Understanding Bigeye Autothresholds

Bigeye Data Science Team

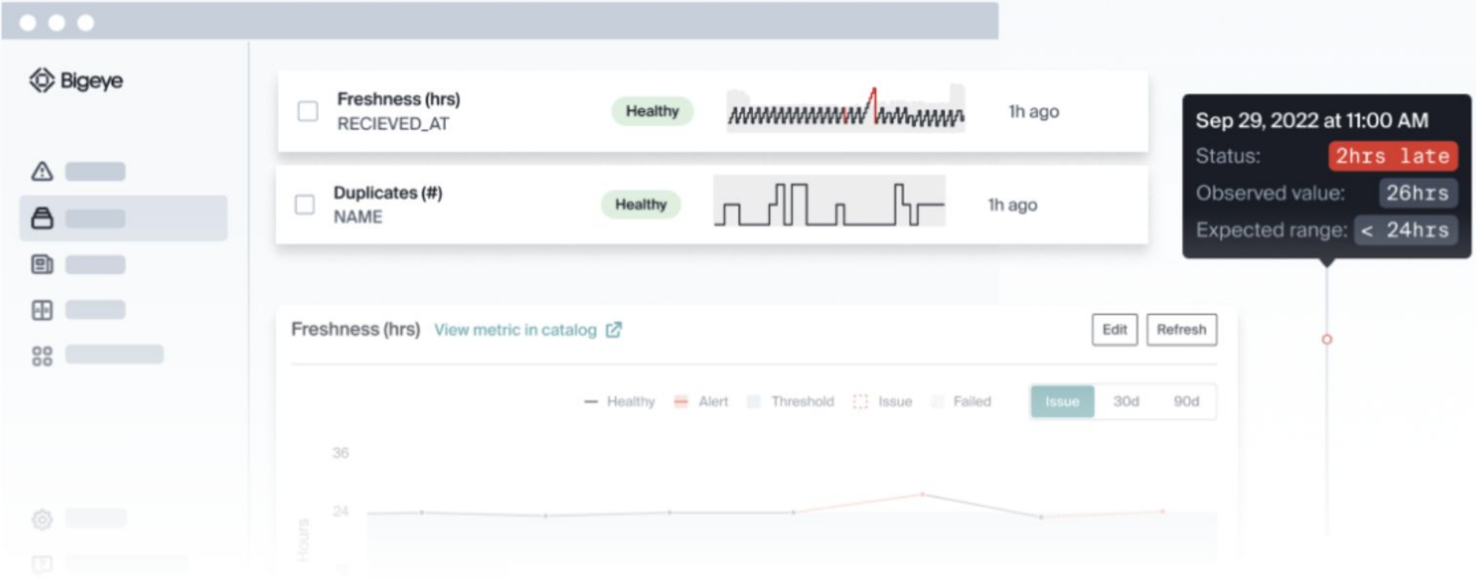
Contents

Introduction

Enterprise data engineering teams are responsible for maintaining and ensuring the health of hundreds of datasets with complex data pipelines.

In practice, this often means manually defining a large amount of tests that track attributes like acceptable latency, expected volume, field-level distribution, format checks, and others. This process is time consuming, error-prone, and not scalable. In some cases, data engineers lack the business context or domain expertise to know what is expected or where to start. Even when tests are defined correctly, data tends to “drift” over time, requiring ongoing tweaks and updates—creating even more overhead.

In this paper, we’ll explore how Bigeye Autothresholds help solve these challenges and how you can use them to take the manual effort out of monitoring data pipelines at scale.



What Are Autothresholds?

Data observability platforms like Bigeye use thresholds to alert data teams to anomalies that need attention. While Bigeye users can choose to apply their own thresholds using common methods such as standard deviation or constants, Autothresholds use a proprietary machine-learning engine to automatically generate thresholds for every data attribute you track—giving you meaningful and actionable alerts with zero manual effort.

Autothresholds automate the setup and management of Bigeye's observability system. And, because they start learning from your data automatically, they're especially helpful at providing insights to infrastructure teams without requiring them to have a deep understanding of the data they're monitoring.

Autothresholds also adapt to seasonalities and trends and learn from user feedback, reducing false positives and preventing alert fatigue.

