Anomaly Detection in Application Performance Monitoring Data

Thomas J. Veasey and Stephen J. Dodson

Abstract—**Performance issues and outages in IT systems have significant impact on business. Traditional methods for identifying these issues based on rules and simple statistics have become ineffective due to the complexity of the underlying systems, the volume and variety of performance metrics collected and the desire to correlate unusual application logging to help diagnosis. This paper examines the problem of providing accurate ranking of disjoint time periods in raw IT system monitoring data by their anomalousness. Given this ranking a decision method can be used to identify certain periods as anomalous with the aim of reducing the various performance metrics and application log messages to a manageable number of timely and actionable reports about unusual system behaviour. In order to be actionable, any such report should aim to provide the minimum context necessary to understand the behaviour it describes.**

In this paper, we argue that this problem is well suited to analysis with a statistical model of the system state and further that Bayesian methods are particularly well suited to the formulation of this model. To do this we analyse performance data gathered for a real internet banking system. These data highlight some of the challenges for accurately modelling the system state; in brief, very high dimensionality, high overall data rates, seasonality and variability in the data rates, seasonality in the data values, transaction data, mixed data types (continuous data, integer data, lattice data), bounded data, lags between the onset of anomalous behaviour in different performance metrics and non-Gaussian distributions. In order to be successful, subject to the criteria defined above, any approach must be flexible enough to handle all these features of the data.

Finally, we present the results of applying robust methods to analyse these data, which were effectively used to pre-empt and diagnose system issues.

Index Terms—**Anomaly detection, APM.**

I. Introduction

Businesses today have become dependent on increasingly large and complex IT systems. Understanding and managing these systems relies on instrumenting their behaviour and understanding the resulting monitoring data. Traditionally, static thresholds and simple statistical methods, such as thresholds on number of standard deviations of an observation, have been used to proactively alert IT staff of anomalous system behaviour. With the increasing scale and complexity of these systems and improvements in application performance management (APM) monitoring, the coverage and volume of machine generated data has increased significantly, exposing deficiencies in these traditional methods for accurately identifying anomalies with low false alarm rate [1], [2].

Over recent years, the area of outlier detection has received a lot of research interest, and a large number of different algorithms have been presented. However, many approaches can be classified as variants on a similar theme. In particular, most approaches either use some statistical model of the data set, or look at distances between data points.

The statistical approaches include parametric and non-parametric distribution fitting, and consider various tests for outliers based on the distribution(s) they fit. A good introduction to statistical outlier detection is Hawkins [3]. For more recent work in this area, see Caussinus and Roiz [4] and Liu *et al*. [5]. We would also tentatively include PCA and particularly robust PCA approaches in this category, since these are statistical techniques that can give rise to natural definitions of outliers; although really they are dimension reduction techniques and aren't specifically formulated with outlier detection in mind. For more information on these techniques see, for example, Jackson and Chen [6].

The distance based approaches look at distance between neighbouring points or local density at a point in some metric space. These approaches typically consider functions based on the set of the k nearest neighbours, and classify outliers as (relatively) far from their neighbours or in regions of low density. For approaches of this sort, see Knorr and Ng [7], Breunig *et al*. [8] and Fan *et al*. [9]. It is important to note that all these approaches to outlier detection assume the data can be embedded in a metric space.

The traditional statistical definition of outliers is framed in the language of hypothesis testing. For example, Hawkins' definition is "an outlier is an observation which deviates so much from the other observations as to arouse suspicions that it was generated by a different mechanism". Here, the normal mechanism, which describes most of the observations, would be the null hypothesis. This definition is the basis for the *Z*-test and *t*-test for identifying significant deviations from the mean of the data. In the context of APM, and indeed many other applications, for example, network intrusion, fraud detection and so on, the definition of outlier should also capture something about the observation rarity. For example, consider a collection of time stamped log messages. Often, of primary interest are identifying time periods where an unexpectedly high number of rare log messages occur, where rare would be those messages with the lowest arrival rate. In this case, we are *not* interested in rejecting some null hypothesis concerning the mechanism that generates a particular observation, only that the observation is highly unlikely. Note also that distance based definitions of outliers are not well suited to identifying such periods as outliers. Many of these are local measures, in the sense that an

Manuscript received August 19, 2013; revised November 18, 2013.

