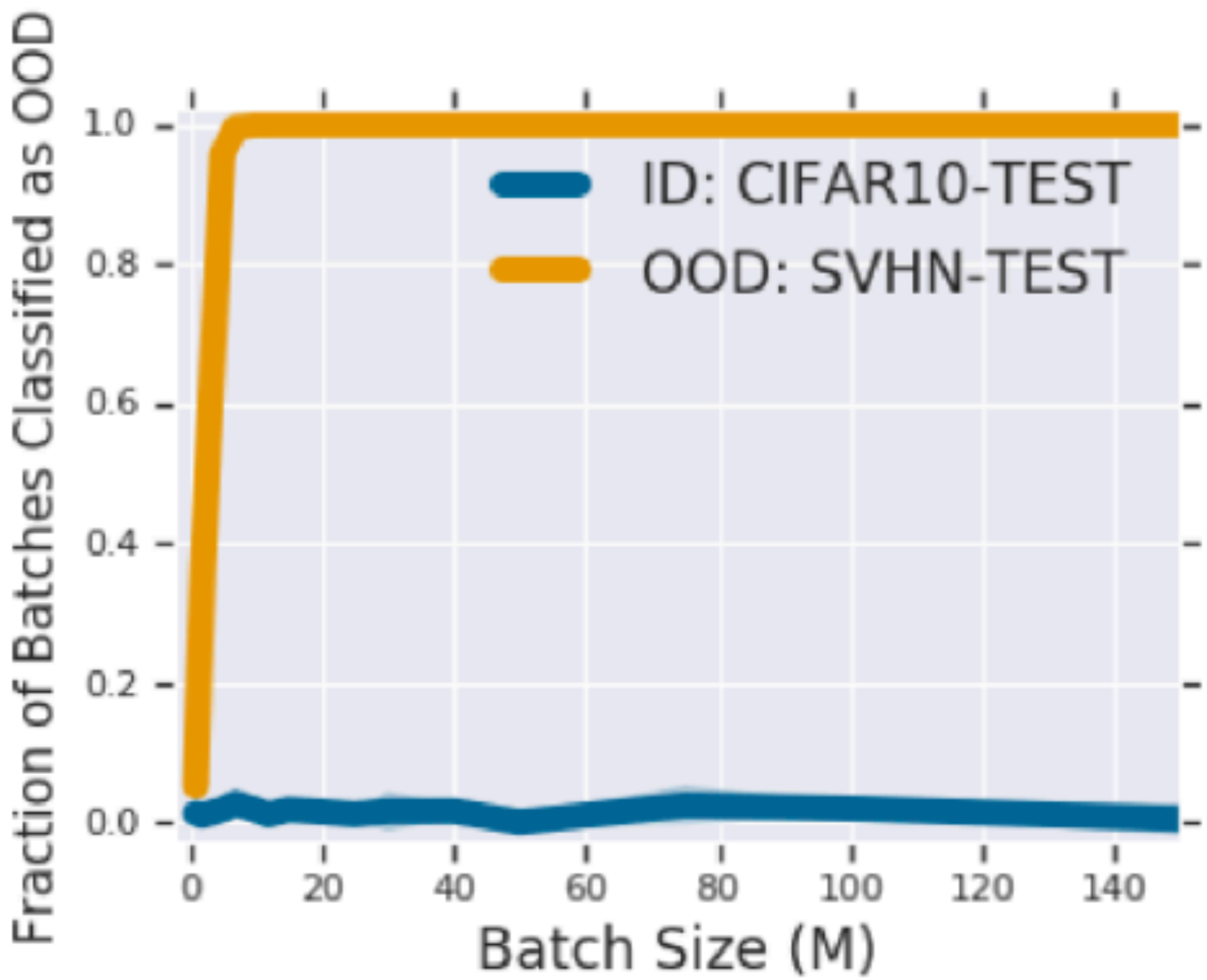
GLOW: CIFAR-10 VS SVHN



(d) CIFAR10 Train, SVHN Test

[Nalisnick et al., BDL 2019]

