Score-based Hypothesis Testing and Chang-point Detection for Unnormalized Models

Suya Wu

Vahid Tarokh’s team

Duke University

Outline

01

Score Matching

02

Score-based Hypothesis Testing

03

Score-based Change-point Detection

04

Results and Applications

Score

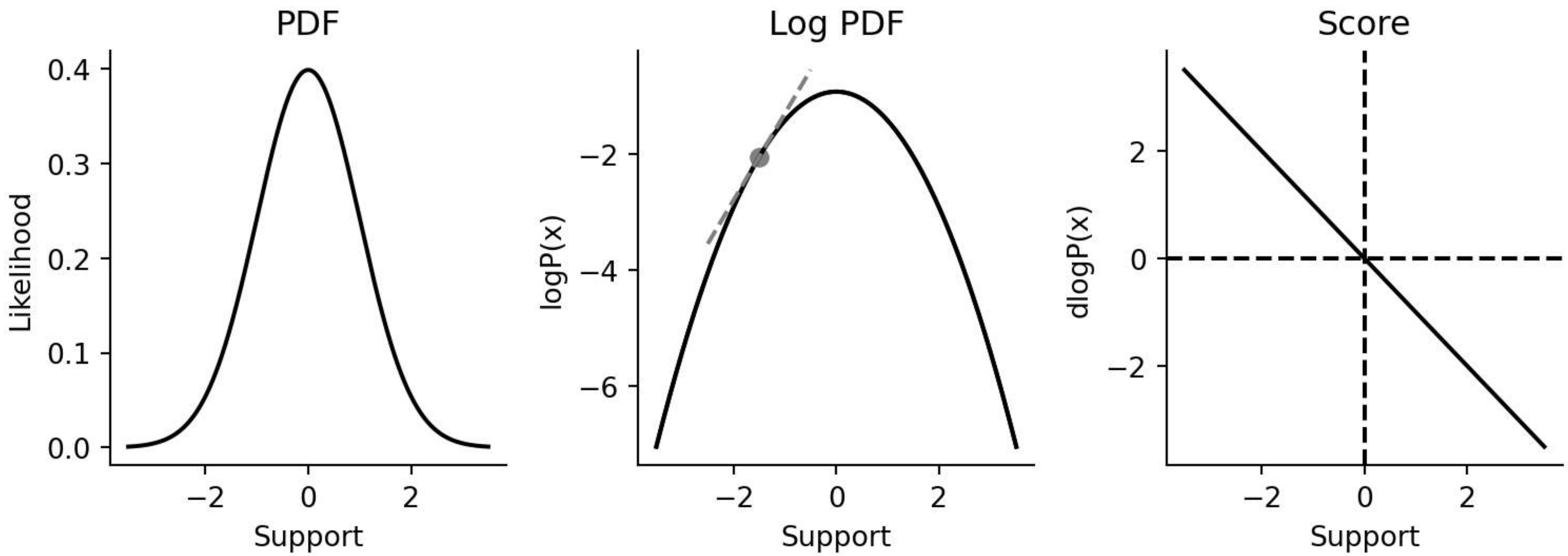


Figure: P(x) (likelihood, PDF), logP(x) (log likelihood, logp), and dlogP(x) (score) of a Gaussian.
Souce: https://ericmjl.github.io

Fisher Divergence

- For a random variable $\boldsymbol{x} \in \boldsymbol{X} \subseteq R^d$, and the probability density functions (PDFs) $\boldsymbol{x}: \longmapsto p(\boldsymbol{x})$ and $\boldsymbol{x}: \longmapsto q(\boldsymbol{x})$, which represent two probability distributions P and Q on $\boldsymbol{X}$.

Kullback–Leibler (KL) Divergence

$$D_{KL}[p||q] = E_{\boldsymbol{x}\sim p}\left[\log p(\boldsymbol{x}) - \log q(\boldsymbol{x})\right] = E_{\boldsymbol{x}\sim p}[-\log q(x)] + c$$

- The minimization of KL-divergence over Q is equivalent to minimize the negative log likelihood (also called log-score or logarithmic scoring rule).

Fisher Divergence

$$D_F[p||q] = E_{\boldsymbol{x}\sim p}[||\nabla_{\boldsymbol{x}} \log p(\boldsymbol{x}) - \nabla_x \log q(\boldsymbol{x})||^2]$$

Duke

Fisher Divergence

- Suppose that our knowledge is up to an unnormalized term, say $q(\boldsymbol{x}) \propto \tilde{q}(\boldsymbol{x})$, and $q(\boldsymbol{x}) = \frac{\tilde{q}(\boldsymbol{x})}{\int_x \tilde{q}(\boldsymbol{x})\mathrm{d}x}$.
- Minimizing the KL divergence can be computationally challenging
  - The partition function $\int \tilde{q}(\boldsymbol{x})\mathrm{d}x$ is not easy to compute
  - But the unnormalized form (numerator) is simple
- Alternative solution ?
- Consider minimizing the Fisher divergence from p to q instead of the KL divergence
  - Invariant to any positive scale
  - Avoid computing cumbersome normalizing constants

Duke

Hyvarinen Score

Fisher Divergence

$$D_F[p||q] = E_{\boldsymbol{x}\sim p}||\nabla_{\boldsymbol{x}} \log p(\boldsymbol{x}) - \nabla_x \log q(\boldsymbol{x})||^2 = E_{\boldsymbol{x}\sim p}\{s_{\mathrm{H}}(\boldsymbol{x}, q)\} + c_*$$

- The term c_* only depends on p, the true data-generating distribution
- The minimum, zero, is achieved if and only if $q(x) = p(x)$
- $s_{\mathrm{H}}(\boldsymbol{x}, q) \triangleq \frac{1}{2} ||\nabla_{\boldsymbol{x}} \log q(\boldsymbol{x})||_2^2 + \Delta_{\boldsymbol{x}} \log q(\boldsymbol{x})$ is known as the Hyvarinen score. Here, the Laplacian operator $\Delta_{\boldsymbol{x}} \log q(\boldsymbol{x}) = \sum_{i=1}^{d} \frac{\partial^2}{\partial x_i^2}$
- The minimization over Fisher divergence is then reduced to the minimization of the expected Hyvarinen score $E_{\boldsymbol{x}\sim p}\, s_{\mathrm{H}}(\boldsymbol{x}, q)$.

