

Anomaly Detection

[< << Back to Technical Glossary](#)

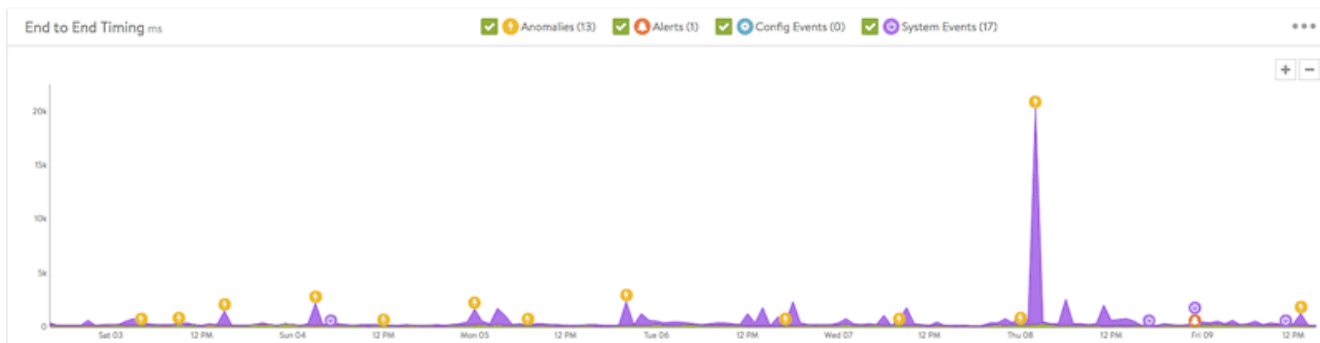
Anomaly Detection

Anomaly detection, also called outlier detection, is the identification of unexpected events, observations, or items that differ significantly from the norm. Often applied to unlabeled data by data scientists in a process called unsupervised anomaly detection, any type of anomaly detection rests upon two basic assumptions:

- Anomalies in data occur only very rarely
- The features of data anomalies are significantly different from those of normal instances

Typically, anomalous data is linked to some sort of problem or rare event such as hacking, bank fraud, malfunctioning equipment, structural defects / infrastructure failures, or textual errors. For this reason, identifying actual anomalies rather than false positives or data noise is essential from a business perspective.





Anomaly Detection FAQs

What is Anomaly Detection?

Anomaly detection is the identification of rare events, items, or observations which are suspicious because they differ significantly from standard behaviors or patterns. Anomalies in data are also called standard deviations, outliers, noise, novelties, and exceptions.

In the network anomaly detection/network intrusion and abuse detection context, interesting events are often not rare—just unusual. For example, unexpected jumps in activity are typically notable, although such a spurt in activity may fall outside many traditional statistical anomaly detection techniques.

Many outlier detection methods, especially unsupervised techniques, do not detect this kind of sudden jump in activity as an outlier or rare object. However, these types of micro clusters can often be identified more readily by a cluster analysis algorithm.

There are three main classes of anomaly detection techniques: unsupervised, semi-supervised, and supervised. Essentially, the correct anomaly detection method depends on the available labels in the dataset.

Supervised anomaly detection techniques demand a data set with a complete set of “normal” and “abnormal” labels for a classification algorithm to work with. This kind of technique also involves training the classifier. This is similar to traditional pattern recognition, except that with outlier detection there is a naturally strong imbalance between the classes. Not all statistical classification algorithms are well-suited for the inherently unbalanced nature of anomaly detection.

Semi-supervised anomaly detection techniques use a normal, labeled training data set to construct a model representing normal behavior. They then use that model to detect anomalies by testing how likely the model is to generate any one instance encountered.

Unsupervised methods of anomaly detection detect anomalies in an unlabeled test set of data based solely on the intrinsic properties of that data. The working assumption is that, as in most cases, the large majority of the instances in the data set will be normal. The anomaly detection algorithm will then detect instances that appear to fit with the rest of the data set least congruently.

What Are Anomalies?

Anomalies can be classified generally in several ways:

Network anomalies: Anomalies in network behavior deviate from what is normal, standard, or expected. To detect network anomalies, network owners must have a concept of expected or normal behavior. Detection of anomalies in network behavior demands the continuous monitoring of a network for unexpected trends or events.

Application performance anomalies: These are simply anomalies detected by end-to-end application performance monitoring. These systems observe application function, collecting data on all problems, including supporting infrastructure and app dependencies. When anomalies are detected, rate limiting is triggered and admins are notified about the source of the issue with the problematic data.

Web application security anomalies: These include any other anomalous or suspicious web application behavior that might impact security such as XSS attacks or DDOS attacks.

Detection of each type of anomaly relies on ongoing, automated monitoring to create a picture of normal network or application behavior. This type of monitoring might focus on point anomalies/global outliers, contextual anomalies, and/or collective anomalies; the context of the network, the performance of the application, or the [web application security](#) is more important to the goal of the anomaly detection system.

Anomaly detection and novelty detection or noise removal are similar, but distinct. Novelty detection identifies patterns in data that were previously unobserved so users can determine whether they are anomalous. Noise removal is the process of removing noise or unneeded observations from a signal that is otherwise meaningful.