T. J. Veasey and S. J. Dodson are with Prelert Ltd, UK (e-mail: tveasey@prelert.com, steve@prelert.com).

observation is compared to its neighbours. For example, in [8] if a point is near some other (rare) observations it will generally not have a high local outlier factor.

If it were possible to make a small set of hypotheses concerning the mechanisms which give rise to the data, and further that some of these mechanisms could be labelled as normal, then either the Neyman-Pearson framework, or Bayesian statistics, give natural classifiers of observations as outliers, or probabilities of observations being outliers. However, for APM these mechanisms might be expected to be isomorphic to the number of states of the system to be modelled, which, even for a moderately complex system, would make this approach intractable.

Where no alternative hypothesis is available, the *p*-value of a test statistic can be used as the basis for rejecting the null hypothesis, or in our case identifying a point as an outlier. As a method for rejecting the null hypothesis *p*-value tests are open to significant criticisms, see Schervish [10]. However, from the point of view of outlier detection the *p*-value does capture something useful about the rarity of an observation. Furthermore, *p*-values provide a natural ranking of all the observations by their unusualness. Given this ranking, a number of approaches are available for classifying observations as outliers: natural breaks, quantiles and so on. Note, however, in its traditional formulation the *p*-value identifies extreme values. These are strictly a subset of what one might consider an outlier, in that they are points that lie outside some convex hull of most of mass of the distribution. (Note also that the test always applies a to single value, so one has to construct some suitable statistic for multivariate data, for example the Mahalanobis distance from the sample mean.) To explain the distinction, if a mixture of two univariate Gaussians, that are well separated, describes some one-dimensional data, we expect some points between the modes of these two distributions to occur with low probability. We would typically want to classify these points as outliers, although they are certainly not extreme values.

In this paper, we argue that a natural extension of the *p*-value of an observation provides a good measure for identifying outliers for APM use cases, and in fact for many other applications. In particular, outliers would correspond to small values of this statistic. Furthermore, this measure is a strict weak ordering of the events to be classified as outliers (or normal by analogy), and so can be used to order all events by their deviance (in the sense of Hawkins' the definition of an outlier) or anomalousness. This approach also has the advantage that it doesn't need to embed all the data in a metric space, so provides a consistent measure of anomalies in metric data and categorical data. It also naturally captures the fact that we'd like rare observations to be treated as anomalous. Given this statistic, the problem of outlier detection reduces to the problem of fitting a distribution to the data set on which to perform outlier detection.

For APM data, we are always interested in fitting this model to a data stream. The data volumes are far too high to consider analysing the entire corpus of data in one go, and, in fact, much data are only kept on disk for a relatively short period. Also, people want to be alerted to changes in their system (as near as possible to) when they occur. As a result of these constraints, we use a Bayesian formulation for all our system models, since they are intrinsically online. They also have several other characteristics, which make them particularly well suited to analysis of an evolving system: it is easy to implement an intuitive aging mechanism based on relaxing the prior distributions to something less informative, and there is a natural formulation for the problems of i) choosing between different models for the data based on their observed likelihoods, see, for example, Bishop [11], and ii) incorporating user feedback.

The area of parametric and non-parametric distribution fitting is large, and we do not attempt to give a survey of the methods we use; rather, the rest of the paper is concerned with a discussion of those characteristics of a canonical APM data set which we have found it is most important to capture in order to get good results in practice. In this context, good means a high proportion of user validated incidents detected with a low rate of false alarms. We look at a data set generated by monitoring an internet banking system over a three day period. This is around 12 GB and comprises around 33500 distinct metric time series. We show that parametric families of statistical distributions provide an extremely compact representation of (portions of) this data set, and, provided the overall approach tests the goodness of fit of these distributions and is able to propose non-parametric models where necessary, the traditional objection of rigidity does not apply. Finally, we note that these data are very high dimensional, as is the majority of APM data: people gather many different performance measures about their system. Anomaly detection in high dimensional data poses some particular problems. We discuss these in more detail in Section III.