Duke

Score Matching

- Consider the parametric family represented by $\{q_\theta : \theta \in \Theta\}$
- Suppose that a finite sample of points $\boldsymbol{X}_n \triangleq \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ are independent and identically distributed (IID) observations according to $p = q_{\theta^\star}$

Score Matching

$$\hat{\theta}_{\mathrm{sm}} \triangleq \mathrm{argmin}_{\theta \in \Theta} \frac{1}{2} \sum_{i=1}^{n} s_{\mathrm{H}}(\boldsymbol{x}_i, q_\theta)$$

- It is approximately minimizing $E_{\boldsymbol{x} \sim p}\, s_{\mathrm{H}}(\boldsymbol{x}, q_\theta)$
- It leads to $\hat{\theta}_{\mathrm{sm}} \to \theta_*$ in probability

Duke

Hypothesis Testing

- Suppose $p = q_{\theta^\star}$ for some $\theta^\star \in \Theta$
- To test the hypothesis if $\theta^\star = \theta_0$ for a given $\theta_0 \in \Theta$

Problem Definition

$$H_0\colon\ \theta^\star = \theta_0 \qquad \text{versus} \qquad H_1\colon\ \theta^\star \in \Theta \backslash \{\theta_0\}$$

- Likelihood Ratio Test (LRT):
  - Take the ratio of likelihoods as the test statistic
  - Widely accepted: The uniformly most powerful test for simple hypothesis testing

Duke

Hypothesis Testing

- Suppose that $q\left(\boldsymbol{x}\right) \propto \tilde{q}\left(\boldsymbol{x}\right)$, then $q\left(\boldsymbol{x}\right) = \frac{\tilde{q}\left(\boldsymbol{x}\right)}{\int_x \tilde{q}\left(\boldsymbol{x}\right)\mathrm{d}x}$
- Obtaining the alternative estimation $\hat{\theta}_{\mathrm{MLE}}$ of LRT can be computationally challenging
- Alternative solution to LRT?
  - Construct a hypothesis testing that estimate the alternative by minimizing the Hyvarinen score $\sum_{i=1}^{n} s_{\mathrm{H}}(\boldsymbol{x}_i, q_\theta)$
- We develop a new statistical test, referred to as the Hyvarinen score test (HST), based on the Hyvarinen score [1].

Duke

Hyvarinen Score Test

- We develop a new statistical test, referred to as the Hyvarinen score test (HST), based on the Hyvarinen score [1].

The Score-based Test Statistic

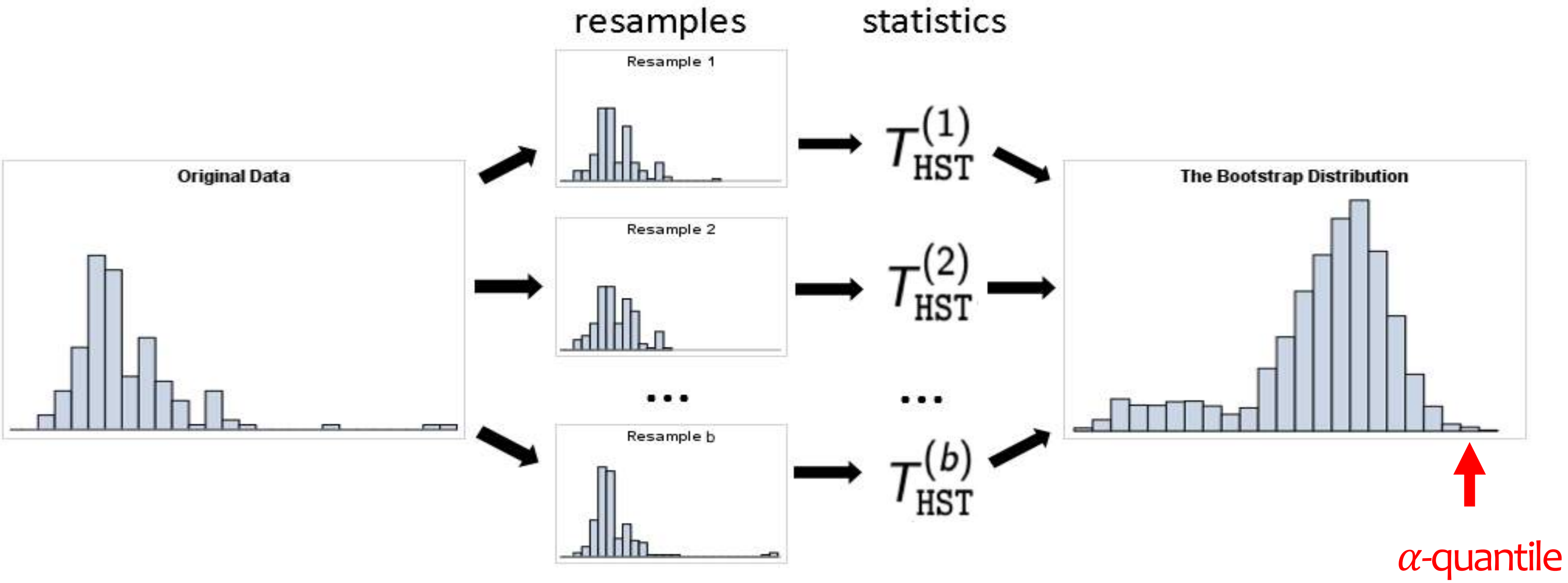
$$T_{\mathrm{HST}}(\boldsymbol{X}_n) \triangleq 2\left(S\left(\boldsymbol{X}_n, q_{\theta_0}\right) - S\left(\boldsymbol{X}_n, q_{\hat{\theta}}\right)\right)$$

- $S\left(\boldsymbol{X}_n, q\right) \triangleq \sum_{i=1}^{n} s_{\mathrm{H}}(\boldsymbol{x}_i, q)$, and $\hat{\theta}$ is learned by score matching.

- The HST rejects the null hypothesis when the test statistic T_{HST} is larger than some critical value, which can be identified using a large-sample asymptotic distribution.

Duke

Hyvarinen Score Test



- To determine the rejection region for HST in this case, we propose to use a bootstrap method developed in [2]
- The main idea is to approximate the critical value by the empirical α-quantile of the distribution of $T_{\mathrm{HST}}(\boldsymbol{X}_n)$ under the null hypothesis ($\alpha \in (0, 1)$, usually a small value)

Duke

Change-Point Detection

- Let $\{X_n\}_{n>1}$ denote a sequence of independent random observations
- Assume that for some unknown time instance ν, the observations
  - $X_1, X_2, \dots, X_{\nu-1}$ are IID according to $p_\infty(\{X_n\}_{n>1} \sim p_\infty$when $\nu = \infty)$
  - $X_\nu, X_{\nu+1}, \dots$ are IID according to $p_1(\{X_n\}_{n>1} \sim p_1$ when $\nu = 1)$
- We may intuitively think of p_∞ and p_1 as normal and abnormal observations distributions.
- A change detection rule T (the time of stop) is expected to detect ν as soon as possible but not raising a false alarm. (Let E_ν denote the expectation of p_ν)

Problem Definition [3]

minimize $sup_{\nu\geq 1} E_\nu[T-\nu|T \geq \nu]$ subject to $E_\infty[T] \geq \gamma$.

