Introduction to Anomaly Detection

Thamindu Dilshan Jayawickrama Aug 18, 2020 · 5 min read ★

Photo by Harley-Davidson on Unsplash

Have you ever thought about what happens when someone's social media account, bank account or any other account/ profile get hacked? How do some systems automatically detect such activities and notify them to relevant authorities or suspend the account immediately? That is mainly done through the process called **Anomaly Detection** (AKA **Outlier Detection**). In fact, the above particular example is called Fraud Detection and it is a popular application of anomaly detection in lots of domains. So let's dive into anomaly detection terminology.

Anomaly detection or outlier detection is the process of identifying rare items, observations, patterns, outliers, or anomalies which will significantly differ from the normal items or the patterns. Anomalies are sometimes referred to as outliers, novelties,

noise, deviations or exceptions. According to some literature, three categories of anomaly detection techniques exist. They are **Supervised Anomaly Detection**, **Unsupervised Anomaly Detection,** and **Semi-supervised Anomaly Detection**.

Supervised vs Unsupervised Anomaly Detection

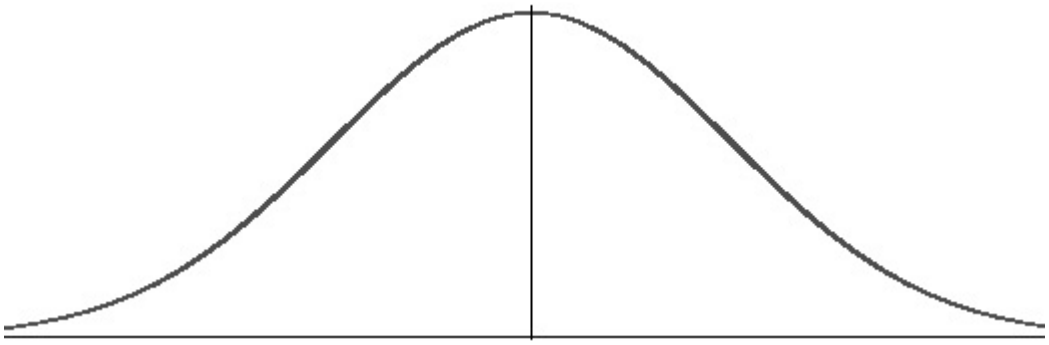
The most common version of anomaly detection is using the unsupervised approach. In there, we train a machine-learning model to fit to the normal behavior using an unlabeled dataset. In that process, we make an **important assumption that the majority of the data in the training set are normal examples**. However, there can be some anomalous data points among them (a small proportion). Then any data point which differs significantly from the normal behavior will be flagged as an anomaly. In supervised anomaly detection, a classifier will be trained using a dataset that has been labeled as 'normal' and 'abnormal'. When a new data point comes, it will be a typical classification application. There are pros and cons in both of these methods. The supervised anomaly detection process requires a large number of positive and negative examples. Obtaining such a dataset will be very difficult since anomalous examples are rare. Even though you obtain such a dataset, you would only be able to model the abnormal patterns in the gathered dataset. However, there are many different types of anomalies in any domain and also future anomalies may look nothing like the examples seen so far. It will be very hard for any algorithm to learn from anomalous examples; what the anomalies look like. That is why the unsupervised approach is popular. **Capturing the normal behavior is much easier than capturing the many different types of abnormalities**.