II. A Definition of Anomalousness

For all the models we will need to deal with we can safely assume that a distribution function exists. Specifically, our system is defined as some random variable from a probability space $(\Omega, \mathcal{F}, P)$ to some measure space $(X, \mathcal{A})$ and there exists a measurable function $f : X \to \mathbb{R}^+$, where $\mathbb{R}^+$ denotes the non-negative real line with Borel algebra, which recovers the probabilities of the measurable sets of X. We define the generalized *p*-value, or *q*-value, of an observation $x \in X$ as:

$$q(x) = P\left(\left\{y : f(y) \leq f(x)\right\}\right) \tag{1}$$

This is clearly well defined, since the closed interval $[0, f(x)]$ is Borel measurable and so its preimage is $\mathcal{A}$ measurable. Since $q(x)$ is a probability it takes values in the interval $[0,1]$. Any subset of $[0,1]$ has the usual strict total ordering of the reals, and so we can define a strict weak ordering of observations by their anomalousness, i.e. $x >_a y$ if and only if $q(x) < q(y)$. In particular, anomalousness could be defined as some monotonic decreasing function of the q-value, for example $-\log(q(x))$.

We will look at this definition on some examples to understand it better. Suppose our model for the system is a univariate normal distribution with mean μ and variance σ, then for a given observation *x* the anomalousness is defined as:

$$q(x) = P(\{y \le \mu - |x-\mu|\}) + P(\{y \ge \mu + |x-\mu|\}) = 1 - \text{erf}\left(\frac{|x-\mu|}{\sqrt{2\sigma^2}}\right) \quad (2)$$

This is just the standard two-tail p-value for a normal distribution. Note that for a non-symmetric single mode one-dimensional random variable this is somewhat different to the standard p-value, since equal values for the cumulative density function don't occur at equal values for the density function.

For a mixture of two univariate Gaussians, points between the modes with low probability density would have high anomalousness, i.e. they'd have low *q*-values. The sublevel set comprises one, two or three intervals. In the problem case for a *p*-value, discussed in the Section I, the sublevel set is three intervals. Then denoting the level set $\{a,b,c,d\}$, we have that

$$q(x) = P(\{x \le a\}) + P(\{b \le x \le c\}) + P(\{x \ge d\}) = F(a) + F(c) - F(b) + 1 - F(d) \quad (3)$$

Here, $F(x)$ is the cumulative density function

$$F(x) = \frac{w_1}{2}\left(1 + \text{erf}\left(\frac{x-\mu_1}{\sqrt{2\sigma_1^2}}\right)\right) + \frac{w_2}{2}\left(1 + \text{erf}\left(\frac{x-\mu_2}{\sqrt{2\sigma_2^2}}\right)\right) \quad (4)$$

and $w_1 + w_2 = 1$. Finally, for a discrete random variable with mass function $f(x_i) = f_i$, where x_i takes values in some fixed number of nominal categories, then

$$q(x_j) = \sum_{f_i \le f_j} f_i \quad (5)$$

Note that in all these, equivalence classes of the strict weak ordering $>_a$ are given by the sets for which the probability density function, or probability mass function, are equal.

For many specific models there exist closed form solutions or efficient numerical methods to compute the *q*-value, or at least good upper and lower bounds. We note also that for any generative model, then, given *n* independent samples from the model $Y_n = \{y\}$, the q-values can always be estimated from

$$q_n(x) = \frac{\left|\{y \in Y_n : f(y) \le f(x)\}\right|}{n} \quad (6)$$

Here, we've used $|A|$ to denote the cardinality of the set A. We show that $q_n(x) \xrightarrow{a.s.} q(x)$ as $n \to \infty$.

Proof: As before, define our system model to be the random variable Y with probability distribution function $f : X \to \mathbb{R}^+$. Let, $A(x)$ denote the $\mathcal{A}$ measureable set $f^{-1}[z : z \ge 0, z \le f(x)]$, and $I_{A(x)}$ denote the indicator function of $A(x)$. Given a random sample y of Y then, by definition, $I_{A(x)}(y) = 1$ with probability $q(x)$ and 0 otherwise. Therefore, $I_{A(x)}(Y)$, which we understand as $I_{A(x)} \circ Y$, is a Bernoulli random variable with success probability $p = q(x)$. By definition,

$$\frac{\left|\{y \in Y_n : f(y) \le f(x)\}\right|}{n} \sim \frac{1}{n}\sum_{i=1}^{n} I_{A(x)}(Y) \quad (7)$$

