
Image courtesy: Asset Guardian (https://www.assetguardian.com/how-ai-is-changing-the-cyber-security-landscape/)

# Quantitative Analysis of Machine Learning Classification Algorithms for the Design of Intrusion Detection/Prevention Systems.

## Research Interest

Intersection of

- Cybersecurity
- Digital Forensics
- Data Science
- Machine Learning

## Current Research

Focuses on extracting and analyzing data patterns created by deleted and decaying digital files, and their application in digital forensics investigations to reconstruct previous user activity.


# O. Cheche Agada, Ph.D.

Assistant Professor
Information Sciences and Technology
College of Engineering and Computing
George Mason University


- Ph.D. in Information Technology from George Mason University.
- MSc in Digital Forensics from George Mason University.
- MSc in Computer and Information Sciences from Southern Arkansas University.
- BSc in Computer Science from the University of Lagos, Nigeria.

# Agenda

Intrusion Detection/Prevention Systems

Problem Description/Statement

Algorithms & Metrics

Dataset

Methodology

Results

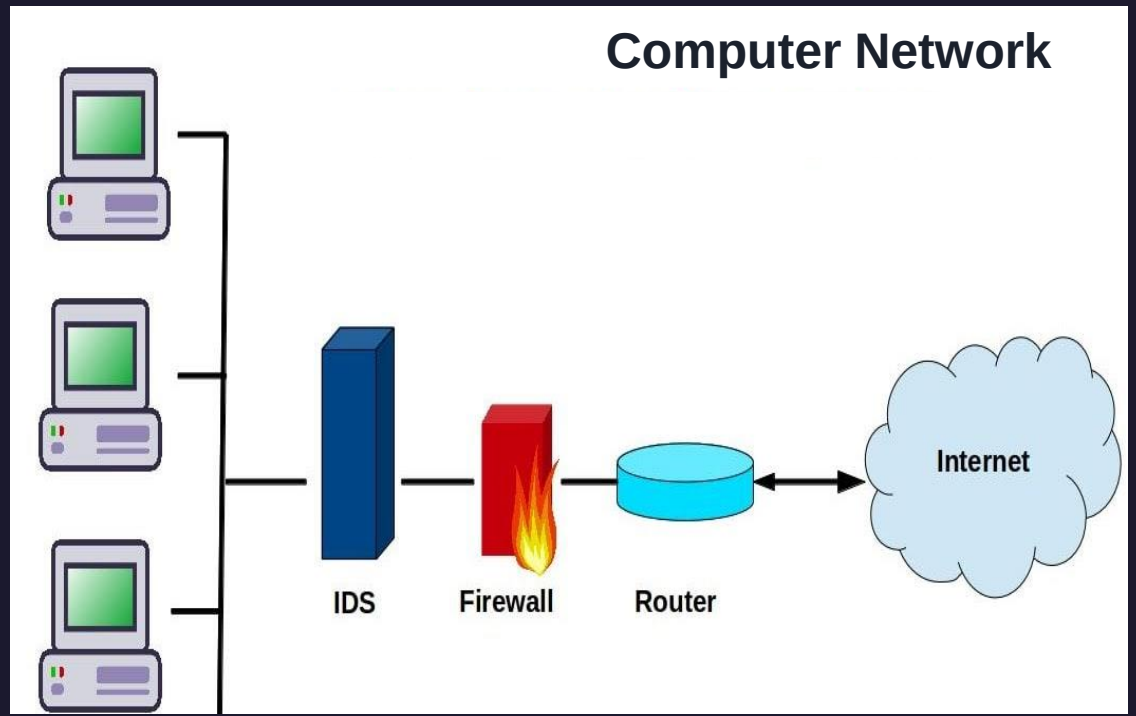
Conclusion

# Intrusion Detection/Prevention Systems

# Intrusion Detection/Prevention Systems

## What are IDS/IPS?

- Device or application that monitors network traffic.
- Detects/prevents malicious activities.
- Prevents security policy violations.




# Problem Description

# Problem Description

## Legacy IDS/IPS

- Legacy IDSs use static databases.
- They are non-adaptive.
- They are unable to detect previously unknown attacks.
- Don't deal very well with the complexities of modern cyber-attacks.

## M/L IDS/IPS

- Machine Learning-Based IDSs use self-learning algorithms.
- They are dynamic and adaptive.
- They can detect previously unknown attacks.
- Can deal with the complexities of modern cyber-attacks.
- Higher detection rates, lower false alarm rates, and reasonable computation and communication costs.