Terminology Behind Anomaly Detection

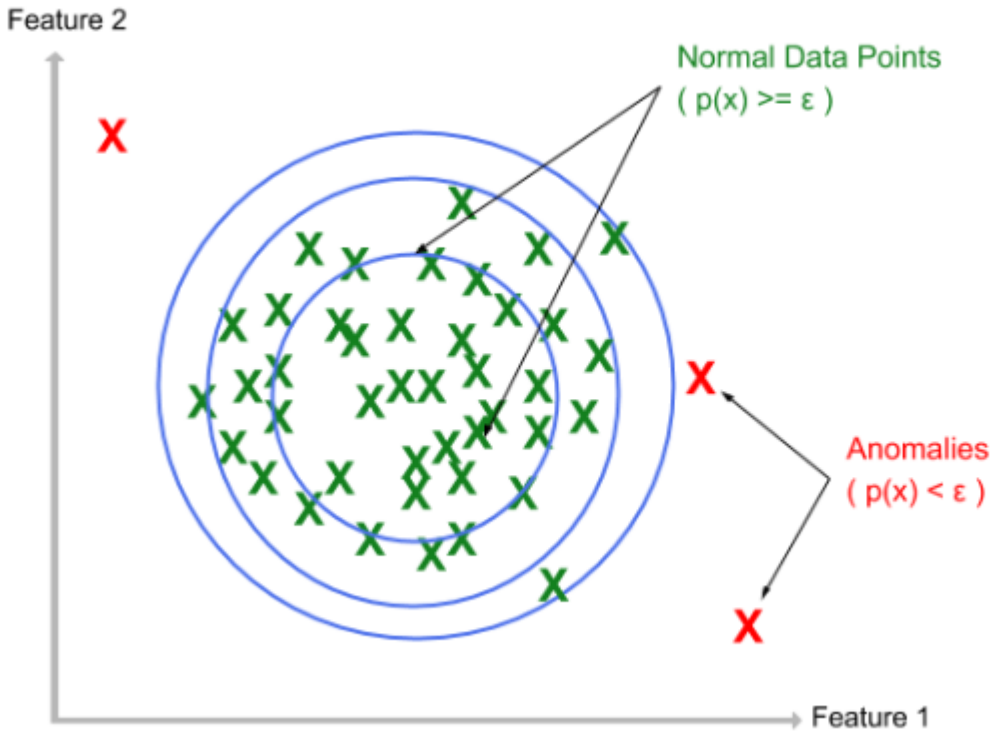
Now let's dive into the idea behind unsupervised anomaly detection. The process will be referred to as anomaly detection rather than unsupervised anomaly detection throughout the article. Before start discussing the anomaly detection algorithm, there is something called the **Gaussian (Normal) Distribution**, which the entire algorithm has been built on. In statistics, a Gaussian or a Normal Distribution is a form of continuous probability distribution for a real-valued random variable. The probability function is calculated using the below formula. μ is the mean and σ^2 is the variance.

$$\mathcal{N}(x\ ;\ \mu, \sigma) = \frac{}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{}{2}(x-\mu)^2/\sigma^2\right]$$

The probability distribution is symmetric about its mean and non-zero over the entire real line. The normal distribution is sometimes called the bell curve because the density graph looks like a bell.



The basic idea of anomaly detection is to find a probability function to capture the normal behavior and discover a probability threshold such that, the data points far away from that threshold are considered anomalies. Considering the probability function as p(x) and threshold as ε, this can be depicted as below. The data points are represented by the crosses.



In the basic anomaly detection algorithm, we assume that each feature is distributed according to its own Gaussian Distribution with some set of means and variances. So using the training dataset, we fit a set of parameters $\mu 1, \mu 2, ..., \mu n$ and $\sigma 1^2, \sigma 2^2, ..., \sigma n^2$ with respect to features x1, x2, ..., xn. Then the probability function p(x) is calculated as the probability of x1 times the probability of x2 times the probability of x3 and so on up to the probability of xn. That is the machine learning model for anomaly detection. When a new data point comes in, we simply calculate the probability p(x) and flag it as an anomaly if the probability is less than ε. Finding the correct value for ε is an optimization objective and I'm not going to explain it in this article.

Anomaly Detection Algorithm (using Gaussian Distribution)

1. Choose a set of features x_i and fit the parameters $\mu_1, \mu_2, ..., \mu_n$ and $\sigma_1^2, \sigma_2^2, ..., \sigma_n^2$ assuming gaussian distributions.