Furthermore, $\sum_{i=1}^{n} I_{A(x)}(Y) \sim B(n,p)$, i.e. it is a binomial random variable with number of trials n and probability of success p. Noting that $B(n,p) \xrightarrow{a.s.} N(np, np(1-p))$ as $n \to \infty$ it follows that

$$\frac{1}{n}\sum_{i=1}^{n} I_{A(x)}(Y) \xrightarrow{a.s.} N\left(q(x), \frac{q(x)(1-q(x))}{n}\right) \quad (8)$$

In particular, it is normally distributed with mean $q(x)$ and variance $q(x)(1-q(x))/n$. The variance is maximized when $q(x) = 1/2$, and so $q_n(x) \xrightarrow{a.s.} q(x)$ as $n \to \infty$ for all x and we are done.

In fact, the result we've proved is stronger: it tells us the limiting distribution of the error. It is interesting to note that this distribution only depends on $q(x)$ and n, and not on the probability distribution function at all. Note, also, that the normal approximation to the binomial is good even for relatively small n. Fig. 1 indicates the convergence of the scheme given by (6), and the exact *q*-value as a function of the observation density values $f(x)$, for mixture of three univariate Gaussians. Specifically, this shows the variation in the curve $q_n = q_n(f(x))$ for 100 different random trials for the case $n = 100$, and the red line shows the exact value of the curve, i.e. $\lim_{n\to\infty} q_n(f(x))$. Recall that the result we proved relates to the expected spread in the *Y*-values of the curves, and the spread in their *X*-values is scaled by the inverse gradient.

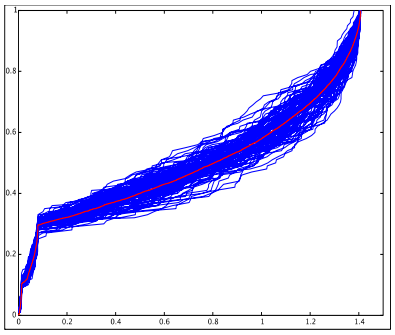

Fig. 1. *Q*-value verses probability density for mixture of three Gaussians.

III. Multivariate Anomaly Detection

This section takes an introductory look at some of the additional complications for outlier detection in a high number of dimensions. Note that for the canonical set presented in this paper the dimensionality is around 33500. For any general outlier detection algorithm this would represent a very significant challenge. However, the requirements are somewhat different for typical APM use cases, and they mean that one can usually get away with considering marginal distributions. We return to this after some general discussion.

One of the key complications for multivariate outlier detection is that outliers are typically only observable in some directions. This happens whenever the data are clustered near some much lower dimensional manifold, and can easily be illustrated with a bivariate Gaussian distribution. Fig. 2 shows the case that the principal component is parallel to the line $y = -x$ and the variance in this direction is much larger than the orthogonal one. In this case, the outlier at the red cross is not visible when the data are projected on to the lines $x = 0$ or $y = 0$, but is clear when the data are projected on to the line $y = x$.

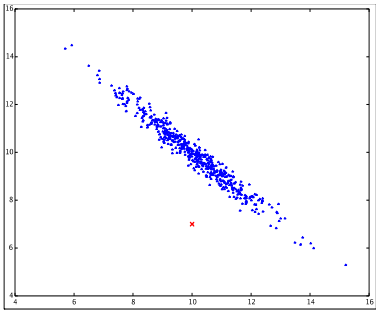

Fig. 2. Outlier from bivariate Gaussian that is not visible in the *X*- and *Y*-marginal distributions.

The observation that this tends to occur when the variance in some low dimensional manifold is much greater than the variance in the data away from that manifold has inspired approaches which look at projecting data points onto the orthogonal complement of the first few principal components. If these account for much of the total variation in the data, then, heuristically, any set of coordinate axes can be used for detecting anomalies on these projected data, or in fact one could look at the distance from mean in the orthogonal complement, see Huang *et al.* [12].

