Detecting Anomalous Time Series by GAMLSS-Akaike-Weights-Scoring

Preprint · January 2020

CITATIONS
0

READS
275

1 author:

Cole Sodja
Microsoft
32 PUBLICATIONS **1** CITATION

SEE PROFILE

Detecting Anomalous Time Series by GAMLSS-Akaike-Weights-Scoring

Cole Sodja
Microsoft, Redmond Wa

September 20, 2020

Abstract

An extensible statistical framework for detecting anomalous time series including those with heavy-tailed distributions and nonstationarity in higher-order moments is introduced based on penalized likelihood distributional regression. Specifically, generalized additive models for location, scale, and shape are used to infer sample path representations defined by a parametric distribution with parameters comprised of basis functions. Akaike weights are then applied to each model and time series, yielding a probability measure that can be effectively used to classify and rank anomalous time series. A mathematical exposition is also given to justify the proposed Akaike weight scoring under a suitable model embedding as a way to asymptotically identify anomalous time series. Studies evaluating the methodology on both multiple simulations and a real-world dataset also confirm that anomalies can be detected with high accuracy. Both code implementing the algorithm for running on a local machine and the datasets referenced in this paper are available online.

Keywords: Anomaly Detection, Time Series, GAMLSS, Penalized Likelihood

1 Introduction

Univariate time series anomaly detection involves identifying rare or otherwise unusual points in time for a given time series. A related but fundamentally different problem is the detection of anomalous series among a collection as discussed in (Gupta et al. 2014), (Hyndman et al. 2015), (Beggel et al. 2018), and recently in (Talagala et al. 2020). While a universal definition of an anomalous time series is not agreed upon and depends not only on the data but likely on the domain, in this paper anomalous series are viewed as having features that are implausible relative to an understood set of model representations.

The literature discussing the detection of anomalous time series has involved introducing methods to measure deviations of latent features extracted from time series. That is, given a set of time series, the anomalous series are those having the most relatively unusual features in terms of a measure of distance or density. This is a challenging problem as no deterministic labels are readily available indicating which features are considered anomalous, thus making the problem unsupervised. Furthermore, many real world time series are nonstationary and can have complex features and Non-Gaussian noise. Finally, the inherent uncertainty in characterizing the features of time series should be quantified.

A time series y can be modeled as a realization from a random function, that is, given a basis function parameterization θ_y, and associated probability distribution P_{θ_y}, view $y \sim P_{\theta_y}$ as a random sample, called a sample path or just path for short. The vector of basis functions are designed to capture the latent features, such as trend, seasonality, skewness, and so on, and the distribution reflects the degree of noise in the data generating process. It is assumed that there exists such an unknown distribution P_{θ_y} generating y.

Taking the statistical association between a time series and the set of models that yield plausible representations, defining anomalous series can then proceed by working with collections of suitably high-dimensional model embeddings capturing all the variation in normal features and probability distributions, called the normal or null *model space*. A measure can then be defined to compare the distance of the best time series alternative model representations to the families in the null model space, as similarly proposed in (Viele 2001). The anomalous series are then identified as those that have sufficiently small distance to the alternative space, and therefore, large distance from the null space.

The construction of a null and alternative model space determines how paths are characterized, and thus is clearly crucial to the above proposal for defining anomalous series. But how should one proceed with specifying model spaces when the right choice of model families is unknown? This is a model selection problem and is a central theme in this paper. Additionally, the choice of measure to compare models is an important consideration.

In this paper a new algorithm is developed to quantify the uncertainty of selected models for representing time series and quantifying deviations from normal models based on rela-

tive penalized likelihood distributional regression. Mixtures of bases are estimated through leveraging generalized additive models for location, scale, and shape (GAMLSS) introduced in (Rigby and Stasinopoulos 2005). Comparing models is done based on Kullback-Leibler divergence which is then used to construct an interpretable likelihood score to measure the plausibility of models selected. The likelihood scores are efficiently approximated using Akaike weights, which have a rigorous justification linked to information theory, see (Akaike 1973) and (Akaike 1981) or (Burnham and Anderson 2002). Applying Akaike weights hasn't been proposed in the context of anomaly detection to the best of the author's knowledge, though they have been shown to yield competitive results for model averaging in time series forecasting, see (Kolassa 2011). The abbreviation *GAWS*, for GAMLSS-Akaike-Weights-Scoring is used henceforth to refer to the proposal for integrating these two methodologies for anomaly detection. This proposal is extensible enough to accommodate scoring paths inferred from collections of time series that come from different classes. Furthermore, it can handle series with missing values or irregular time points, nonstationarity, nonlinearity and Non-Gaussian noise.