$\mu_j = \frac{1}{m} \sum_{i=1}^{m} x_j^{(i)}$ and $\sigma_j^2 = \frac{1}{m} \sum_{i=1}^{m} (x_j^{(i)} - \mu_j)^2$ where m is the number of training examples

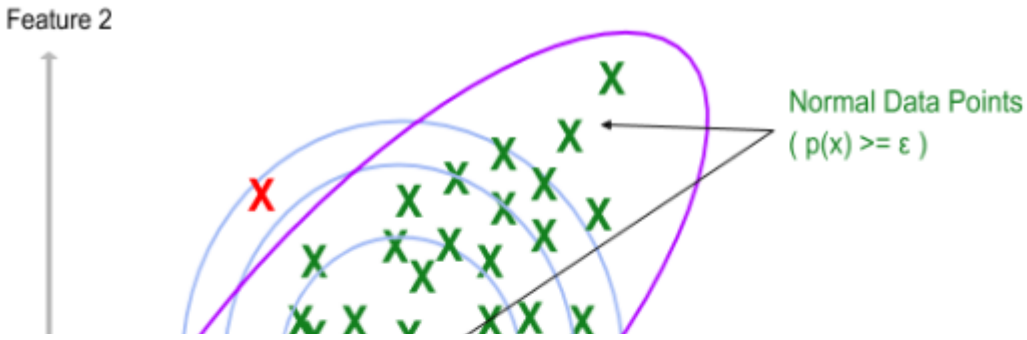
2. Given new example x, compute;

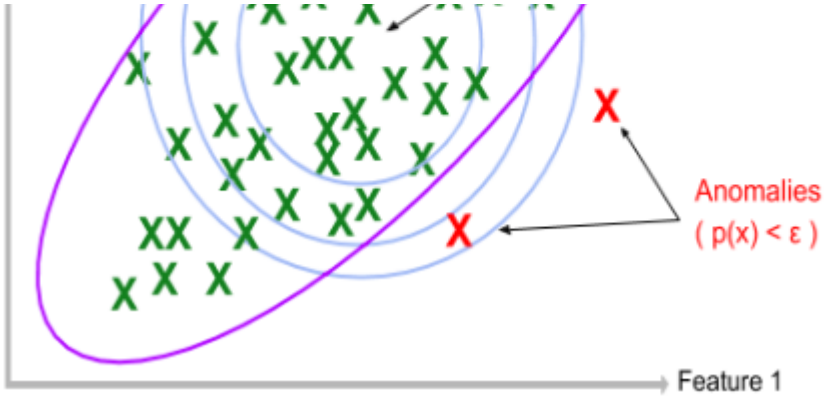
$p(x) = \prod_{j=1}^{n} p(x_j; \mu_j, \sigma_j^2)$ where n is the number of features

3. Flag as anomaly if p(x) < ε

More on Anomaly Detection

An important fact is to choose a set of features that are indicative of anomalies. That is to choose a set of features in which the features might take on an unusually large or small value in the presence of an anomaly. On the other hand, there might be some sort of drawback since we have to manually create the features. Also, this algorithm could only detect a few anomaly scenarios. For example, how can we detect anomalies when the data is distributed as below.





As you can see, the earlier model using the Gaussian Distribution flag some of the normal data points as anomalies and some anomalous examples as normal data points (blue boundaries in the above figure). In fact, we need an advanced shaped boundary for this sort of data distribution (maybe boundary in purple color). This particular problem is addressed using **Multivariate Gaussian Distribution**. In there, instead of modeling each feature in its own Gaussian distribution and multiplying the probabilities, all the features are modeled into one common distribution called the Multivariate Gaussian Distribution.

Popular Techniques

Today, when implementing an anomaly detection algorithm you don't have to worry about the above details. The popular libraries like scikit-learn (for python) provides much easier ways to implement anomaly detection algorithms. Here are some of the popular techniques used for anomaly detection.

- Density-based techniques (KNN, Local Outlier Factor, Isolation Forest, etc)
- Cluster analysis based techniques (KMeans, DBSCAN, etc)
- Bayesian Networks
- Neural networks, autoencoders, LSTM networks
- Support vector machines
- Hidden Markov models
- Fuzzy logic based outlier detection

Some Applications

- Fraud detection (Ex: network, manufacturing, credit card fraud, etc)
- Intrusion detection
- Fault detection
- System health monitoring
- Detect ecosystem disturbances
- Anomaly detection for product quality

Conclusion

Anomaly detection is a technique of finding rare items or data points that will differ significantly from the rest of the data. Even though the terminology behind anomaly detection uses the probability theory and some statistics, there are many techniques to easily implement an anomaly detection algorithm.