# Problem Statement

The goal is to quantitatively evaluate a group of Machine Learning classification algorithms, to determine the most efficient machine learning algorithm for designing an IDS/IPS.

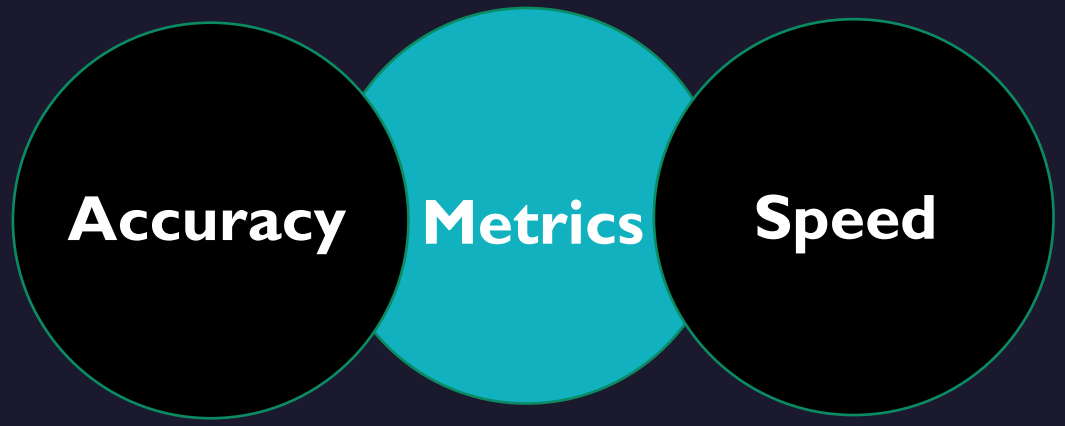
# Algorithms & Metrics

# Algorithms & Metrics

## Algorithms

- Logistic Regression
- Naïve Bayes
- Decision Tree
- K-Nearest Neighbor
- Support Vector Machine

## Metrics



# Dataset

# Dataset – KDD CUP 99 → NSL-KDD

| Attack Type | Count |
|---|---|
| Normal | 67343 |
| Dos (Denial of Service) | 45927 |
| R2L (Remote to Local) | 11656 |
| Probe | 995 |
| U2R (User to Root) | 52 |
| Total | 125973 |

Table 1. Breakdown of attacks in the Training Set

| Traffic Type | Count |
|---|---|
| Normal | 67343 |
| Attack | 58630 |
| Total | 125973 |