The remainder of this paper proceeds as follows: Section 2 provides an overview of related research work. Section 3 motivates the GAWS solution by framing the central problem of path detection with a statistical formulation of model embeddings, the connection of information theory to the model selection problem, and the properties of Akaike weights. Section 4 overviews GAMLSS and outlines parameter estimation based on penalized likelihood. Section 5 describes the GAWS algorithm. Section 6 references all software used in this paper. Section 7 describes model performance in simulation studies and sensitivity of GAWS to different data. Section 8 shares empirical results benchmarking GAWS on a real world dataset. Finally, Section 9 concludes with a summary and lists future research considerations. A section on supplementary materials including links to all the data and code is also provided at the end.

2 Related Work

Statistics research discussing the detection of entire anomalous series has been mentioned in various settings, such as for the identification of failures in space shuttle valves (Salvador and Chan 2005), server monitoring (Hyndman et al. 2015), or tracking star light curves in astronomy (Twomey et al. 2019). Functional data clustering and curve (or shape) detection are also closely related research areas. Some papers of note discussing applicable methodologies are summarized below.

A probabilistic approach for comparing inferred basis functions including in the presence of multiple, unknown heterogeneous subspaces was presented in (Schmutz et al. 2018). An extension of a Gaussian mixture model for multivariate functional data is developed with functional principal component analysis (FPCA). While this method is model-based, it is sensitive to the selection of the number of clusters, requires choosing the number of eigenfunctions, and yields basis functions that are not recoverable as no constraints are put on the principal components.

A general purpose though difficult to interpret algorithm for finding anomalous time series was proposed in (Beggel et al. 2018). They defined subsequences of time series of a fixed length called "shapelets" to create a feature vector, and then applied a kernel approach to the features using a support vector data description (SVDD) algorithm to detect anomalies of the paths which lie outside an estimated decision boundary. Experiments showed that this approach was able to discover several types of unusual paths, particularly for smooth series, but it had problems for time series with high variability. Moreover, understanding the type of anomalies that can be discovered via shapelet learning is challenging.

An explainable method for identifying anomalous series was given in (Hyndman et al. 2015). A context-relevant feature space is first constructed, then PCA is applied to reduce dimensionality and finally a multivariate outlier method based on learning an α convex hull is applied to generate a ranking of anomalous series. A nice advantage of this method is that the anomalies detected can be controlled through careful construction of interpretable features. However, it can take substantial effort to expand the feature space for new diverse data sets, compared to basis function learning.

Another general and recent anomaly detection algorithm termed *stray* applicable for high-dimensional data including time series was proposed in (Talagala et al. 2020). This approach also works with extracted features of time series but incorporates a density-based measure using extreme value theory together with a modification to k-nearest neighbor searching. This method is used as the main benchmark to compare GAWS performance.

Research that is similar to the GAWS proposal working with basis function expansions, such as penalized splines was discussed in (Abraham et al. 2003) for clustering, and in (Tzeng et al. 2018) for defining dissimilarity. Pairwise comparison of functions via their spline basis coefficients using metrics such as L2 or dynamic time warping has been shown to yield good performance in discriminating among clusters. In fact, (Tzeng et al. 2018) construct a modified L2-distance that accounts for the uncertainty in smoothing spline estimates, yielding a measure of dissimilarity of functions that is integrated with a nearest neighbor algorithm to identify clusters and anomalies. Experiments showed that their approach can be preferable compared to model-based clustering. However, the authors only consider learning a conditional expectation using a natural spline, and extending their proposal to ranking anomalies based on conditioning on other moments like the variance or skewness is not mentioned. The GAWS methodology can be viewed as a generalization of this work, while also offering a theoretically justified alternative and interpretable measure.

Finally, it's also worth mentioning that from the Bayesian perspective, Gaussian processes (GP) are fully probabilistic and yield interpretable bases in terms of choice of reproducing kernels. GP have been successfully applied for analyzing functional data for functional clustering and classification tasks, e.g. see (Shi and Choi 2011), as well as detecting anomalous series, see (Pimentel et al. 2013), and (Twomey et al. 2019). However, estimating GP can be computationally expensive compared to penalized MLE.