Finally, we note that in very high dimensional spaces distance based approaches suffer because the distance between all points is usually very nearly equal (and, in fact, tends to infinity). Informally, we note that as the dimensionality n increases the proportion of the volume of an *n*-ball near its surface increases. For a high dimensional ball centred on one of the data points, we expect nearly all other points it contains to be very near its surface. Furthermore, we expect it to contain another point when its volume V satisfies $2 = \rho V$, where ρ is the density of points. Even if the density varies significantly from one region to another, the implied change in radius r from this relation is small because $V \propto r^n$. Aggarwal *et al.* studied this effect in detail in [13].

Returning to the APM use cases, we note that system problems are nearly always associated with outliers in one or more of the performance metrics people gather, which will typically include database query times, dropped packets, CPU utilization and so on. Effects of the sort discussed earlier could correspond to a relative phase change in two periodic signals, and are typically not the only symptoms of some significant hardware or software failure. As such, people are generally interested in whether any of the metrics they gather display significant anomalies in their own right. In this context, we can aggregate individual q-values of a collection of observations by considering either their order statistics or some suitable estimate of the joint probability.

IV. Analysis of a Canonical APM Data Set

A. Basic Characteristics

The data set we discuss was gathered by the CA APM product monitoring three serves of an internet banking site, see [14] for more information on these data. Every performance metric is reported at 60s intervals, although some record transactions and are not necessarily available at this granularity. It contains 33,159,939 distinct records and 33,456 distinct time series. The data cover a period of 72 hours and so the total data rate is around 500,000 values per hour. There are 38 categories of metric; these include "responses per interval", "average response time", "errors per interval", "stall counts" and "average result processing". Note that various categories are split out by SQL command, host and so on, which accounts for the total number of distinct time series.

The period we will analyse contains one confirmed significant incident, corresponding to system performance degradation. This will be discussed in Section V.

B. Strong Seasonality

A significant number of the time series in the data set display strong diurnal variation, for example "responses per interval", which dips during the night time for the typical site user's location. In addition, one would also expect different weekday and weekend behaviour, although this is not captured by the data set. Fig. 3 shows diurnal variation in a specific SQL query rate.

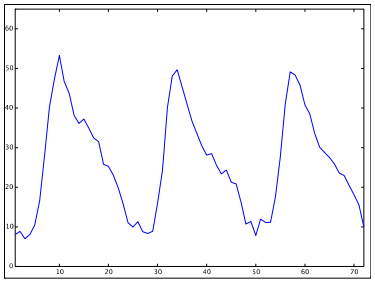

Fig. 3. Average responses per hour for a specific SQL query verses time after the start of the data set in hours.

Any model must be capable of describing these fluctuations in order to look for anomalies in the residuals. A number of approaches are available for this purpose, for example ARMA attempts to find a data transform, in terms of a lag function, which causes the residuals to be independent and identically distributed (usually Gaussian), see Box and Jenkins [15]. Alternatively, a suitably flexible interpolating function, such as single layer radial basis function network, can be used to fit and remove the main trends in the data. We note that for the purpose of outlier detection, the accuracy with which the model fits any slower trends in the data can usually be lower than for prediction.

C. Integer and Lattice Valued Data

Many data correspond to counts, for example "responses per interval", "errors per interval", "stall counts" and so on. Counts are obviously integer valued and they are also non-negative. These characteristics should be considered when fitting a distribution. Particular issues that can arise if one tries to model their values by a continuous distribution are that scale parameters cannot be reliably estimated, as occurs when a long sequence of identical values is received,

and the data show lower than expected variation (most of the time), as occurs when the probability of a particular integer value becomes large. This second effect causes problems for both model selection and any aggregation technique that looks at the joint probability of a large collection of observations.

Lattice valued data occur when a metric is truncated to fixed precision. In this case, it takes values on some (non-integer) regular grid. These can be handled by looking for the greatest common (rational) divider, by first multiplying values by some fixed constant, so that they are integer, and then using Euclid's algorithm. Since even floating point numbers are rational, this is only worth explicitly handling when the grid spacing is reasonably large, certainly much larger than floating point precision!

D. Low Coefficient of Variation Data

Data with a very low coefficient of variation, i.e. standard deviation divided by mean, can occur when for example monitoring memory usage of a program, such as the "GC heap bytes" category in the canonical data set. Often memory is not returned to the operating system, but kept in pools by a program's memory allocator, so it ratchets up and can remain constant for extended periods. Furthermore, if the memory usage is measured in bytes then the values will be large, often in excess of 1,000,000,000. These data cause numerical stability problems for many techniques that try and determine distribution scale parameters.