Table 2. Breakdown of Traffic Type – Training Set

# Dataset – KDD CUP 99 → NSL-KDD

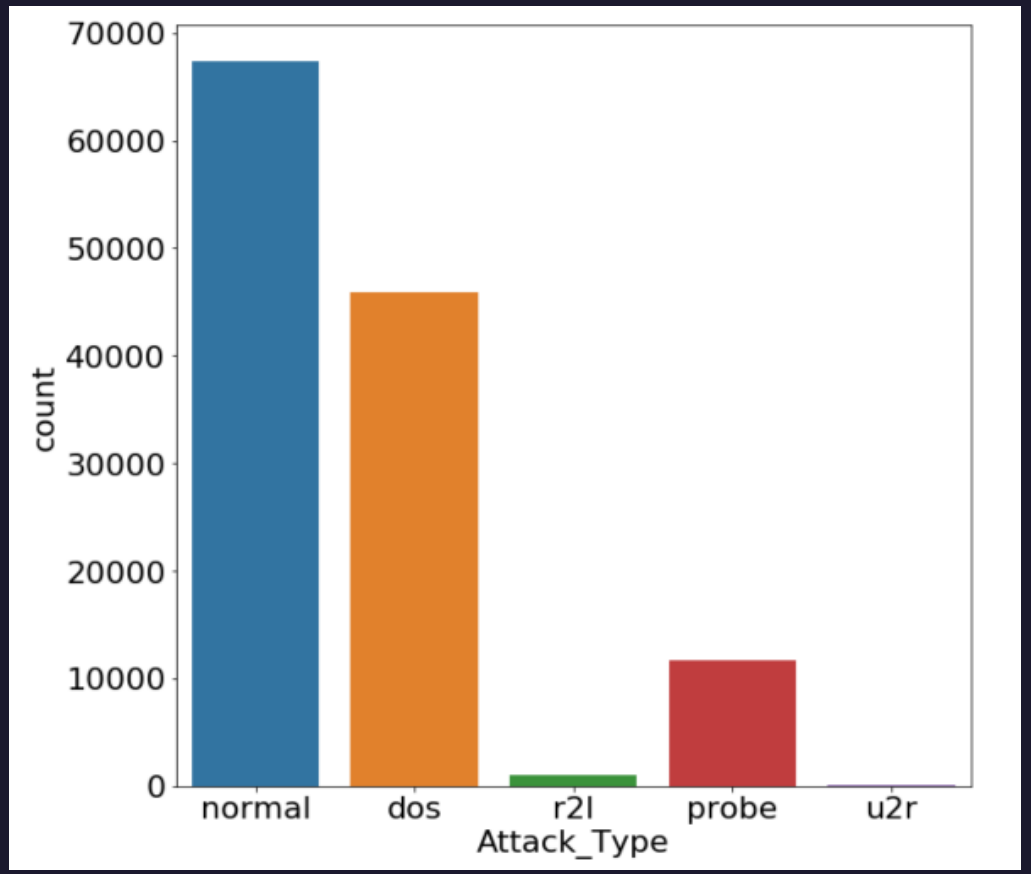



**Figure 1. Training Set**

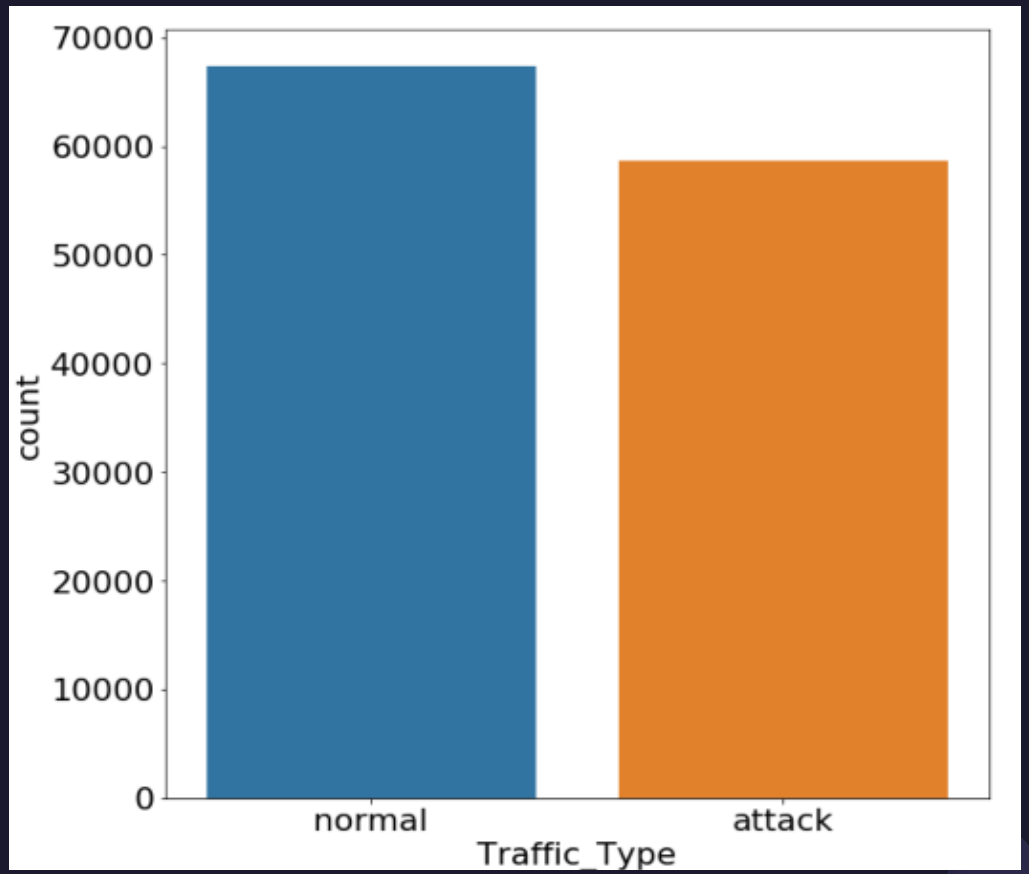



**Figure 2. Traffic Type – Training Set**

# Dataset – KDD CUP 99 → NSL-KDD

| Attack Type | Count |
|---|---|
| Normal | 9710 |
| Dos (Denial of Service) | 7456 |
| R2L (Remote to Local) | 2754 |
| Probe | 2421 |
| U2R (User to Root) | 202 |
| Total | **22543** |