How Autothresholds Work

To calculate Autothresholds, Bigeye analyzes recent metric history, evaluates the performance of several forecasting techniques, and then applies the most accurate to predict expected values. This forecast is adjusted to user settings like **sensitivity** and **bounds**, as well as technical considerations like uncertainty and metric patterns.

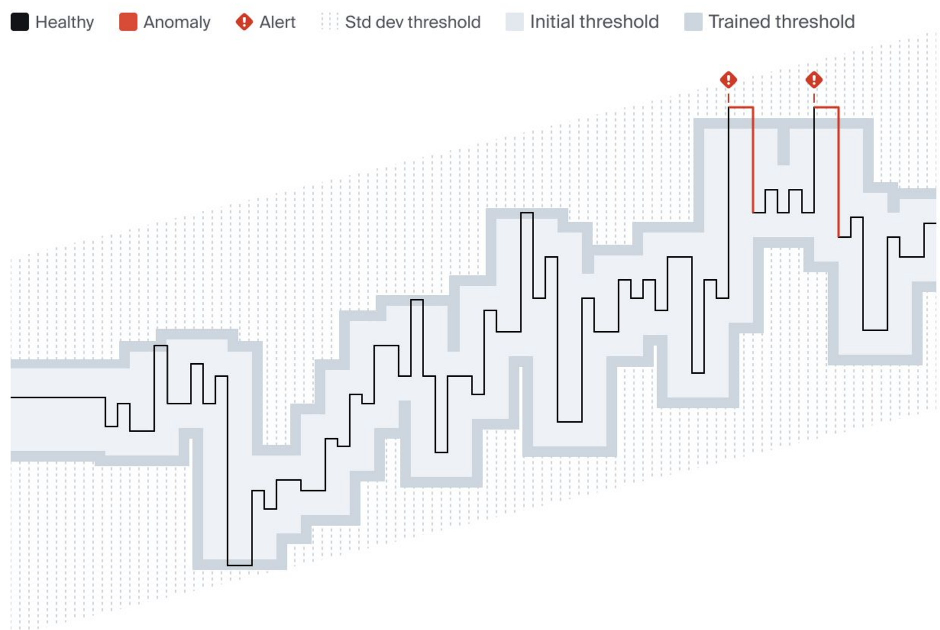
Autothresholds are periodically calculated using the following process:

1. Analyze the underlying structure of the series using preliminary statistical tests
2. Perform a blind-prediction test with an array of techniques and select the most accurate
3. Analyze past data to develop a model for the uncertainty of future values
4. Integrate forecasts, information about the underlying structure, and uncertainty to calculate boundaries for the limits of “expected” behavior
5. Adjust expected range for user settings such as sensitivity & bounds

Understanding Autothresholds

Calculating Autothresholds

Bigeye can automatically backfill metric history if there is a **row creation time** set on the table. This enables Bigeye to generate Autothresholds immediately, without waiting for training data. If there is no suitable row creation time for the table or if the backfill is unavailable, Autothresholds will be generated once the metric has a sufficient amount of training data.