Duke

Score-based CUSUM (SCUSUM) Rule

Log Likelihood-based CUSUM Rule

$$T_{CUSUM} \equiv \inf\left\{n \geq 1: \max_{1 \leq k \leq n} \sum_{i=k}^{n} \left(\log p_1(\boldsymbol{x_i}) - \log p_\infty(\boldsymbol{x_i})\right) \geq \tau\right\}, \tau > 0.$$

Score-based CUSUM (SCUSUM) Rule

$$T_{SCUSUM} \equiv \inf\left\{n \geq 1: \max_{1 \leq k \leq n} \sum_{i=k}^{n} \lambda\left(s_{\mathrm{H}}(\boldsymbol{x}_i, p_1) - s_{\mathrm{H}}(\boldsymbol{x}_i, p_\infty)\right) \geq \tau\right\}, \tau > 0.$$

Duke

Application to Out-of-distribution Detection

Out-of-distribution Detection

- The aggregate Hyvarinen score $S\,(\boldsymbol{Y}_n, \hat{q}) = \sum_{i=1} s_{\mathrm{H}}(\boldsymbol{y}_i, \hat{q})$ is used for the change-point detection, where $\boldsymbol{Y}_n$ is the test data, the density function $\hat{q}$ is learned from the training (normal) data $\boldsymbol{X}_n$.
- We reject the in-distribution hypothesis when $S\,(\boldsymbol{Y}_n, \hat{q})$ is larger than some threshold, which can be decided by repeating the tests over the in-distribution training data.

Application to Network Intrusion Detection

- Consider the Network intrusion detection task on the KDD Cup 1999[1] dataset, which contains includes a wide variety of intrusions simulated in a military network environment [3].
- Implementation Details:
  - Train a Gauss-Bernoulli RBM with $\boldsymbol{X}_n$ to estimate the density function $q_{\hat{\theta}}$
  - Calculate $S\left(\boldsymbol{Y}_n, \hat{q}\right) = \sum_{i=1}^{n} s_{\mathrm{H}}\left(\boldsymbol{y}_i, \hat{q}\right)$, where $s_{\mathrm{H}}\left(\boldsymbol{y}_i, \hat{q}\right)$ has a closed-form for Gauss-Bernoulli RBM
  - Determine the threshold such that $\alpha = 0.05$.

[1]The Fifth International Conference on Knowledge Discovery and Data Mining. The data was collected by the 1998 DARPA Intrusion Detection Evaluation Program managed by MIT Lincoln Labs

Duke

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}(\boldsymbol{Y}_n, \hat{\theta})$ for detecting the "ipsweep" network attack.

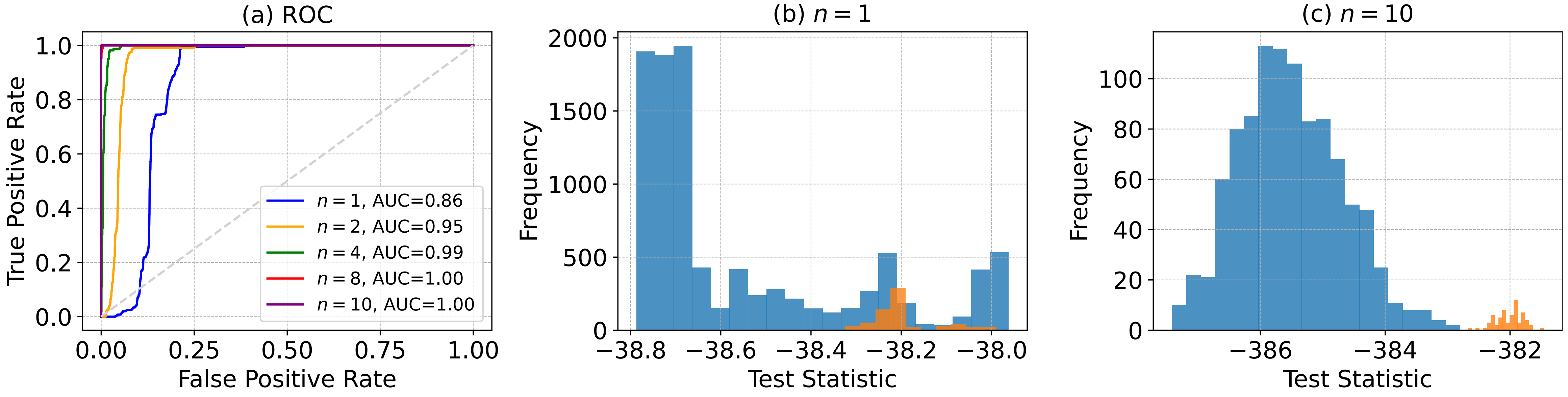


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "ipsweep" attack (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.

Network Intrusion Detection

The results demonstrate that our method can detect adversarial network attacks even with a single out-of-distribution data point. Naturally, our method's performance significantly improves when more out-of-distribution samples are available.

n (size)\ Attack Types	back	ipsweep	neptune	nmap	pod
1	0.785	0.869	0.896	0.835	0.802
2	0.895	0.961	0.986	0.946	0.933
4	0.937	0.997	1.000	0.993	0.983
8	**0.991**	**1.000**	**1.000**	**1.000**	**1.000**
10	**0.999**	**1.000**	**1.000**	**1.000**	**1.000**

n (size)\ Attack Types	portsweep	satan	smurf	teardrop	warezclient
1	0.921	0.928	0.818	0.882	0.645
2	0.979	0.983	0.942	0.963	0.731
4	1.000	1.000	0.972	0.996	0.803
8	**1.000**	**1.000**	**1.000**	**1.000**	**0.889**
10	**1.000**	**1.000**	**1.000**	**1.000**	**0.928**

Table 1: Area Under the Curve of Receiver Operating Characteristics **(AUC)** for our test to detect malicious network attack for various values of sample size **n**.

Duke

Reference

[1] S. Wu, E. Diao, K. Elkhalil, J. Ding and V. Tarokh, "Score-Based Hypothesis Testing for Unnormalized Models," in IEEE Access, vol. 10, pp. 71936-71950, 2022, doi: 10.1109/ACCESS.2022.3187991.