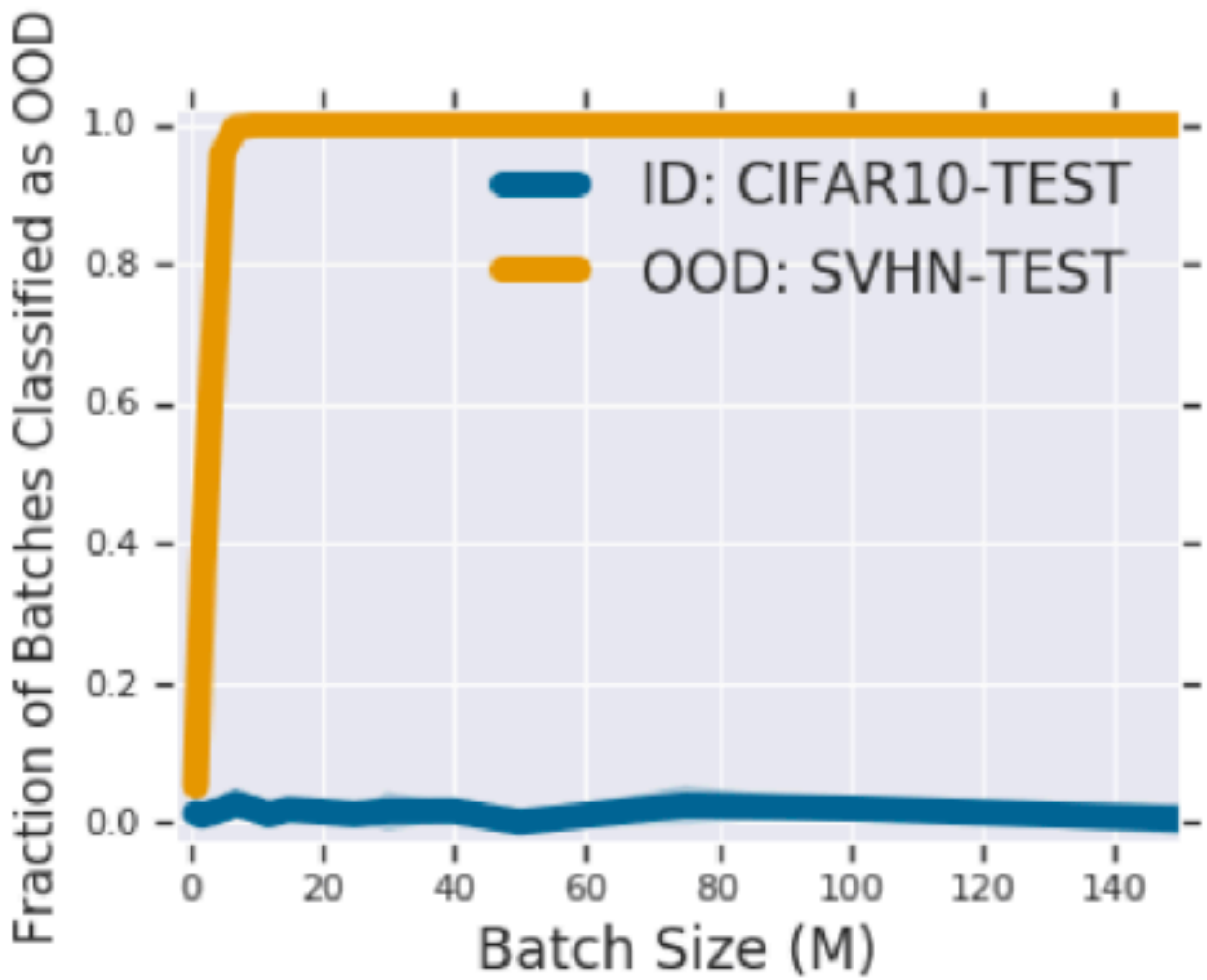
GLOW: CIFAR-10 VS SVHN



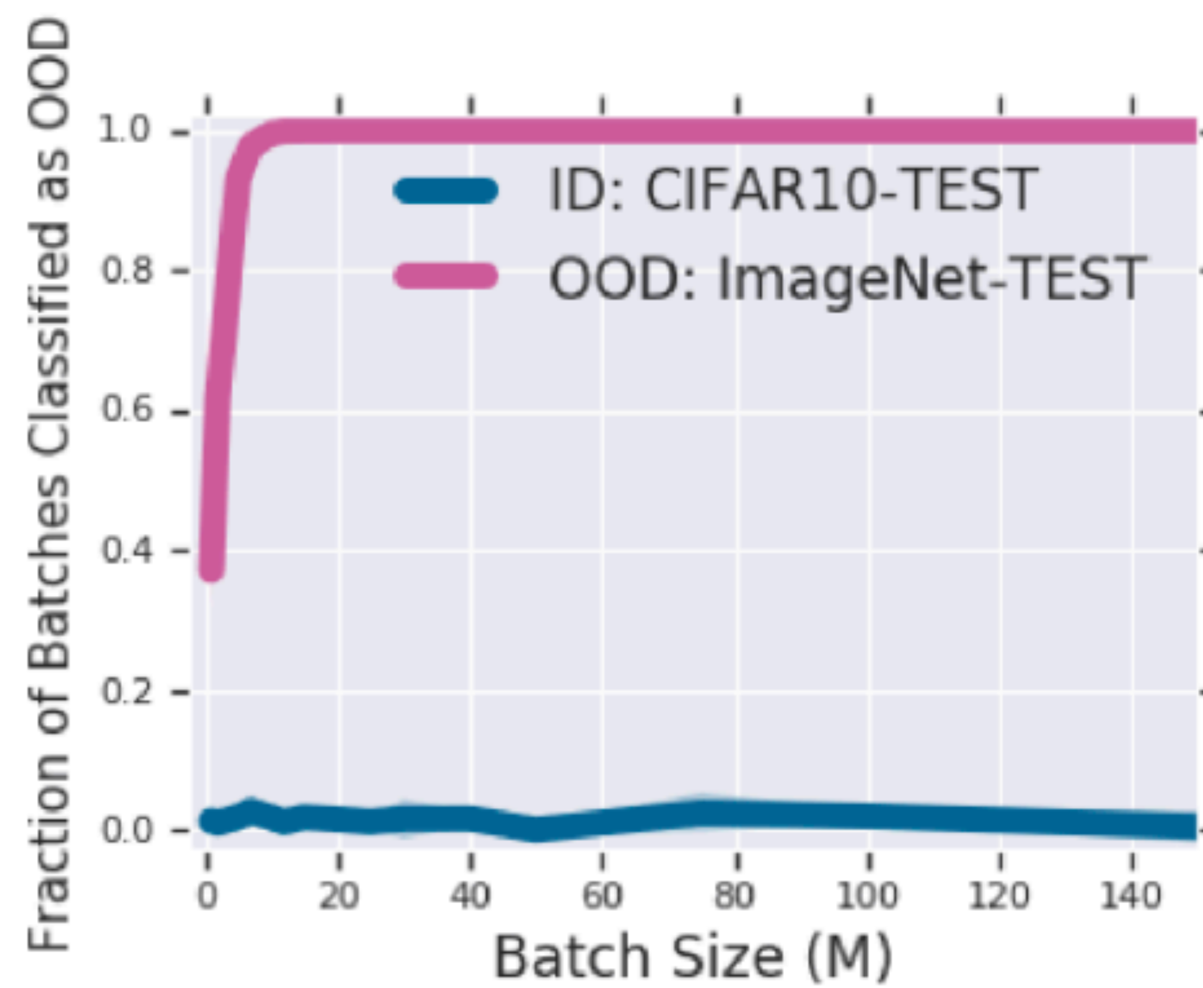
(d) CIFAR10 Train, SVHN Test

[Nalisnick et al., BDL 2019]

GLOW: CIFAR-10 VS SVHN



(d) CIFAR10 Train, SVHN Test



(f) CIFAR10 Train, ImageNet Test

[Nalisnick et al., BDL 2019]

COMPARISON WITH STEIN DISCREPANCY

OOD Accuracy (M=10) for
FashionMNIST vs NotMNIST

[Nalisnick et al., BDL 2019]

COMPARISON WITH STEIN DISCREPANCY

OOD Accuracy (M=10) for
FashionMNIST vs NotMNIST

GLOW		
	Typicality Test	**69%**
	Kernel Stein Discrep.	1%

[Nalisnick et al., BDL 2019]

COMPARISON WITH STEIN DISCREPANCY

OOD Accuracy (M=10) for
FashionMNIST vs NotMNIST

GLOW	Typicality Test	**69%**
	Kernel Stein Discrep.	1%
PIXEL CNN	Typicality Test	1%
	Kernel Stein Discrep.	**61%**

[Nalisnick et al., BDL 2019]

COMPARISON WITH STEIN DISCREPANCY

OOD Accuracy (M=10) for
FashionMNIST vs NotMNIST

GLOW	Typicality Test	**69%**
	Kernel Stein Discrep.	1%
PIXEL CNN	Typicality Test	1%
	Kernel Stein Discrep.	**61%**
VAE	Typicality Test	100%
	Kernel Stein Discrep.	100%

[Nalisnick et al., BDL 2019]