E. Fat Tailed and Multimodal Distributions

Fat tailed data are ubiquitous. Many classes of phenomena display power law distributions for large values and APM data are no exception. In APM data, this behaviour can be amplified by the sampling techniques. In particular, monitoring tools often sample maximum values in a time interval that could include many measurements. This results in heavier right tails.

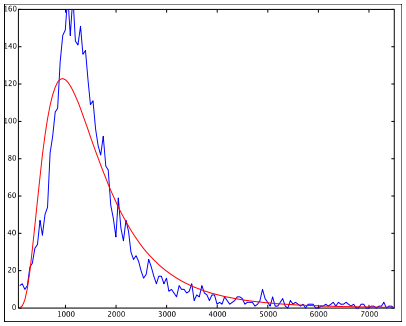

Fig. 4. Count verses "average result processing time" for a specific time series and the maximum likelihood log-normal fit.

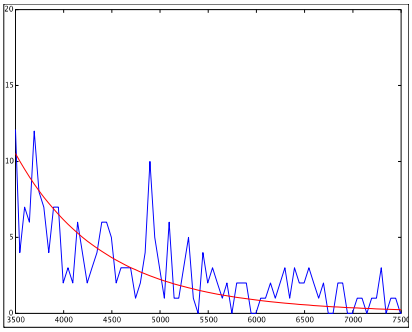

Fig. 5. Detail of maximum likelihood log-normal fit to tail of count verses "average result processing time" for a specific time series.

Fig. 4 shows the long tail behaviour of one particular "average result processing time" time series. In fact, this distribution is closely log-normal as can be seen by the maximum likelihood fit, which has been superposed. Note especially that this accurately fits the counts in the tail, which is important for anomaly detection. Fig. 5 shows the tail fit.

Metrics can have multiple distinct modes when several distinct phenomena are modelled by one metric. For example, if the response time for a number of different SQL queries were grouped into a single metric value then one would expect different modes for the different queries. Fig. 6 shows some of the distinct modes present in one particular "average result processing time" time series (presumably due to different result object sizes).

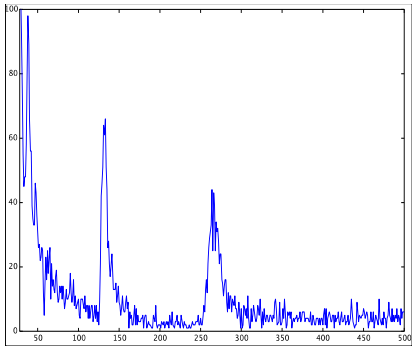

Fig. 6. Count verses "average result processing time" for a specific time series, showing some of the distinct modes present in the distribution.

F. Lags between Anomalies in Different Time Series

When a system problem occurs then often different time series display anomalies at different times. The reasons for this are varied and include: low and sporadic data rates, which mean that no observations are available for extended periods in some time series, and causal relationships which introduce a lag, such as a slow memory leak showing up first as a spike in process footprint and eventually triggering increased response times as the process starts swapping.

Fig. 7 shows anomalousness behaviour in a mixture of two specific "average response time" and three specific "concurrent invocations" time series from the canonical set. Individual time series values have been normalized so that their maximum value in the interval is 100. Note that some of these series, especially "s2" and "s5", contain breaks, when no observations are available. For simplicity, the chart value is set to zero in this case, although clearly any system model must handle missing data points correctly.

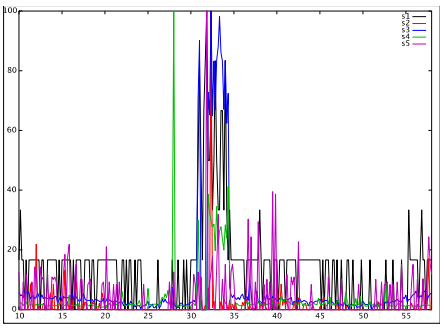

Fig. 7. Normalized "average response time" and "concurrent invocations" verses time displaying an anomaly in the interval 31 to 35 hours after the start of the data set.