[2] Peter J. Bickel. David A. Freedman. "Some Asymptotic Theory for the Bootstrap." Ann. Statist. 9 (6) 1196 - 1217, November 1981. https://doi.org/10.1214/aos/1176345637.

[3]

[4] KDD Cup 1999 Data, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[5] S. J. Stolfo, Wei Fan, Wenke Lee, A. Prodromidis and P. K. Chan, "Cost-based modeling for fraud and intrusion detection: results from the JAM project," Proceedings DARPA Information Survivability Conference and Exposition. DISCEX'00, 2000, pp. 130-144 vol.2, doi: 10.1109/DISCEX.2000.821515.

Duke

Backup Slides

Hyvarinen Score Test Algorithm

Algorithm 1 Bootstrap Hyvärinen Score Test

Input: Test sample $\mathbf{X}_n \triangleq \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, number of bootstrap samples b, bootstrap sample size m, and significance level α

Independently sample $\mathbf{Y}_m \triangleq \{\mathbf{y}_1, \cdots, \mathbf{y}_m\}$ from the null distribution
for $i = 1, \ldots, b$ **do**
  Resample $\mathbf{Y}_n^{(i)}$ from $\mathbf{Y}_m$ with replacement
  Compute $T_{\mathtt{HST}}^{(i)} = 2\big(\mathcal{S}_{\mathtt{H}}(\mathbf{Y}_n^{(i)}, \theta_0) - \inf_{\theta\in\Theta} \mathcal{S}_{\mathtt{H}}(\mathbf{Y}_n^{(i)}, \theta)\big)$
end for
Determine $C_\alpha = \text{quantile}(\{T_{\mathtt{HST}}^{(1)}, \ldots, T_{\mathtt{HST}}^{(b)}\}, 1-\alpha)$
Compute $T_{\mathtt{HST}} = 2\big(\mathcal{S}_{\mathtt{H}}(\mathbf{X}_n, \theta_0) - \mathcal{S}_{\mathtt{H}}(\mathbf{X}_n, \hat{\theta}_{\mathtt{sm}})\big)$

- To determine the rejection region for HST in this case, we propose to use a bootstrap method developed in [2]
- The main idea is to approximate the critical value by the empirical α-quantile of the distribution of $T_{\mathrm{HST}}(\boldsymbol{X}_n)$ under the null hypothesis

Duke

SCUSUM Algorithm

Algorithm 1: SCUSUM Detection Algorithm

Input: Hyvarinen score functions $\mathcal{S}_{\mathrm{H}}(\cdot, P_\infty)$ and $\mathcal{S}_{\mathrm{H}}(\cdot, P_1)$ of pre- and post-change distributions, respectively.

Data: m previous observations $\mathbf{X}_{[-m+1,0]}$ and the online data stream $\{X_n\}_{n\geq 1}$

Initialization:

Current time $k = 0$, hyperparameter $\lambda > 0$, stopping threshold $\tau > 0$, and detection score $Z(0) = 0$

while $Z(k) < \tau$ **do**

$k = k + 1$

Update $z_\lambda(X_k) = \lambda(\mathcal{S}_{\mathrm{H}}(X_k, P_\infty) - \mathcal{S}_{\mathrm{H}}(X_k, P_1))$

Update $Z(k) = \max(Z(k-1) + z_\lambda(X_k), 0)$

Record the current time k as the stopping time $\hat{T}$

Locate the change point by $\hat{\nu} = \arg\min_{1\leq i\leq k} Z(i)$

Output: $\hat{T}$ and $\hat{\nu}$

- We proved the λ exists, and can be solved empirically by m previous observations.

Application to Network Intrusion Detection

Consider the Network intrusion detection task on the KDD Cup 1999[1] dataset, which contains includes a wide variety of intrusions simulated in a military network environment [3].

- Data Collection:
  - The raw TCP dump data was collected for a local-area network (LAN) simulating a typical U.S. Air Force LAN[2].
  - The binary raw data was then processed into connection records.
  - Stolfo et al. used domain knowledge to add features of connection records that look for suspicious behavior in the data portions [4].
  - Each connection is labeled as either “normal”, or as “attack”, with exactly one specific attack type.

[1]The Fifth International Conference on Knowledge Discovery and Data Mining

[2]The data was collected by the 1998 DARPA Intrusion Detection Evaluation Program managed by MIT Lincoln Labs

Duke

Features of connection records

Table 1: Basic features of individual TCP connections.

feature name	description	type
duration	length (number of seconds) of the connection	continuous
protocol_type	type of the protocol, e.g. tcp, udp, etc.	discrete
service	network service on the destination, e.g., http, telnet, etc.	discrete
src_bytes	number of data bytes from source to destination	continuous
dst_bytes	number of data bytes from destination to source	continuous
flag	normal or error status of the connection	discrete
land	1 if connection is from/to the same host/port; 0 otherwise	discrete
wrong_fragment	number of ``wrong'' fragments	continuous
urgent	number of urgent packets	continuous

Duke

Table 2: Content features within a connection suggested by domain knowledge.

feature name	description	type
hot	number of ``hot'' indicators	continuous
num_failed_logins	number of failed login attempts	continuous
logged_in	1 if successfully logged in; 0 otherwise	discrete
num_compromised	number of ``compromised'' conditions	continuous
root_shell	1 if root shell is obtained; 0 otherwise	discrete
su_attempted	1 if ``su root'' command attempted; 0 otherwise	discrete
num_root	number of ``root'' accesses	continuous
num_file_creations	number of file creation operations	continuous
num_shells	number of shell prompts	continuous
num_access_files	number of operations on access control files	continuous
num_outbound_cmds	number of outbound commands in an ftp session	continuous
is_hot_login	1 if the login belongs to the ``hot'' list; 0 otherwise	discrete
is_guest_login	1 if the login is a ``guest''login; 0 otherwise	discrete

Table 3: Traffic features computed using a two-second time window.

feature name	description	type
count	number of connections to the same host as the current connection in the past two seconds	continuous
	Note: The following features refer to these same-host connections.	
serror_rate	% of connections that have ``SYN'' errors	continuous
rerror_rate	% of connections that have ``REJ'' errors	continuous
same_srv_rate	% of connections to the same service	continuous
diff_srv_rate	% of connections to different services	continuous
srv_count	number of connections to the same service as the current connection in the past two seconds	continuous
	Note: The following features refer to these same-service connections.	
srv_serror_rate	% of connections that have ``SYN'' errors	continuous
srv_rerror_rate	% of connections that have ``REJ'' errors	continuous
srv_diff_host_rate	% of connections to different hosts	continuous

