[Anomaly Detection is the New Black](http://data-informed.com/anomaly-detection-new-black/" \o "Permanent Link to Anomaly Detection is the New Black)

In a smooth-running business, something that stands out from normal usually is not good. But even if it’s a happy accident, you still need to look at it.

Sounds simple, but with huge amounts of data this can be challenging, and the volume of incoming data is growing fast. More and more things are being attached to the Internet, and these things are often continuously making measurements to determine how they are working and what is around them. How can you look at it all and figure out what is anomalous? You definitely can’t do it manually. Even with an automated system, how do you tell a program to recognize what’s unusual when you don’t know what the next anomaly will look like? Saying, “I’ll know it when I see it” isn’t really a valid computer program.

The answer is to build an automated, self-adaptive anomaly detection system. Fortunately, doing so is much easier than it used to be, and the anomaly detector you can build today will work even better than those before. These improvements are due in part to new practical machine learning algorithms that are simpler to use, and to better understanding of how to address problems that require anomaly detection. Ironically, it’s also easier to build an effective anomaly detector partly because of the very thing causing the challenge: huge amounts of data. For anomaly detection, having more data means it’s easier to detect an unusual event against the background of normal events.

**Who needs anomaly detection?**

Good anomaly detection provides benefits in many areas, including:

* + **Security**. Anomaly detection can let you stop attacks before they succeed. This works in part because you can often detect patterns of events that indicate when somebody is probing your defenses. This results in less loss from external attacks due to the ability to deploy more effective countermeasures when needed and to detect successful attacks more quickly.

* + **Quality assurance and predictive maintenance**. Anomaly detection helps you understand when sensor and log data indicates that a machine or process is headed into new territory, which is often a signal of impending failure. Less down time results in fewer production losses and higher product quality.

* **Changes in traffic**. Catching problems as soon as possible allows you to minimize down time. This allows you to better respond to changes in loads or user behavior.

And, perhaps most importantly, good anomaly detection allows you to find problems before your CEO does.

Let’s look at how to build anomaly detectors. Along the way, we’ll examine how you have to view problems like this in order to succeed.

**Discover, Don’t Define**

When people think of anomaly detection, they often think in terms of setting particular thresholds or defining rules to characterize problematic behavior. While these methods can definitely flag some of the problems you may encounter, it is generally better to discover patterns rather than to define them. Discovery means that you let the data speak to you to determine what “normal” is. When done well, such discovery allows an anomaly detection system to adapt itself to a changing world while staying effective.

**Related Stories**

The Internet of Things: More Connectivity Can Mean More Vulnerability.  
[**Read the story »**](http://data-informed.com/internet-of-things-connectivity-can-mean-vulnerability/)

4 Ways to Analyze IT Systems Data to Manage Application Performance.  
[**Read the story »**](http://data-informed.com/4-ways-analyze-systems-data-manage-application-performance/)

Cyber Security Skill Shortage: A Case for Machine Learning.  
[**Read the story »**](http://data-informed.com/cyber-security-skill-shortage-case-machine-learning/)

Using Evolutionary Biology to Inform Machine Learning Algorithms.  
[**Read the story »**](http://data-informed.com/using-evolutionary-biology-inform-machine-learning-algorithms/)

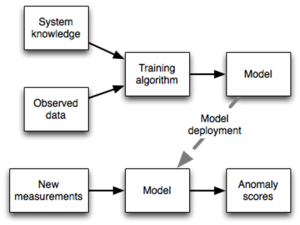
The key idea here is that you should discover the normal patterns in your system so that you can recognize anomalies. Discovery allows you to identify anomalies even though you don’t know what they might look like. Once you have enough examples of certain kinds of anomalies, you can add conventional predictive analytics to the mix to handle what you now know about and leave the anomaly detection system to flag what you still don’t know about.

The current best method for implementing adaptive anomaly detection comes from an area of machine learning known as [one-class classification](https://en.wikipedia.org/wiki/One-class_classification). The basic idea behind statistical anomaly detection is that you encode the patterns of what is normal as a probabilistic model. The benefit here is two-fold. First, probabilistic models come with a built-in measure of anomaly that is as good as or better than any other possible measure in terms of determining what is normal and what is anomalous. Second, the probabilistic models we are talking about here come with a way of learning from observed data that is called a training algorithm. Human insight in the form of prior knowledge is also used to inform and constrain the training algorithm.

**Why is Probability Important?**

The key property that makes a probabilistic model good for anomaly detection is the constraint that the probability of all possible things has to sum to one. This means that if you make the probability of something higher, something else has to have lower probability. Training algorithms for probabilistic models can use this constraint by concentrating probability around what is normal, and thus making the modeled probability of anomalies lower.