To track monitoring KPIs such as bounce rate and churn rate, time series data anomaly detection systems must first develop a baseline for normal behavior. This enables the system to track seasonality and cyclical behavior patterns within key datasets.



Why Anomaly Detection Is Important

It is critical for network admins to be able to identify and react to changing operational conditions. Any nuances in the operational conditions of data centers or cloud applications can signal unacceptable levels of business risk. On the other hand, some divergences may point to positive growth.

Therefore, anomaly detection is central to extracting essential business insights and maintaining core operations. Consider these patterns—all of which demand the ability to discern between normal and abnormal behavior precisely and correctly:

- An online retail business must predict which discounts, events, or new products may trigger boosts in sales which will increase demand on their web servers.
 - An IT security team must prevent hacking and needs to detect abnormal login patterns and user behaviors.
 - A cloud provider has to allot traffic and services and has to assess changes to infrastructure in light of existing patterns in traffic and past resource failures.
- A evidence-based, well-constructed behavioral model can not only represent data behavior, but also help users identify outliers and engage in meaningful predictive analysis. Static alerts and thresholds are not enough, because of the overwhelming scale of the operational parameters, and because it's too easy to miss anomalies in false positives or negatives.

To address these kinds of operational constraints, newer systems use smart algorithms for identifying outliers in seasonal time series data and accurately forecasting periodic data patterns.

Anomaly Detection Techniques

In searching data for anomalies that are relatively rare, it is inevitable that the user will encounter relatively high levels of noise that could be similar to abnormal behavior. This is because the line between abnormal and normal behavior is typically imprecise, and may change often as malicious attackers adapt their strategies.

Furthermore, because many data patterns are based on time and seasonality, there is additional baked-in complexity to anomaly detection techniques. The need to break down multiple trends over time, for example, demands more sophisticated methods to identify actual changes in seasonality versus noise or anomalous data.

For all of these reasons, there are various anomaly detection techniques. Depending on the circumstances, one might be better than others for a particular user or data set. A generative approach creates a model based solely



on examples of normal data from training and then evaluates each test case to see how well it fits the model. In contrast, a discriminative approach attempts to distinguish between normal and abnormal data classes. Both kinds of data are used to train systems in discriminative approaches.

Clustering-Based Anomaly Detection

Clustering-based anomaly detection remains popular in unsupervised learning. It rests upon the assumption that similar data points tend to cluster together in groups, as determined by their proximity to local centroids.

K-means, a commonly-used clustering algorithm, creates 'k' similar clusters of data points. Users can then set systems to mark data instances that fall outside of these groups as data anomalies. As an unsupervised technique, clustering does not require any data labeling.

Clustering algorithms might be deployed to capture an anomalous class of data. The algorithm has already created many data clusters on the training set in order to calculate the threshold for an anomalous event. It can then use this rule to create new clusters, presumably capturing new anomalous data.

However, clustering does not always work for time series data. This is because the data depicts evolution over time, yet the technique produces a fixed set of clusters.

Density-Based Anomaly Detection

Density-based anomaly detection techniques demand labeled data. These anomaly detection methods rest upon the assumption that normal data points tend to occur in a dense neighborhood, while anomalies pop up far away and sparsely.

There are two types of algorithms for this type of data anomaly evaluation:

K-nearest neighbor (k-NN) is a basic, non-parametric, supervised machine learning technique that can be used to either regress or classify data based on distance metrics such as Euclidean, Hamming, Manhattan, or Minkowski distance.

Local outlier factor (LOF), also called the relative density of data, is based on reachability distance.

Support Vector Machine-Based Anomaly Detection

A support vector machine (SVM) is typically used in supervised settings, but SVM extensions can also be used to identify anomalies for some unlabeled data. A SVM is a neural network that is well-suited for classifying linearly



separable binary patterns—obviously the better the separation is, the clearer the results.

Such anomaly detection algorithms may learn a softer boundary depending on the goals to cluster the data instances and identify the abnormalities properly. Depending on the situation, an anomaly detector like this might output numeric scalar values for various uses.

Anomaly Detection Machine Learning

As mentioned briefly above, supervised, semi-supervised, or unsupervised machine learning techniques provide the foundation for anomaly detection algorithms.

Supervised Machine Learning for Anomaly Detection

Supervised machine learning builds a predictive model using a labeled training set with normal and anomalous samples. The most common supervised methods include Bayesian networks, k-nearest neighbors, decision trees, supervised neural networks, and SVMs.

The advantage of supervised models is that they may offer a higher rate of detection than unsupervised techniques. This is because they can return a confidence score with model output, incorporate both data and prior knowledge, and encode interdependencies between variables.

Unsupervised Machine Learning for Anomaly Detection

Unsupervised methods do not demand manual labeling of training data. Instead, they operate based on the presumption that only a small, statistically different percentage of network traffic is malicious and abnormal. These techniques thus assume collections of frequent, similar instances are normal and flag infrequent data groups as malicious.

The most popular unsupervised anomaly detection algorithms include Autoencoders, K-means, GMMs, hypothesis tests-based analysis, and PCAs.