Citation: KDD Cup 1999 Data, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

Duke

Application to Network Intrusion Detection

- Type of Attacks:
  - A total of 24 training attack types in the training data with an additional 14 types in the test data.
  - Attacks fall into four main categories:
    - DOS: denial-of-service, such as “syn flood”;
      - back, land, neptune, pod, smurf, teardrop
    - R2L: unauthorized access from a remote machine, such as “guessing password”;
      - ftp write, guess passwwd, imap, multihop, phf, spy, wareclient, warezmaster
    - U2R: unauthorized access to local superuser (root) privileges, such as various “buer overflow” attacks;
      - buffer overflow, loadmodule, perl, rootkit
    - probing: surveillance and other probing, such as port scanning.
      - ipsweep, nmap, portsweep, satan

Citation: KDD Cup 1999 Data, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html

Duke

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}(\boldsymbol{Y}_n, \hat{\theta})$ for detecting the "back" network attack.

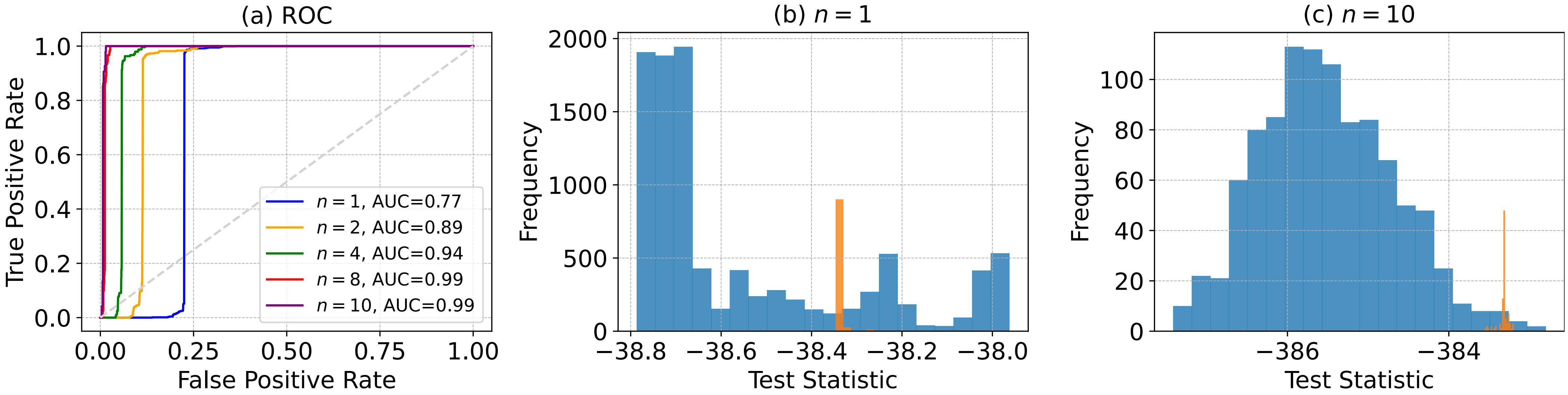


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "back" attack (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}\left(\boldsymbol{Y}_n, \hat{\theta}\right)$ for detecting the "neptune" network attack.

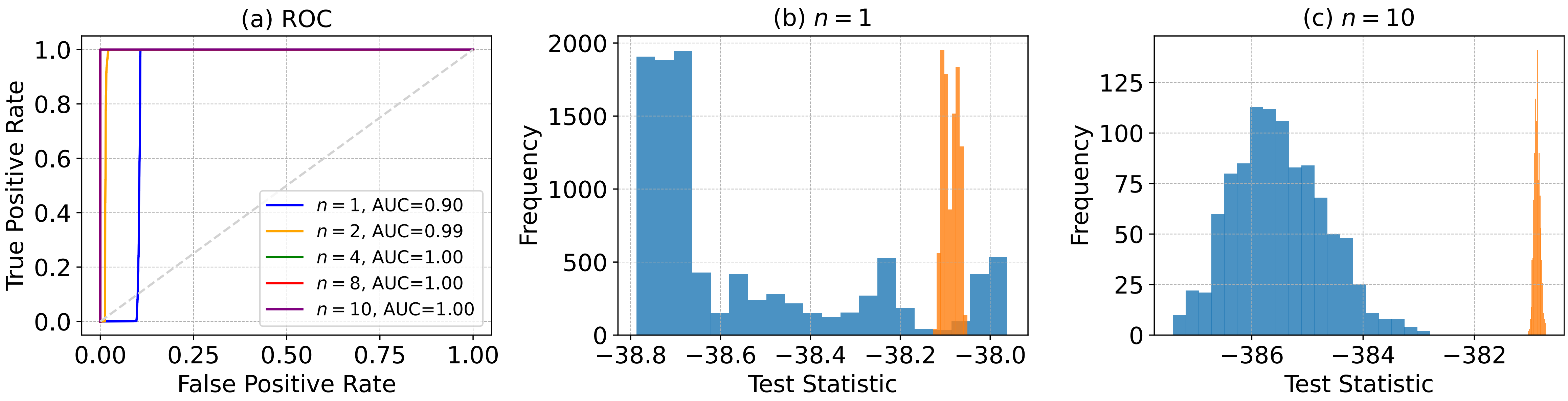


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "neptune" attack (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.

Duke

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}(\boldsymbol{Y}_n, \hat{\theta})$ for detecting the "nmap" network attack.