3 Motivation: Framing the Problem of Path Detection

A formal statistical framework is proposed in this section for defining anomalous entire time series motivated by model embeddings and an information theoretic measure. A review is also given on the use of information criterion measures for model selection, and a

justification provided for the use of Akaike weights in scoring anomalies.

3.1 Model Embedding Formulation

The perspective is to treat each univariate time series as a sample path coming from an existing but unknown probability distribution that has basis functions as parameters capturing the location, scale and shape of the distribution, or more succinctly features. For the remainder of the paper, y will denote a real-valued path over a continuous time interval T coming from a probability distribution P_{θ_y}, where θ_y is a vector valued function over T representing the features; when the particular association with a given time series is not needed the subscript will be dropped and the notation θ used to represent features.

It is assumed that the paths can be embedded into a suitably large but finite set of models consisting of distribution families and basis function classes such that the basis functions well approximate the features. Formally, if the paths live in an infinite dimensional space then there exists a countably dense subspace where the shapes have a pointwise convergent representation consisting of finitely many orthogonal basis functions. While this assumption may seem restrictive, it covers a substantial class of time series that have finite mean and covariance living in spaces with well-behaved finite representations, namely the reproducing kernel Hilbert spaces, see (Berlinet and Agnan 2004) for mathematical details.

Let $b : T \to \mathbb{R}$ be a basis function generating a finite dimensional subspace $\mathcal{H}_b = span\{b_j | j = 1, ..., dim_b < \infty\}$, and $\mathcal{B} := \{b : T \to \mathbb{R}\}$ be a class of basis functions that approximate the space of features for the collection of paths. Define $\mathcal{F} := \{\mathbf{f} := (P_{\mathbf{\Theta}}, \mathcal{H}_b) | b \in \mathcal{B}\}$ to be the class of model families, where the elements are different distribution families and orthogonal basis functions. Let $\mathcal{M}_{\mathbf{f}} := \{\mathbf{M_f} := (\mathbf{f}, \mathbf{a}_b, | \mathbf{f} \in \mathcal{F}\}$, where $\mathbf{a}_b : \mathbb{N} \to \mathbb{R}$ is a vector of coefficients dependent on basis function b. Define $\mathcal{M} := \bigcup_{\mathbf{f}} \mathcal{M}_{\mathbf{f}}$, referred to as the model space, with elements called models.

The above construction produces a finite model space as an idealization of the space of unknown distributions P_{θ_y}. It is further taken that this model space can be decomposed as $\mathcal{M} = \mathcal{M}_0 \cup \mathcal{M}_a$, where each $P_\theta \in \mathcal{M}_0$ is associated with a subset of paths with

features designated as "normal", and $\mathcal{M}_a := \mathcal{M} \setminus \mathcal{M}_0$ contain models yielding "feature anomalies". Given a new path y and uncertainty about its model P_{θ_y}, to assess if y is anomalous, that is, if it has features anomalies, a measure π is introduced to quantify how much P_y deviates from models in $\mathcal{M}_0$. The values associated with this measure π_y are deemed anomaly scores or simply scores. Formally, given a sufficiently small threshold value α, a binary classification of y being assigned as an anomalous path is given based on $\pi_y := \pi(P_y, \mathcal{M}_0) < \alpha$. A ranking of which paths are considered most anomalous also readily follows based on ordering scores in ascending order. Hence forth, the problem of ranking or classifying feature anomalies of time series is referred to as path detection, and the term path anomaly will be adopted. Note that in the literature, shape anomaly and shape detection are defined. However, to avoid confusion, using shape to refer to the distribution properties in GAMLSS as well as features of time series will be avoided.

While there are different choices of measures for comparing distributions, here the Kullback-Leibler divergence is used to construct a proper relative likelihood score.

Framing the identification of anomalous paths based on model embeddings and a KL-divergence measure follows a similar approach as given in (Viele 2001), where the author attempts to quantify how close functional models are to an idealized data generating process for evaluating lack of fit. It is perhaps worth noting that the idea of embedding a sample path representation into a large family of models comprised of both normal and alternative basis functions and then using KL-divergence to define anomaly scores in this paper was not motivated by the work of Viele but rather inspired from the philosophy of multimodel inference as discussed in (Burnham and Anderson 2002). Indeed, the perspective taken here is that model selection should be central to how inference is carried out.