Semi-Supervised Anomaly Detection

The term semi-supervised anomaly detection may have different meanings. Semi-supervised anomaly detection may refer to an approach to creating a model for normal data based on a data set that contains both normal and anomalous data, but is unlabelled. This train-as-you-go method might be called semi-supervised.

A semi-supervised anomaly detection algorithm might also work with a data set that is partially flagged. It will then build a classification algorithm on just



that flagged subset of data, and use that model to predict the status of the remaining data.

Anomaly Detection Use Cases

Some of the primary anomaly detection use cases include anomaly based intrusion detection, fraud detection, data loss prevention (DLP), anomaly based malware detection, medical anomaly detection, anomaly detection on social platforms, log anomaly detection, internet of things (IoT) big data system anomaly detection, industrial/monitoring anomalies, and anomalies in video surveillance.

An anomaly based intrusion detection system (IDS) is any system designed to identify and prevent malicious activity in a computer network. A single computer may have its own IDS, called a Host Intrusion Detection System (HIDS), and such a system can also be scaled up to cover large networks. At that scale it is called Network Intrusion Detection (NIDS).

This is also sometimes called network behavior anomaly detection, and this is the kind of ongoing monitoring network behavior anomaly detection tools are designed to provide. Most IDS depend on signature-based or anomaly-based detection methods, but since signature-based IDS are ill-equipped to detect unique attacks, anomaly-based detection techniques remain more popular.

Fraud in banking (credit card transactions, tax return claims, etc.), insurance claims (automobile, health, etc.), telecommunications, and other areas is a significant issue for both private business and governments. Fraud detection demands adaptation, detection, and prevention, all with data in real-time.

Data loss prevention (DLP) is similar to prevention of fraud, but focuses exclusively on loss of sensitive information at an early stage. In practice, this means logging and analyzing accesses to file servers, databases, and other sources of information in near-real-time to detect uncommon access patterns.

Malware detection is another important area, typically divided into feature extraction and clustering/classification stages. Sheer scale of data is a tremendous challenge here, along with the adaptive nature of the malicious behavior.

Detecting anomalies in medical images and records enables experts to diagnose and treat patients more effectively. Massive amounts of imbalanced data means reduced ability to detect and interpret patterns without these techniques. This is an area that is ideal for artificial intelligence given the tremendous amount of data processing involved.



Detecting anomalies in a social network enables administrators to identify fake users, online fraudsters, predators, rumor-mongers, and spammers that can have serious business and social impact.

Log anomaly detection enables businesses to determine why systems fail by reconstructing faults from patterns and past experiences.

Monitoring data generated in the field of the Internet of things (IoT) ensures that data generated by IT infrastructure components, radio-frequency identification (RFID) tags, weather stations, and other sensors are accurate and identifies faulty and fraudulent behavior before disaster strikes. The same is true of monitoring industrial systems such as high-temperature energy systems, power plants, wind turbines, and storage devices that are exposed to massive daily stress.

How Does Avi Networks Help With Anomaly Detection?

Avi's [distributed load balancers](#) collect telemetry in the path of application traffic in real-time. This enables the platform to leverage its position, monitoring traffic spikes, detecting any anomalies—and, ultimately, the [distributed denial of service \(DDoS\) attacks](#) that often underlie these spikes in anomalous traffic.

As an intrinsic feature of the Application Delivery Controller (ADC), Avi's approach is inline, and does not demand any additional anomaly detection software integration. Avi's intelligent approach allows for better decisions on detection of outliers, elastic capacity planning, prediction of load, and more.

The system is actionable, and with anomaly detection happening in real-time and mitigation actions built into operations, it's also fast. Admins do not need to manually start a new server, block offending clients, or increase capacity based on predictions about resource utilization. The result is real-time, automatic anomaly detection.

For more information on Avi's anomaly detection platform see: "[Supercharging Anomaly Detection with the holt-winters Algorithm](#)".

For more on the actual implementation of load balancing, security applications and web application firewalls check out our [Application Delivery How-To Videos](#).



Featured Resources



Why Avi

What We Do
Platform Overview
Platform Architecture

Education

Start Here
Workshops
Operations Videos
How-To Videos
Load Balancing
Web Application Firewall
Microservices & Containers
Application Delivery Controller
Ingress Controller

Solutions

Modern Load Balancing
Upgrade from F5
Web Application and API Protection (WAAP)

VMware Ecosystem

VMware NSX
VMware Horizon
VMware vSphere

Public / Private Cloud

Microsoft Azure
Amazon Web Services
Google Cloud Platform
OpenStack

Containers

Kubernetes
VMware Tanzu

Products

Avi
Overview
Software Load Balancer
Web Application and API Security
Container Ingress

Customers

Application Services Case Studies

Resources

Resource Center

Content Library
Webinars
LBTV
Blog

Technical Help

Knowledge Base
Professional Services
Support
Glossary

Company

Contact Us
Privacy Policy
California Privacy Rights



[Privacy Policy](#) | [Terms of Use](#) | [California Privacy Rights](#)

© 2023 Avi Networks. All rights reserved.