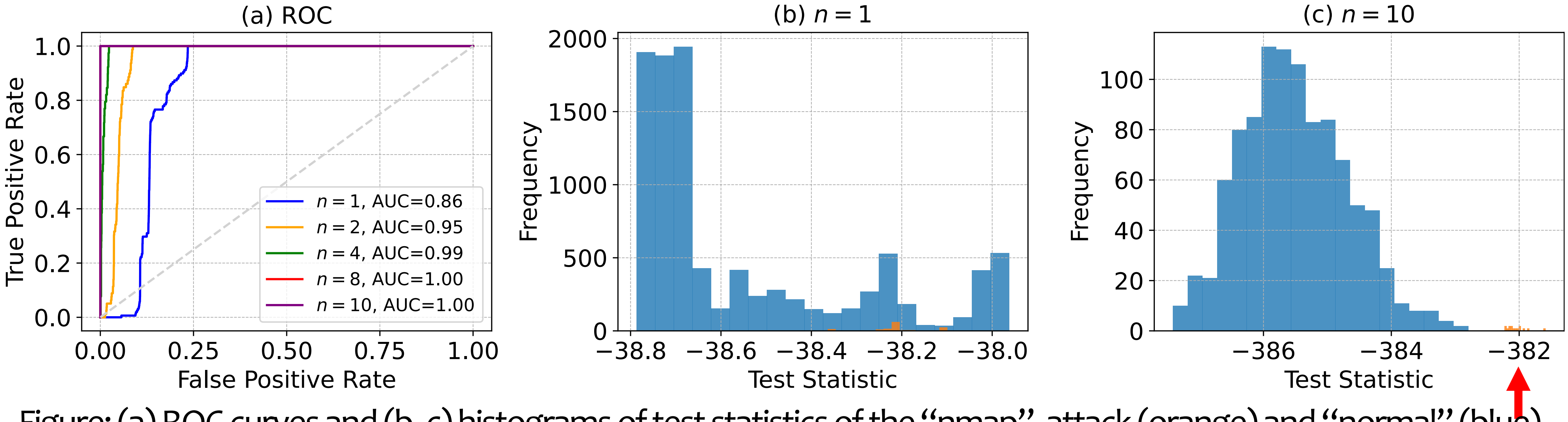


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "nmap" attack (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.

Duke

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}(\boldsymbol{Y}_n, \hat{\theta})$ for detecting the "pod" network attack.

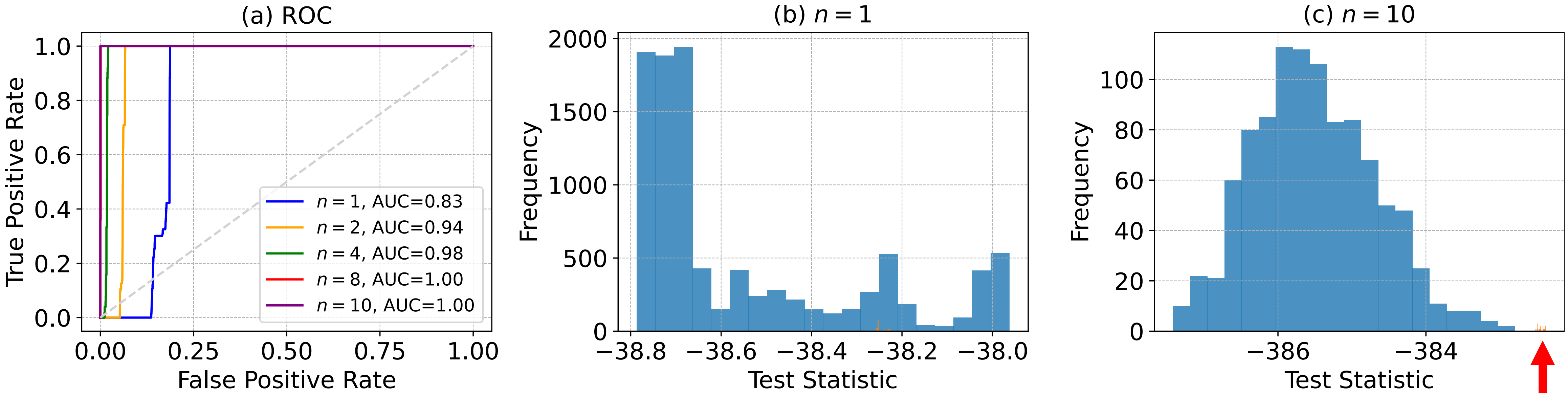


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "pod" attack (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.

Duke

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}(\boldsymbol{Y}_n, \hat{\theta})$ for detecting the "portsweep" network attack.

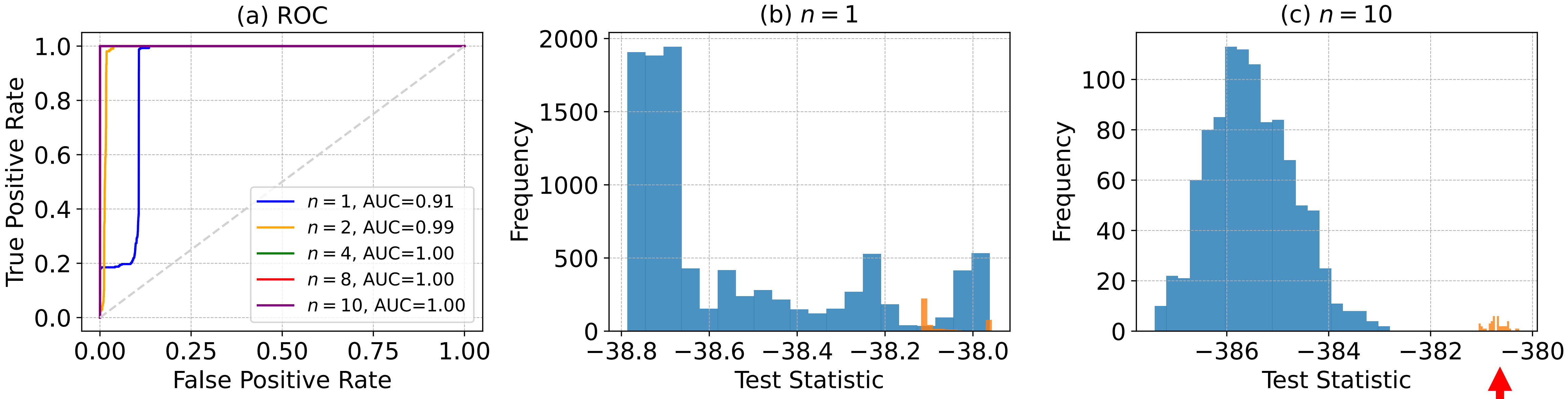


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "portsweep" attack (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.

Duke

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}(\boldsymbol{Y}_n, \hat{\theta})$ for detecting the "satan" network attack.