FOLLOW-UP WORK

Density of States Estimation for Out-of-Distribution Detection

Warren R. Morningstar *
Google Research
wmorning@google.com

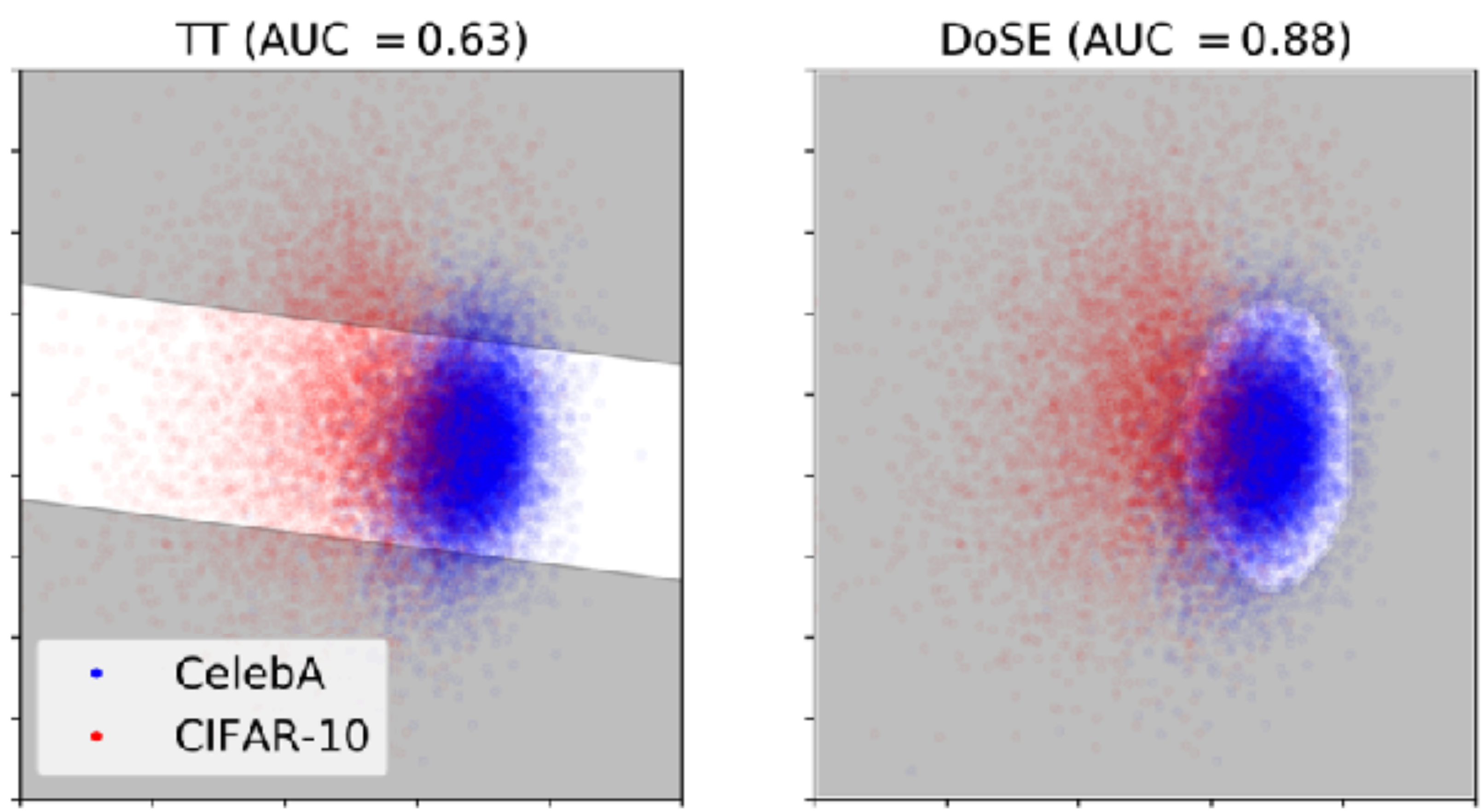
Cusuh Ham †
Georgia Institute of Technology
cusuh@gatech.edu

Andrew G. Gallagher
Google Research
agallagher@google.com

Balaji Lakshminarayanan
Google Research
balajiln@google.com

Alexander A. Alemi
Google Research
alemi@google.com

Joshua V. Dillon *
Google Research
jvdillon@google.com



CONCLUSIONS

- ⊗ **OOD detection is a useful application for DGMs.** Same methods can also assess the DGM's fit to the in-distribution set.

- ⊗ **Likelihood ratio methods assume *M-closed* worlds. In practice we usually need *M-open* assumptions.**

- ⊗ **Can we design DGMs with tractable CDFs?** Computing probabilities would expand the applications of DGMs for statistical inference.

- ⊗ **Time for more theory in OOD detection.** Safety-critical applications require guarantees.

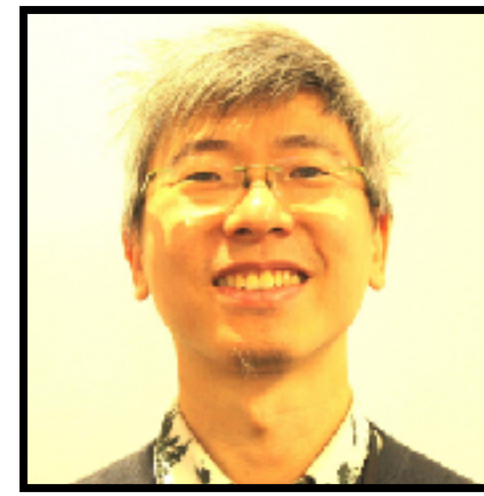
Thank you. Questions?

In collaboration with…

Aki Matsukawa

Dilan Gorur

Balaji Lakshminarayanan



Yee Whye Teh

enalisnick.github.io/

eric_nalisnick