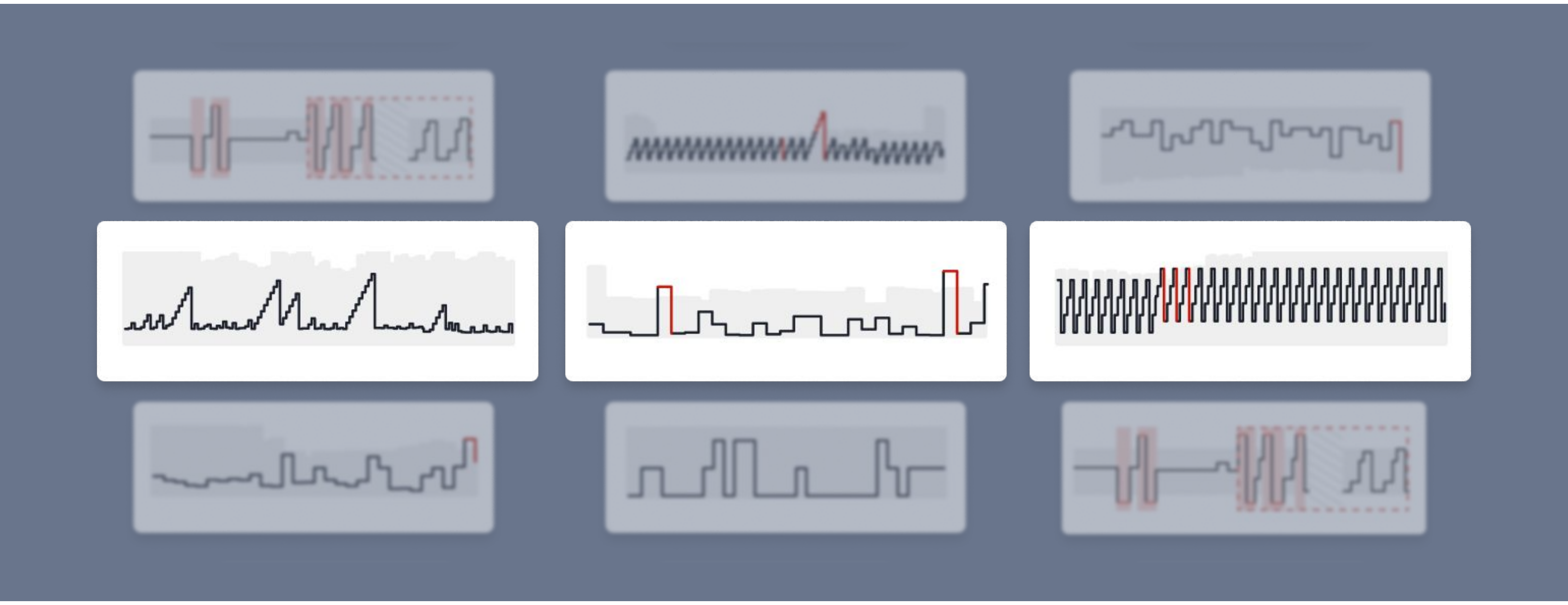


Autothreshold calculations are based off of recent metric history. The exact window can be configured in Advanced settings > Autothresholds > Autothreshold training history. The default training window is 21 days, meaning all observed metric values in the past 21 days are used in model training. This is intended to give the machine learning system time to detect and exploit a range of seasonalities, trends, and business cycles.

Understanding Autothresholds

Accounting for Seasonality

Autothresholds detect and fit against seasonal patterns during preliminary analysis and model selection, integrating open-source and proprietary techniques, and drawing from the domains of traditional statistics and machine learning.



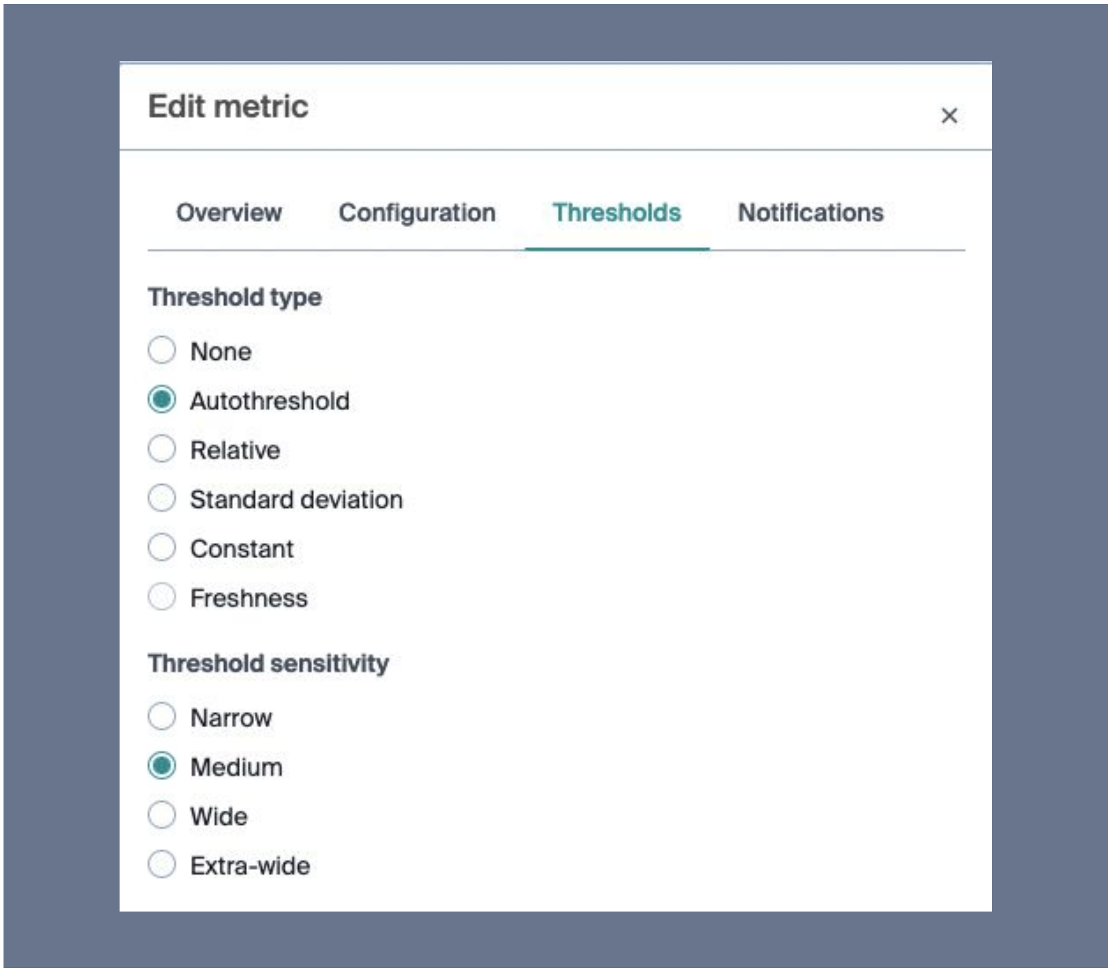
Classical time series forecasting generally requires three or more cycles in order to make strong inferences about seasonal patterns (and avoid false inferences) and Autothresholds abide by this best practice. For example, if you want to model the difference between Friday and Saturday web traffic, you should have at least 3 weeks worth of data in the training history.