3.2 Model Selection via Expected KL-Divergence

Recall that KL-divergence is given by:

$$D^{KL}(P_M, P_N) := \int P_M[y] \cdot \log[P_M[y]] - \int P_M[y] \cdot \log[P_N[y]] \tag{1}$$

The choice of KL-divergence as an appropriate measure of closeness between an alternative probability distribution and its data generating process is warranted both for its nice

interpretation as well desirable asympototic properties. Specifically per (Berk 1966), while not a proper metric, it can be shown that it measures the long term loss involved from using the wrong model given the data, and thus, is very natural to use for model selection.

In practice, both the true data generating process (again assuming it exists) as well as the null model space $\mathcal{M}_0$ comprised of the normal paths are unknown and thus must be estimated relative to the data. Introduce the notation $\mathcal{Y}_N$ to be a collection of N sample paths generated from P_{θ_y}. In the literature this data is referred to as functional data, and inference of such data called functional data analysis, see (Ramsay and Silverman 2005) for a classic reference. Let $\hat{\theta}_{0y}$ denote an estimate of the features for a specific path y for some model $P_{\hat{\theta}_{0y}} \in \mathcal{M}_0$, and $\hat{\theta}_0$ denote an arbitrary estimate. Even for a single path there are potentially many models given there is uncertainty, and in Section 4 an approach using GAMLSS will be outlined to generate the $P_{\hat{\theta}_0}$.

The initial measure of interest to quantify the average relative loss from selecting models with normal paths to represent the data generating process is given by:

$$D^{KL}_{\mathcal{Y}_N} := \inf_{\hat{\theta}_0} \frac{1}{N} \sum_{y \in \mathcal{Y}_N, \hat{\theta}_{0y}} D^{KL}(P_{\theta_y}, P_{\hat{\theta}_{0y}}) \tag{2}$$

Note that $D^{KL}_{\mathcal{Y}_N}$ is a statistic, dependent on the sample size N and variation among the paths $y \sim P_{\theta_y}$. Understanding its distribution is of primary importance. Define $\mathbb{D}^{KL}_0 := \mathbb{E}_{\mathcal{Y}_N}[D^{KL}_{\mathcal{Y}_N}]$ to be the expected value over the null model space. Following Section 4 Main Theorem in (Berk 1966), assuming certain boundedness conditions hold, and $P_{\theta_y} \in \mathcal{M}_a$ then as N increases the sampling distribution of $D^{KL}_{\mathcal{Y}_N}$ will be a point mass at $\mathbb{D}^{KL}_0$. This is an extremely valuable theoretical result as it ensures that eventually with enough data and a suitable model space the anomalous paths can be detected with arbitrary high probability in terms of the relative average KL-divergence.

While theoretically justified per the above discussion, it is difficult to directly work with KL-divergence as again the true data generating process cannot be compared against. A model selection procedure that would result in yielding at least an unbiased estimate of $\mathbb{D}^{KL}_0$ would be desirable. This is where the Akaike information criterion (AIC) is useful.

Recall that the penalized negative log likelihood of a model $\mathbf{M}$ with $\nu_{\mathbf{M}}$ effective degrees

of freedom conditional on a path y is given by:

$$NLL_{pen,y}[\mathbf{M};\lambda] := -2\ell[\mathbf{M};y] + \lambda[\nu_{\mathbf{M}}, y] \tag{3}$$

where ℓ is the log likelihood function associated with its distribution family $P_{\boldsymbol{M}}$, and $\lambda[\nu_{\mathbf{M}}, y]$ is a penalty; for instance, the calculation of AIC has $\lambda = 2\nu_{\mathbf{M}}$, and for the Bayesian Information Criterion (BIC), $\lambda = \log[length(y)]\nu_{\mathbf{M}}$.

The penalized negative log likelihood defined in Equation (3) is also known as the generalised information criterion (GIC), see (Konishi and Kitagawa 1996).

Computing the penalty in Equation (3) not only relies on knowing how to compute the likelihood but also the effective degrees of freedom. While it is straightforward to estimate this statistic as the number of parameters when working with fully specified parametric models, it need not be obvious for nonparametic models. This issue will be revisited in Section 4.