Table 3. Breakdown of attacks in the Test Set

| Attack Type | Count |
|---|---|
| Normal | 9710 |
| Attack | 12833 |
| Total | **22543** |

Table 4. Breakdown of Traffic Type – Test Set

# Dataset – KDD CUP 99 → NSL-KDD

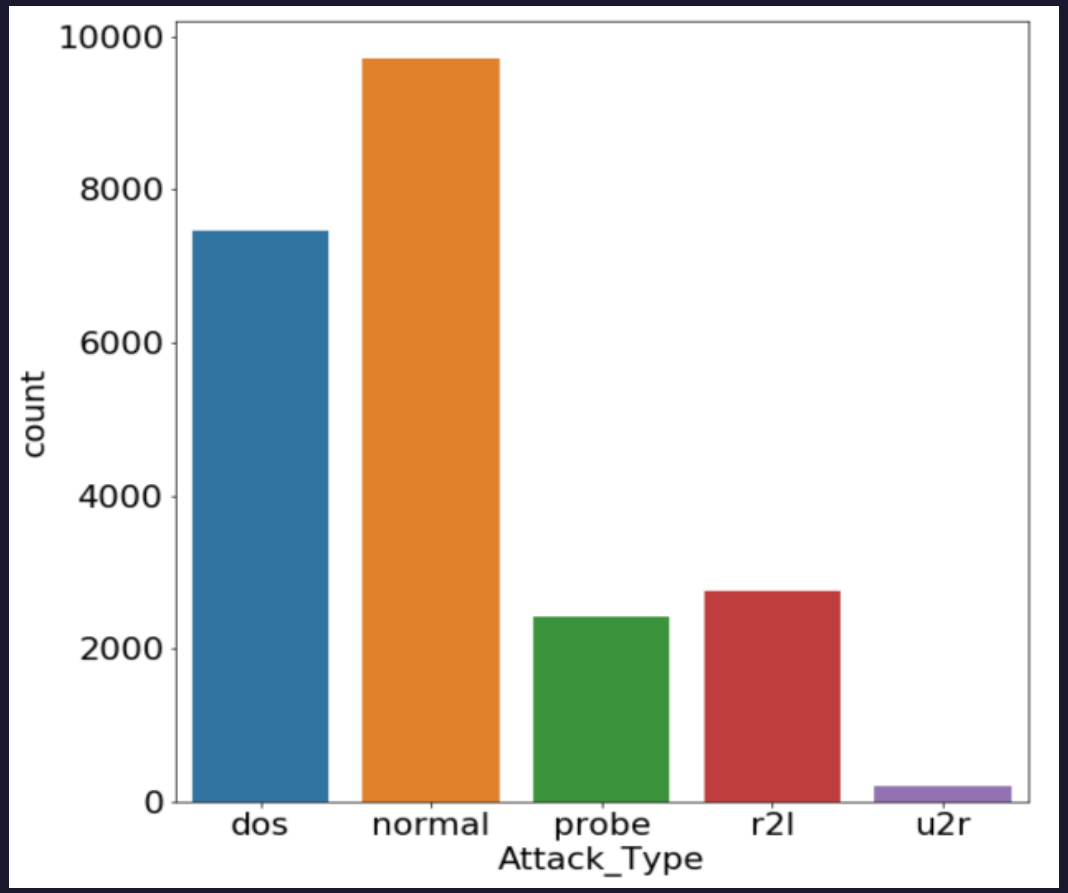



Figure 3. Test Set

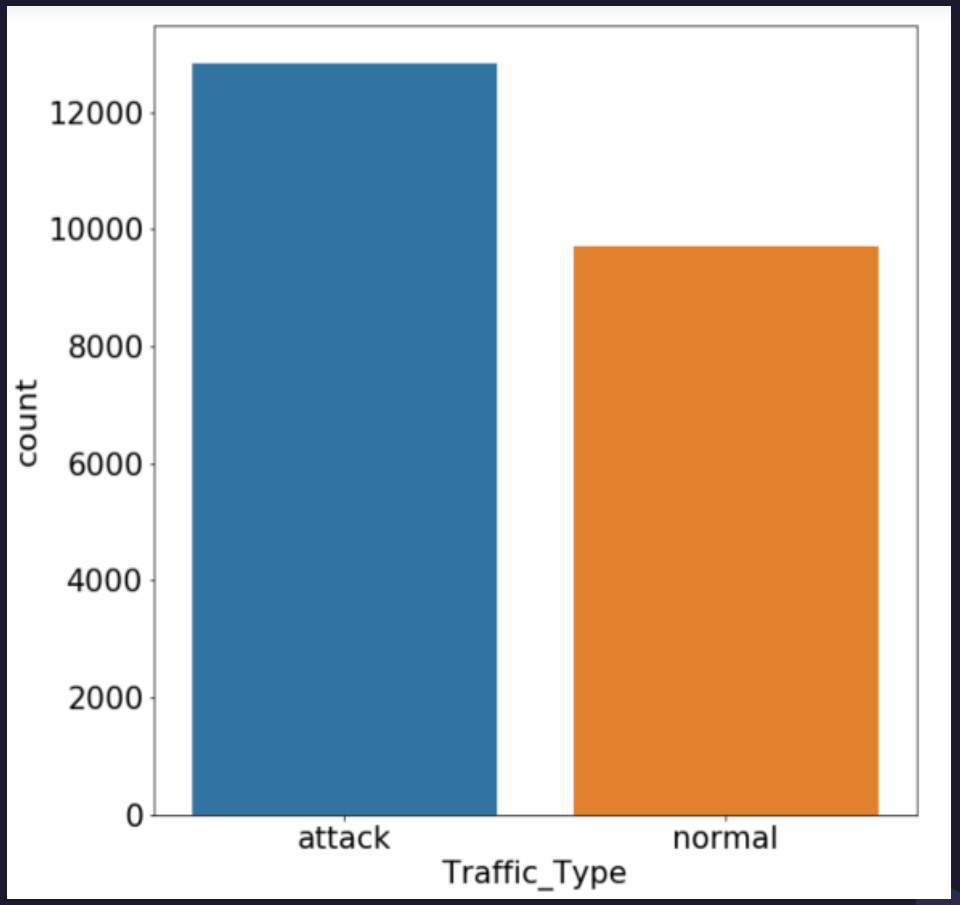



Figure 4. Traffic Type – Test Set

# Dataset – KDD CUP 99 → NSL-KDD

| # | Description | # | Description |
|---|---|---|---|
| 1 | duration | 12 | su_attempted |
| 2 | src_bytes | 13 | num_root |
| 3 | dst_bytes | 14 | num_file_creations |
| 4 | Land | 15 | num_shells |
| 5 | wrong_fragment | 16 | num_access_files |
| 6 | urgent | 17 | num_outbound_cmds |
| 7 | hot | 18 | is_host_login |
| 8 | num_failed_logins | 19 | is_guest_login |
| 9 | logged_in | 20 | count |
| 10 | num_compromised | 21 | srv_count |
| 11 | root_shell | 22 | serror_rate |

| # | Description | # | Description |
|---|---|---|---|
| 23 | srv_serror_rate | 34 | dst_host_srv_diff_host_rate |
| 24 | rerror_rate | 35 | dst_host_serror_rate |
| 25 | srv_rerror_rate | 36 | dst_host_srv_serror_rate |
| 26 | same_srv_rate | 37 | dst_host_rerror_rate |
| 27 | diff_srv_rate | 38 | dst_host_srv_rerror_rate |
| 28 | srv_diff_host_rate | 39 | protocol_type |
| 29 | dst_host_count | 40 | service |
| 30 | dst_host_srv_count | 41 | flag |
| 31 | dst_host_same_srv_rate | | |
| 32 | dst_host_diff_srv_rate | | |
| 33 | dst_host_same_src_port_rate | | |

**Features of the NSL-KDD Dataset**

# Methodology

# Methodology

## Phase 1

- Train all 5 algorithms using training set.
- Run each algorithm over test set to generate predictions.
- Compare predictions with observed classes.
- Determine accuracy of predictions.

## Phase 2

- Retrain best 2 algorithms from phase 1.
- Generate 50 random samples from test set.
- Run both algorithm on each sample to generate predictions.
- Record execution time for each sample.
- Compare performance of both algorithms using 95% CI estimation.