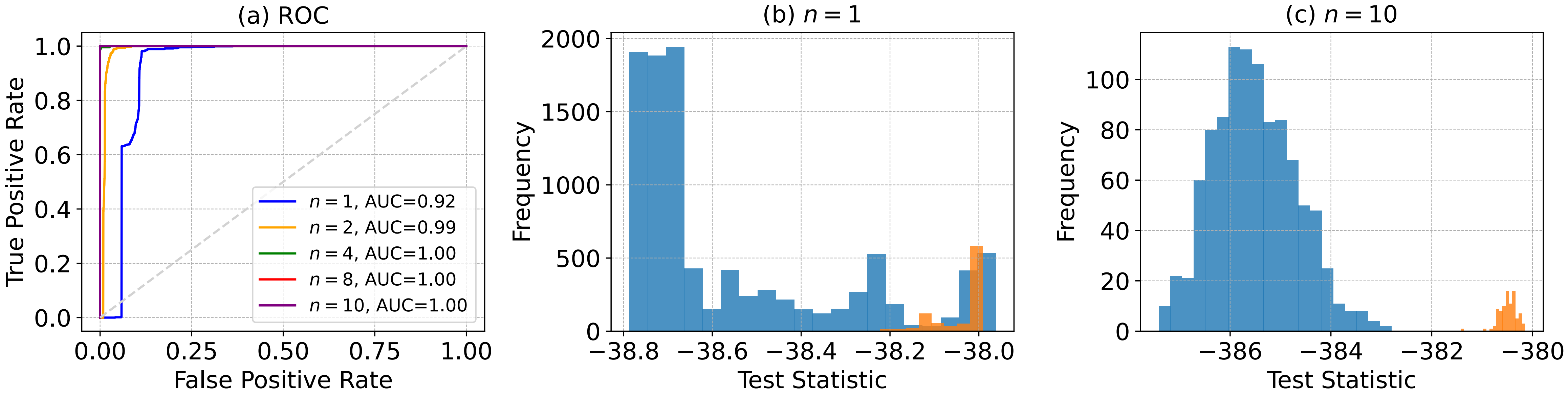


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "satan" attack (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.

[1]Satan is a tool designed to probe a computer system for security loopholes, (Security Administrator Tool for Analyzing Networks).

Duke

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}(\boldsymbol{Y}_n, \hat{\theta})$ for detecting the "smurf" network attack.

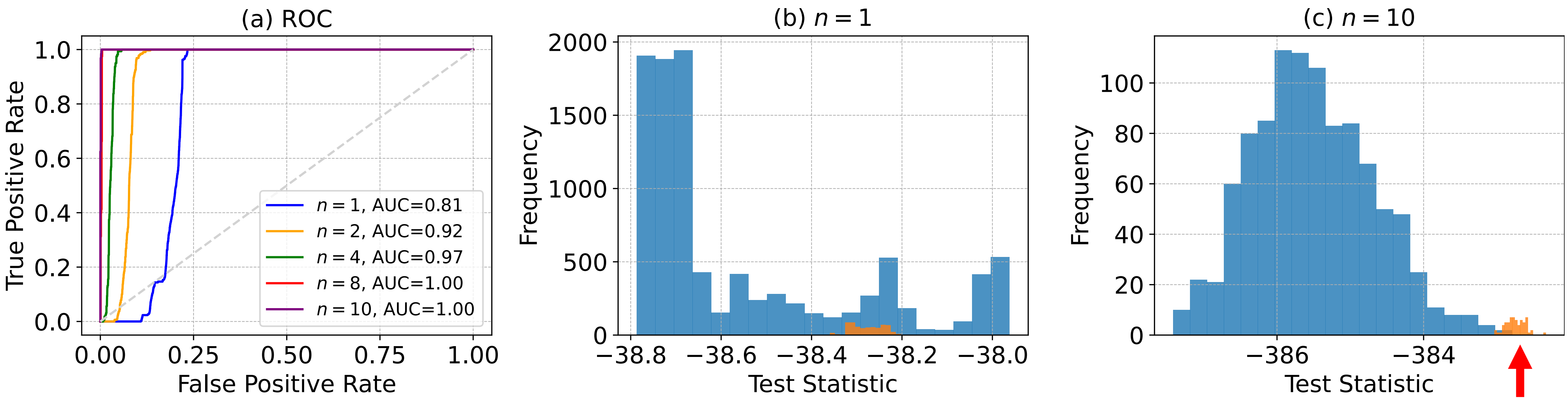


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "smurf" attack (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.

Duke

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}(\boldsymbol{Y}_n, \hat{\theta})$ for detecting the "teardrop" network attack.

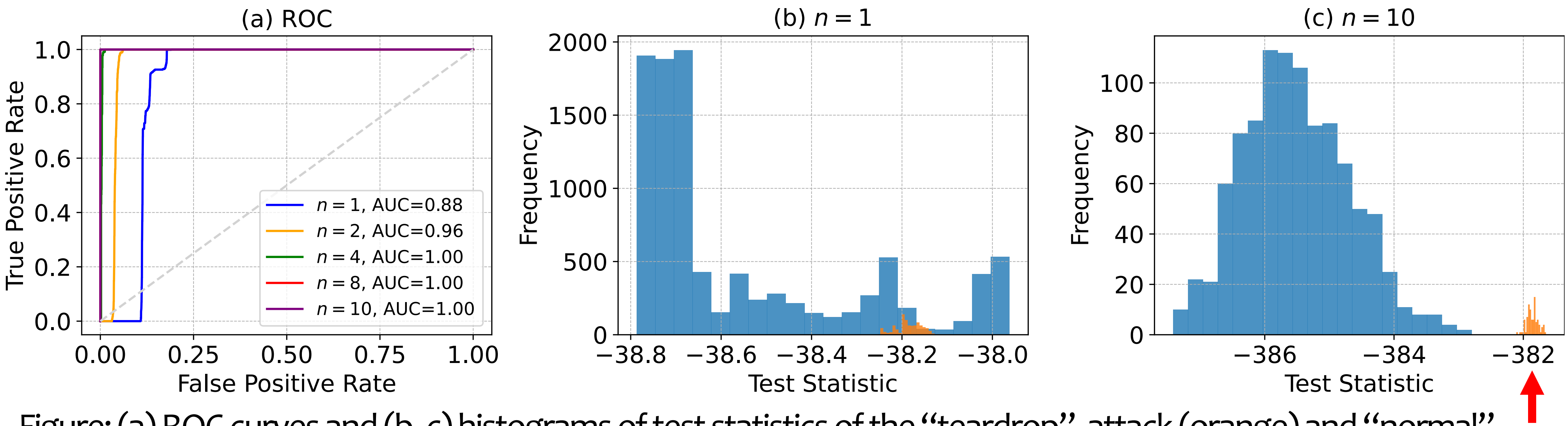


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "teardrop" attack (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.

Duke

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}(\boldsymbol{Y}_n, \hat{\theta})$ for detecting the "warezclient" network attack.