All five series display anomalous behaviour during *part* of a four hour period, from 31 to 35 hours after the start of the

data set. However, these anomalies do not all overlap in time. In particular, looking at Fig. 8, we see that whilst the anomalies in time series "s3" and "s4" are well correlated in time, series "s1" is also anomalous in the interval 31 to 32 and the spikes in "s2" and "s5" are relatively short and mutually disjoint.

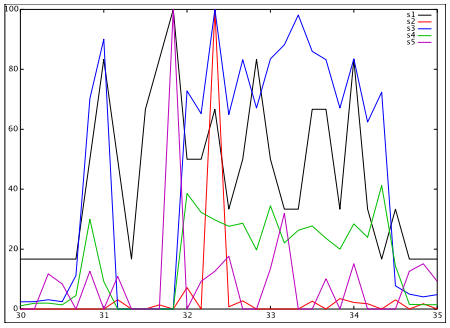

Fig. 8. Detail of the anomalous time period in "average response time" and "concurrent invocations" verses time.

Effects such as this can be handled by doing detection on time bucket statistics, such as the minimum, maximum and arithmetic mean, for a number of different bucket lengths and offsets, or by using dynamic time warping to correlate anomalous behavior in time. For details on dynamic time warping see Muller [16].

V. Results

The large anomaly visible in Fig. 9 manifested itself as system performance degradation during the interval 32 to 35 hours after the start of the data set. In terms of the raw anomaly scores, which were obtained by aggregating individual time series *q*-values, this corresponded to a signal-to-noise ratio of around 330dB. If the time series are ordered by their *q*-values at that time, then 560 of the 33456 time series are significantly anomalous. These results indicated there was an operational issue with a specific component of the backend, which resulted in the response time of a subsection of the website (6 JSPs) having dramatically increased response times. In addition, there was a precursor to the main anomaly, at 27 hours after the start of the data set, which provided the system administrators with early warning of the specific problem before the main failure. This was detected in the performance metrics with a signal-to-noise ratio of around 65dB; however, this was *not* significant enough to result in user noticeable system performance degradation.

As discussed in Section I, once we have a numerical value that allows us to rank all time periods by their anomalousness, we can impose some decision logic on top of this ranking to generate a set of reports about anomalous time periods. A strong requirement for the algorithm to achieve this is that it provides reports as near as possible to the onset of anomalous behaviour. As such, it can only use the anomaly scores received up to an including the time period when it decides to generate a report. Our algorithm for this uses online estimates of historic aggregate q-value quantiles, based on the data structure proposed in Shrivastava *et al*. [17]. Operationally, these reports are presented to the user as alert notifications that inform them in real-time of changes in behaviour of the system. These alerts are generated from a score that is normalized to the range 0 to 100. Fig. 9 show the normalized scores which were generated on the data set, together with our four alert thresholds of increasing severity at 10, 25, 50 and 75.

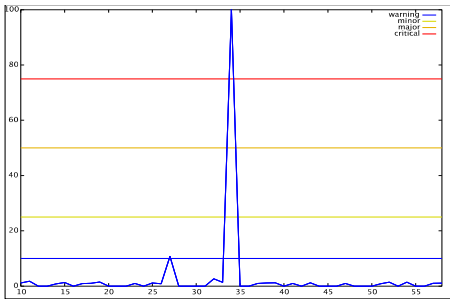

Fig. 9. Normalized anomaly score, based on historic raw anomaly score quantiles, and four alert levels (warning, minor, major, critical).

The effectiveness of this approach was compared to using static thresholds and dynamic thresholds, based on an outlier being 2.5 standard deviations from the rolling mean. In the case of static thresholds, the administrators set the thresholds at such a high watermark, to avoid a stream of false positive alerts, that only a subset of the symptomatic series were identified during the system failure. In the case of dynamic thresholds, the inaccuracies in modelling all series using a Gaussian distribution resulted in a stream of between 1000 and 6000 alerts an hour, with the system degradation alerts being essentially indistinguishable from the noise.

VI. Conclusion

In this paper, we have given a probabilistic definition of anomalousness inspired by the *p*-value concept and show how this can be calculated for various distributions. We also present a numerical scheme for calculating the value when a generating model for the system can be sampled and calculate the limiting distribution for the error in this approximation.