By default, Autothresholds include three weeks of data in training history. Users can adjust this for data that follows longer or shorter seasonal patterns.

Understanding Autothresholds

Adjusting Sensitivity

Users can adjust Autothresholds by selecting a sensitivity from narrow to extra wide—the default is medium. While Autothresholds are designed to set threshold sensitivity automatically, this adjustment enables a user to override that tolerance based on business logic or other priorities. Adjustments for sensitivity are applied after model training and selection.



Bigeye's anomaly detection engine is designed with a 5-stage processing pipeline that captures key behavioral patterns in your data and learns over time

Understanding Autothresholds

Learning Over Time

Tip: Autothresholds pull in new data and re-tune models every 24 hours.

Users can configure this cadence in Advanced settings > Autothresholds > Autothreshold training frequency.

Bigeye periodically adds new observations to the data used to calculate thresholds. Likewise, Autothreshold models are periodically re-trained using this new information as it comes in. This ensures we can adapt to underlying changes and maximize accuracy.

Users can provide feedback on Autothreshold predictions when resolving issues in Bigeye. These annotations affect Autothreshold sensitivity and training data so that Autothresholds can better match business expectations over time.

A user can mark an issue as a false positive (i.e., Bigeye alerted, but the user thinks it should be considered normal) by closing the issue as a "**bad alert**". For the purposes of future predictions, Autothresholds interprets "bad alert" points as normal, valid observations.

If, on the other hand, a user would like to mark an issue as a true positive (i.e., Bigeye alerted, and the user agrees that it is an anomaly) they can close the issue as a "**good alert**".

For good alerts, the user will also have the ability to classify this anomaly as a new normal (by selecting "adapt thresholds") or to indicate that they expect data to go back to normal (by selecting "do not adapt thresholds"). In the former case, past data are de-weighted; in the latter, past data are retained.

Conclusion

Autothresholds generate meaningful alerts that help you find and fix data issues before they impact your business.

Bigeye Autothresholds were built for enterprise data environments that need to know about critical issues without being overwhelmed by noise.

Unlike other alerting systems, Bigeye:

- Creates automated alerts that adapt over time
- Eliminates thousands of manual and fragile alert rules
- Starts working on day one with no waiting period
- Provides easily adjustable tolerance levels
- Incorporates user input into its ML process

Ready to learn more?

Visit bigeye.com or speak to your Bigeye representative.

The data observability platform built
by data people, for data people.

bigeye.com

Bigeye