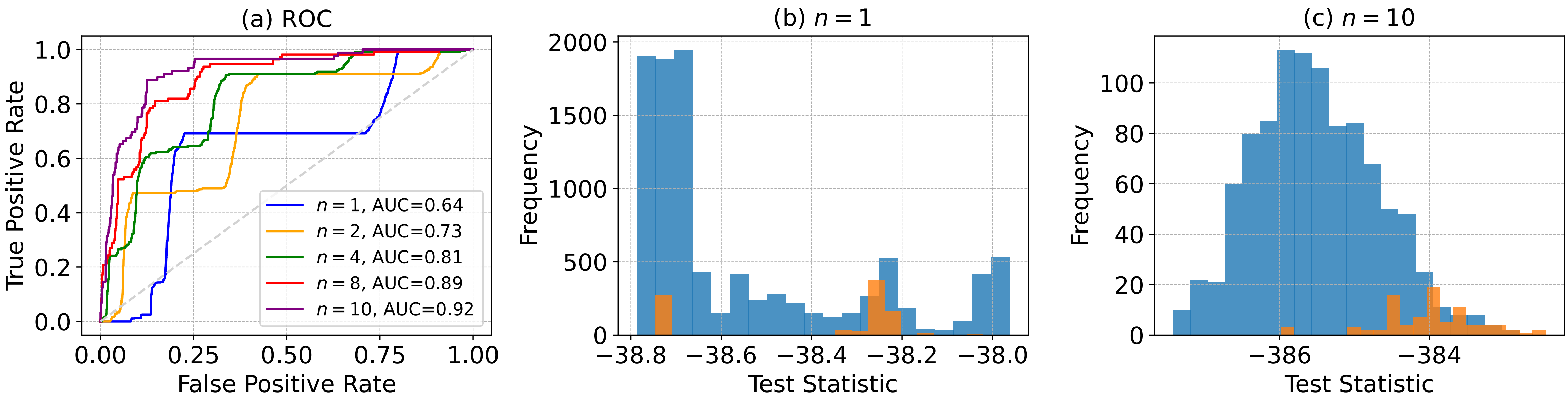


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "warezclient" attack (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.

Duke

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}(\boldsymbol{Y}_n, \hat{\theta})$ for detecting the "unknown" network attack.

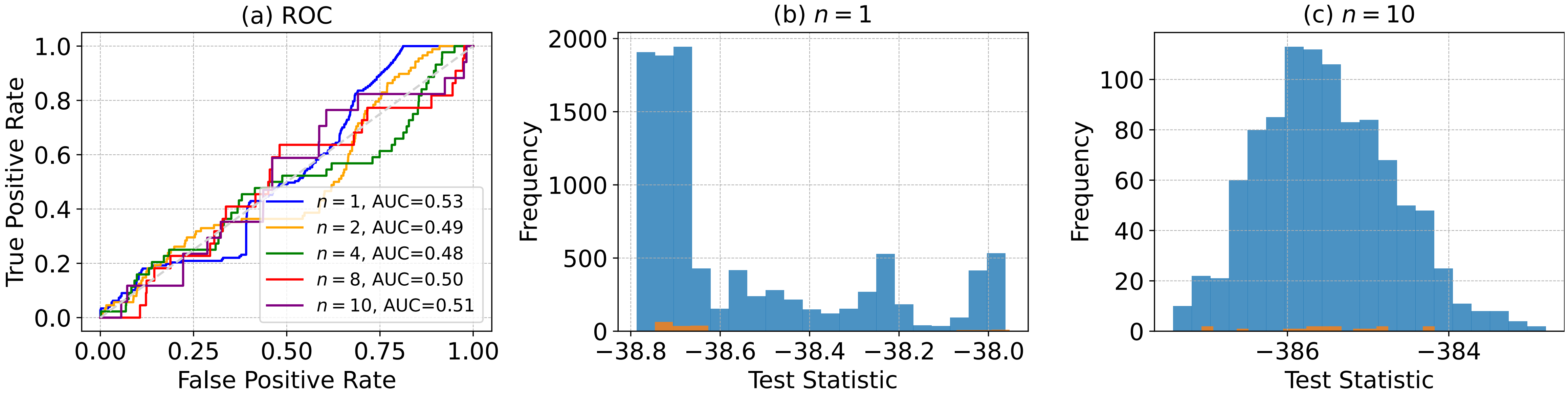


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "unknown" attack (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.

Duke

Network Intrusion Detection

From the figure below, we depict the ROC curves and the histograms of $S_{\mathrm{H}}(\boldsymbol{Y}_n, \hat{\theta})$ for detecting the "normal" network (positive labels).

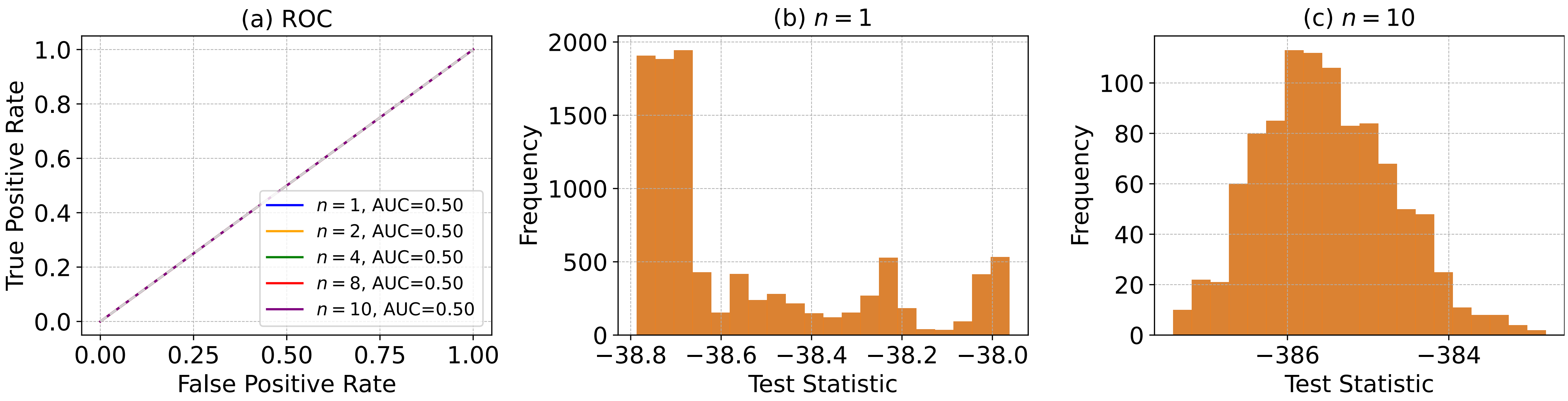


Figure: (a) ROC curves and (b, c) histograms of test statistics of the "normal" (orange) and "normal" (blue) network of HST on KDD Cup 1999 dataset.