Operationally, to build a probabilistic model, you feed observed data and prior system knowledge into the training algorithm. The output is a model. This model is then deployed and fed new measurements to gauge how anomalous they might be. **Figure 1** shows how this works.

[](http://data-informed.com/wp-content/uploads/2015/02/Figure-1-new.png)

**Figure 1**. Observed data and prior knowledge are combined by a training algorithm to produce a model. This model is then used to grade new observations according to how well these new observations fit the model’s discovered definition of normal. The output of the model is a probability that can be used as an anomaly score. Click to enlarge.

Modern training algorithms allow expert domain knowledge to be captured in a model, just like the patterns found in the observed data. With some algorithms, training and deployment can even be done as a continuous process, without any intervention required. If enough similar systems are being monitored, even the system knowledge can be learned so that general patterns can be found.

**What About in Practice?**

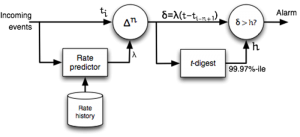
As everyone knows, there is no difference between theory and practice, at least theoretically speaking. In practice, of course, there is always a difference. In the case of statistical anomaly detection, that difference arises from a number of factors, including the amount of data available; the sophistication, accuracy, and accessibility of the prior knowledge; and the sophistication of the models that can be learned.

However, recent trends are improving all of these factors. Massive amounts of additional data are becoming available, prior knowledge is becoming easier to derive as more and more similar systems are being monitored, and the sophistication of models that can feasibly be built is increasing dramatically. The new state of the world is that, statistically, anomaly detection can be applied in a large and growing number of systems.

**A Worked Example**

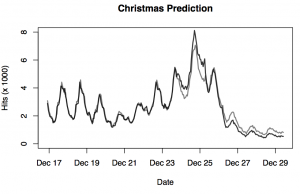
The methods for building good probabilistic models vary a bit according to what kind of system you are modeling. In our recent book, “[A New Look At Anomaly Detection](http://info.mapr.com/resources_ebook_anewlook_anomalydetection.html),” Ellen Friedman and I describe practical ways to build anomaly detectors for continuous signals, discrete events, and symbolic sequences. For each of these kinds of data, we combined what we know about the system with simple statistical techniques that learned about the kinds of patterns that we knew the system was likely to produce. For instance, with EKG signals, we know that the signal is relatively smooth and highly repetitive. That was enough for us to build an accurate model of the short-term dynamics of heartbeats, but it also could have been a model of a chemical factory, a municipal water system, or a steam turbine.

With the web traffic model, our prior knowledge was that the average rate that incoming events arrive varies slowly from minute to minute, and that rate patterns repeat on a daily and weekly basis. **Figure 2** shows the system that we built to detect web server outages, but a very similar system can be applied to purchase processing systems, network communications, and other systems that measure event timing.

[](http://data-informed.com/wp-content/uploads/2015/02/Figure-2-new.png)

**Figure 2**. Arrival times of incoming events are the input to this anomaly detector. Differences between arrival times are multiplied by a predicted rate to get a normalized anomaly score. This score is compared with a threshold computed using a t-digest. The threshold is chosen to select how often we can have the alarm go off. Click to enlarge.

Such an event arrival model depends critically on the ability to forecast expected traffic over a short time period. Often, this comes down to predicting how many events we will see in the upcoming hour. **Figure 3** shows how a well a surprisingly simple model can do in this prediction task. Here, the rate predictor was trained on hourly visit counts to the Wikipedia page for Christmas during the last few weeks of November and the first two weeks of December, 2008. Then the rate predictor was let loose on the last two weeks of December and the first week of January. In spite of the wild variations in the traffic through Christmas, this model was able to predict hour-ahead traffic very accurately until the last few days of the month, when relative error degraded to about 20 percent.

[](http://data-informed.com/wp-content/uploads/2015/02/Figure-3.png)

**Figure 3**. A rate model for a web traffic anomaly detector can do a very good job at predicting traffic during late December, even though it was trained only on the last weeks of November and the first weeks of December. Click to enlarge.

A common element of all of the cases we examined in our book was that the model we built gave us an anomaly score in terms of the log of the probability as predicted by the model. This anomaly score is inherently calibrated if the model closely matches reality, but we often use a simplified model that may not be quite as accurate. This means that it often helps to use a technique such as the [*t-digest*](https://github.com/tdunning/t-digest/blob/master/docs/t-digest-paper/histo.pdf?raw=true) to calibrate the actual score according to our experience.

Statistical anomaly detection is quickly becoming a required capability in the new world of the Internet of Things. Happily, it is also very tractable for many important special cases, including continuous signals, discrete event timing, and user log files. Applications are very broad, from rapid diagnosis of unknown fault types to security log analysis.

The key changes in how we look at data include a major change from defining patterns to discovering patterns. The key benefits are an ability to find things that we didn’t even know to look for and to rapidly and transparently diagnose faults.