# Results

# Results – Algorithm Accuracy – Phase 1

| S/N | ALGORITHM | ACCURACY |
|---|---|---|
| 1 | Decision Tree | 80% |
| 2 | SVM | 80% |
| 3 | K-Nearest Neighbor | 79% |
| 4 | Naïve Bayes | 77% |
| 5 | Logistic Regression | 75% |

# Results – Classification Reports – Phase 1

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.92 | 0.61 | 0.73 | 12833 |
| 1 | 0.64 | 0.93 | 0.76 | 9710 |
| accuracy | | | 0.75 | 22543 |
| macro avg | 0.78 | 0.77 | 0.75 | 22543 |
| weighted avg | 0.80 | 0.75 | 0.75 | 22543 |

Logistic Regression

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.97 | 0.68 | 0.80 | 12833 |
| 1 | 0.69 | 0.97 | 0.81 | 9710 |
| accuracy | | | 0.80 | 22543 |
| macro avg | 0.83 | 0.82 | 0.80 | 22543 |
| weighted avg | 0.85 | 0.80 | 0.80 | 22543 |

Decision Tree

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.98 | 0.66 | 0.79 | 12833 |
| 1 | 0.69 | 0.98 | 0.81 | 9710 |
| accuracy | | | 0.80 | 22543 |
| macro avg | 0.83 | 0.82 | 0.80 | 22543 |
| weighted avg | 0.85 | 0.80 | 0.80 | 22543 |

Support Vector Machine

# Results – Classification Reports – Phase 1

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.91 | 0.67 | 0.77 | 12833 |
| 1 | 0.68 | 0.92 | 0.78 | 9710 |
| accuracy | | | 0.77 | 22543 |
| macro avg | 0.79 | 0.79 | 0.77 | 22543 |
| weighted avg | 0.81 | 0.77 | 0.77 | 22543 |

Naïve Bayes

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.97 | 0.64 | 0.77 | 12833 |
| 1 | 0.67 | 0.98 | 0.80 | 9710 |
| accuracy | | | 0.79 | 22543 |
| macro avg | 0.82 | 0.81 | 0.79 | 22543 |
| weighted avg | 0.84 | 0.79 | 0.78 | 22543 |

K-Nearest Neighbor

# Results – System Comparison – Phase 2

The execution times shown below are in seconds.

| S/N | Decision Tree (A) | Support Vector Machine (B) | A - B |
|---|---|---|---|
| 1 | 0.000999 | 1.196236 | -1.19524 |
| 2 | 0.000998 | 1.218428 | -1.21743 |
| 3 | 0.000997 | 1.190477 | -1.18948 |
| 4 | 0.001 | 1.276542 | -1.27554 |
| 5 | 0.001032 | 1.186174 | -1.18514 |
| 6 | 0.000999 | 1.190952 | -1.18995 |
| 7 | 0.000994 | 1.232021 | -1.23103 |
| 8 | 0.000955 | 1.180771 | -1.17982 |
| 9 | 0.001001 | 1.18945 | -1.18845 |
| 10 | 0.001993 | 1.266891 | -1.2649 |

**Execution Times**

| Quantity | Value | Additional Comments |
|---|---|---|
| x-bar (DT - SVM) | -1.1908781 | Mean of (DT - SVM) |
| s(DT - SVM) | 0.02275773 | Standard Dev of (DT - SVM) |
| n | 50 | Sample size |
| Alpha | 0.05 | Significance level |
| 1 - Alpha/2 | 0.975 | Confidence coefficient |
| Z1-Alpha/2 | 1.95996398 | Inverse of the CDF |
| 1/2CI | 0.006308 | Half confidence interval |
| **C1** | **-1.1971861** | Lower bound |
| **C2** | **-1.1845701** | Upper Bound |

**Table 7: 95% Confidence Interval Computation.**

**Confidence Interval Computations**

$$(\bar{x} - z_{1-\alpha/2}\frac{s}{\sqrt{n}}, \bar{x} + z_{1-\alpha/2}\frac{s}{\sqrt{n}})$$

$\bar{x}$ : sample mean
s: sample standard deviation
n: sample size
$z_{1-\alpha/2}$ : (1-α/2)-quantile of a unit normal variate ( N(0,1)).

**Confidence Interval Parameters**

# Conclusion

# Conclusion

| Interval | Value |
|---|---|
| C1 | -1.1971861 |
| C2 | -1.1845701 |

- The interval does not contain zero, therefore both algorithms are not the same.
- They are both negative which indicates that the value of B (SVM) is higher than A (DT).
- Since we are dealing with time, the smaller the better.
- Therefore, DT is faster than SVM.
- This is consistent with established facts in IDS literature.