The KDD Datasets

Software to detect network intrusions protects a computer network from unauthorized users, including insiders. The intrusion detector learning task is to build a predictive model (a classifier) capable of distinguishing between "bad" connections, called intrusions or attacks, and "good" normal connections.

KDD99

Since 1997 the ACM Special Interest Group on Knowledge Discovery and Data Mining has organized an annual KDD Cup competition. The dataset for KDD99 was created by processing raw packet (tcpdump) data created by MIT Lincoln Labs for the 1998 DARPA Intrusion Detection System (IDS) Evaluation dataset. The artificial data was generated using a closed network and injected (planned) attacks to produce a dataset with normal activity in the background of a variety of intrusions.

Around 4gb of compressed binary raw packet data from seven weeks of network traffic was processed into just under five million connection records for the training set, and another two weeks of test data yielded around two million connection records for the test set.

A connection is a sequence of TCP packets starting and ending at some well-defined times, between which data flows to and from a source IP address to a target IP address under some well- defined protocol. The KDD99 dataset has 4,898,431 connection records in the full train set and 311,027 in the test set; each connection record has 42 attributes, including a label of either "normal" or exactly one attack type.

To make the analysis task more realistic, the test data is not from the same probability distribution as the training data, and the test data has 17 unique attack types in addition to the 22 attack types in the training data.

To make the analysis task manageable, the 39 specific attack types are categorized into four general classes:

- Denial of Service (DoS): an attack where an adversary directs a flood of traffic requests to a system in order to push the computing or memory resources beyond their capacity to handle legitimate requests, which results in denying legitimate users access to the system.
- Reconnaissance (Probe): querying networked computers to gather information to be used to compromise security controls.
- Remote to Local attack (R2L): an attacker who has the ability to send packets to a host over a network without having an account on that system attempts to exploit an operating system or application vulnerability to gain access as a local user.
- User to Root attack (U2R): an adversary who has gained access to a normal user account on the system (by sniffing passwords, a dictionary attack, social engineering, or R2L for example) attempts to exploit an operating system or application vulnerability to gain root access to the system.

Scikit-learn “toy” KDD99

While the KDD99 dataset has long been used for machine learning and intrusion detection research, the most common criticism is the number of redundant records. Supervised learning algorithms may be biased towards the frequent records, preventing them from learning infrequent records which are usually more harmful to networks.

On one hand, since the initial goal was to produce a large training set for supervised learning algorithms, the proportion of abnormal data is unrealistic. 80% of the 4.9 million records in the KDD99 train set are attacks, but only 6.675% of these are unique; the corresponding proportions for the test set are 80% and 11.6%. On the other hand, the number of redundant records in the KDD99 dataset can be considered realistic since the DoS class of attack is based on sending too many requests for the target to handle, but it certainly creates statistical challenges.

The fetch function in scikit-learn addresses this redundancy by creating subsets with all of the normal data and a small proportion of attack data, or selecting only connection records with certain features, to create datasets appropriate for unsupervised anomaly detection.

NSL-KDD+

The NSL-KDD data set comes from the Canadian Institute for Cybersecurity, based at the University of New Brunswick.

Three subsets of fifty thousand records were randomly selected from the KDD train set, and seven classifiers were chosen from the Weka collection. Each of the classifiers was trained over each of the training sets, to create 21 models. Each model was then used to label the records of the entire KDD train and test sets, and increment a #successfulPrediction# counter (initialized to zero) for each record where the model prediction matched the label provided with the dataset.

To create the final datasets, each record of the KDD99 test and train datasets was annotated with its #successfulPrediction# value, and redundant records were removed. All but around 2% of the records in the train set and 14% of the records in the test set were correctly classified in all 21 cases, so a “no 21” subset was created with those records that proved difficult to identify.

References

http://kdd.ics.uci.edu/databases/kddcup99/task.html

Adapted from the paper Salvatore J. Stolfo, Wei Fan, Wenke Lee, Andreas Prodromidis, and Philip K. Chan. “Cost-based Modeling and Evaluation for Data Mining With Application to Fraud and Intrusion Detection: Results from the JAM Project”. discex, vol. 02, p. 1130, 2000. Available: http://ids.cs.columbia.edu/sites/default/files/ada511232.pdf

https://www.unb.ca/cic/datasets/nsl.html

Adapted from the paper M. Tavallaee, E. Bagheri, W. Lu, and A. Ghorbani, “A Detailed Analysis of the KDD CUP 99 Data Set,” proc. Second IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA), 2009. Available: https://www.ee.ryerson.ca/~bagheri/papers/cisda.pdf

https://scikit-learn.org/stable/datasets.html

NSL-boosted

The “boosted” train datasets are a modification of the NSL datasets. The “difficulty” feature was dropped from the original train set and the “no-21” test set, and the attack class (R2L, etc.) was added for each row.

All of the attack types in the train set were dropped from the “no-21” test set, leaving 21 attack types unique to the test set, and a subset of each of these was randomly selected to make a “booster” dataset. The proportions were based on the frequency of the attack type: 5% for the 3 attack types with over 600 records, 10% for the 4 attacks with 100-300 records, and 100% for the 5 attack types with 9-20 records.

Both the original NSL train set and the “booster” dataset were split 50-50 using stratified sampling, and the 7 attack types with 1-4 records were added to both halves of the “booster” dataset. Then two unique halves were combined to yield two training sets with all 39 attack types and an equal number of records. In effect, 1.3% of the test set has been “leaked” into the train set to give them a consistent list of attack types.