Given this definition, anomaly detection reduces to fitting distributions to data that describe the system behaviour. We discuss those features that must be accurately modelled in order to get good anomaly detection on a canonical application performance management data set.

Finally, we show that the results of combining accurate system modelling and using our definition of anomalousness, picks out a system performance degradation in a real world APM data set with very high signal-to-noise ratio (330 dB). Furthermore, it identifies a small subset of the time series (560 out of 33500) that characterise the anomaly and provides an intuitive ranking of those time series by their anomalousness, which significantly simplifies problem identification and diagnosis.

References

[1] TRAC Research. (2013) Improving the usability of APM data: essential capabilities and benefits. [Online]. Available: http://prelert.com/resources.html.
[2] A. Oliveira. (2013). Why static thresholds don't work. [Online]. Available: http://www.prelert.com/blog/623.
[3] D. Hawkins, *Identification of Outliers*, Chapman and Hall, 1980.
[4] H. Caussinus and A. Roiz, "Interesting projections of multidimensional data by means of generalized component analysis," in *Proc. Computational Statistics*, 1990, pp. 121-126.
[5] H. Liu, S. Shah, and W. Jiang, "On-line outlier detection and data cleaning," *Computers and Chemical Engineering*, vol. 28, issue 9, pp.

1635-1647, 2004.
[6] D. A. Jackson and Y. Chen, "Robust principal component analysis and outlier detection with ecological data," *Environmetrics*, vol. 15, issue 2, pp. 129-139, 2004.
[7] E. M. Knorr and R. T. Ng, "Algorithms for mining distance-based outliers in large datasets," in *Proc. the 24rd International Conference on Very Large Data Bases*, 1998.
[8] M. M. Breunig, H. P. Kriegel, R. T. Ng, and J. Sander, "LOF: identifying density-based local outliers," in *Proc. the 2000 ACM SIGMOD International Conference on Management of Data*, 2000.
[9] H. Fan, O. Za ̈ane, A. Foss, and J. Wu, "A nonparametric outlier detection for efficiently discovering top-n outliers from engineering data," in *Proc. Pacific-Asia Conference on Knowledge Discovery and Data Mining*, Singapore, 2006.
[10] M. J. Schervish, "P values: what they are and what they are not," *The American Statistician*, vol. 50, no. 3, pp. 203-206, 1996.
[11] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.
[12] L. Huang, X. Nguyen, M. Garofalakis, M. I. Jordan, A. Joseph, and N. Taft, "In-network PCA and anomaly detection," presented at the NIPS, 2006.
[13] C. C. Aggarwal, A. Hinneburg, and D. Keim, "On the surprising behavior of distance metrics in high dimensional space," presented at ICDT Conference, 2001.
[14] (2013). Application management-CA technologies. [Online]. Available: http://www.ca.com/us/application-management.aspx
[15] G. Box and G. M. Jenkins, *Time Series Analysis: Forecasting and Control*, 3rd ed., Prentice-Hall, 1994.
[16] M. Muller, *Information Retrieval for Music and Motion*, Springer, 2007, ch. 4, pp. 69-84.
[17] N. Shrivastava, C. Buragohain, D. Agrawal, and S. Suri, "Medians and beyond: new aggregation techniques for sensor networks," in *Proc. the 2nd International Conference on Embedded Network Sensor Systems*, 2004, pp. 239-249.

Thomas J. Veasey has a M.A. (Hons) in physics from the University of Cambridge, UK (graduated in 2000). He has previously worked on radar track extraction and satellite control systems, in the electronic design automation industry and on foreign exchange derivative pricing. He is currently a senior developer at Prelert Ltd where his interests include Bayesian statistical modelling, clustering and convex optimization.

Stephen J. Dodson holds a M.Eng. (Hons) in mechanical engineering and a Ph.D. in computational methods from Imperial College, London (graduated in 1998) alongside a CES from École Centrale de Lyon. His academic research focused on computation of large scattering problems using integral equation time domain methods. He has worked for commercial enterprise software startup for the past 15 years and is currently a founder and CTO at Prelert Ltd. The company is focused on developing innovative software packages that apply novel machine learning techniques to big data in real-time.