It can be shown that the AIC computed across all paths $y \in \mathcal{Y}$ is an unbiased estimator of $\mathbb{D}_0^{KL}$, and hence, given N is large enough, minimizing AIC is equivalent to working with the average relative KL loss $D_{\mathcal{Y}_N}^{KL}$, see (Akaike 1973) or (Burnham and Anderson 2002).

While the estimate that minimizes AIC won't necessarily identify the true model P_{θ_y}, utilizing the BIC does guarantee this asymptotically, see (Rao and Wu 1989). However, BIC is not asymptotically efficient, and so for small sample sizes relying on it for model selection can result in underfitting. Therefore, it is important to carefully choose a penalty on the model complexity in accordance with the data and task; this issue is later revisited in the simulations section.

A particularly nice property of AIC or BIC is that they can be used to compare different choices of distributions and basis functions, as long as the likelihood is computed on the same time series.

Use of the AIC or BIC for model selection does have its limitations in practical applications. For example, for univariate time series with short sample sizes a modification is often used to correct for bias, but this doesn't correct for unstable parameter estimates. Also, it's known that AIC tends to favor more complex models than what is needed, whereas relying on BIC for selecting the correct complex model when sample sizes are even modest

may fail. While other penalty functions beyond AIC and BIC can be considered, it still is not clear which information criterion is better suited for each particular data set. Probably the biggest limitation is that information criterion-based measures are relative and tell you nothing about the accuracy of the models. Not including enough model families will bias inference, as is later demonstrated through simulations. As always, it is recommended to carefully consider the context when settling on a collection of model families, and perform diagnostic checks where possible to validate model reasonableness.

3.3 Akaike Weights

Given there is uncertainty in both the selection and estimation of models, entertaining all plausible models that may have close AICs should be done rather than relying on selecting a single model. Quantifying the relative probability of choosing the minimum AIC would thus be useful in yielding an interpretable score that could be thresholded. This is where Akaike weights are particularly attractive. The relative log odds of a model given a path is defined by:

$$\Delta_{\mathbf{M},y} := NLL_{pen,y}[\mathbf{M};\lambda] - min_{\mathbf{M}}\{NLL_{pen,y}[\mathbf{M};\lambda]\} \tag{4}$$

and the *Generalized Akaike Weights* is given by:

$$\pi_{\mathbf{M},y} := \exp(-\frac{1}{2}\Delta_{\mathbf{M},y}) / \sum_{\mathbf{M}\in\mathcal{M}} \exp(-\frac{1}{2}\Delta_{\mathbf{M},y}) \tag{5}$$

Anomaly scores for a given time series marginalizing across the set of all models defining the normal paths can then be computed as:

$$\pi_y := \sum_{\boldsymbol{M}\in\mathcal{M}_0} \pi_{\mathbf{M},y} \tag{6}$$

The scores given by summing the Akaike weights over all possible models representing the normal paths are interpretable, and computationally straightforward, so as long as the penalized likelihood per model is available.

Leveraging the results published in (Akaike, 1973, 1974, 1981) that establish that the Akaike weights provide a measure of the relative likelihood of a model approximating the

lowest expected KL-divergence, and given the asymptotic results of the posterior distribution of $D^{KL}_{\mathcal{Y}_N}$ discussed in (Berk 1966), the main result mathematically justifying the proposal for scoring anomalies follows.

Corollary 1: Convergence of Akaike Weight Scores

Under the assumption that a collection of paths $\mathcal{Y} := \{y\}$ are generated from a finite set of models $\mathcal{M} = \mathcal{M}_0 \cup \mathcal{M}_a$ composed of those with normal features and anomalous paths respectively, and the boundedness conditions required for the use of the Lebesgue dominated convergence theorem hold per (Berk, 1966), if a sample path y has data generating process $P_{\theta_y} \in \mathcal{M}_a$, then π_y will converge to a point mass distribution at 0. Consequently, under the above construction, anomalous series can be detected with high probability given enough samples by ranking or thresholding the Akaike weight scores.

4 GAMLSS Penalized Likelihood Estimation

Generalized additive models for location, scale, and shape (GAMLSS) introduced in the paper (Rigby and Stasinopoulos 2005) is a powerful toolbox for building and comparing semiparametric models for the purpose of distributional regression, complimenting other nonparametric methodologies such as quantile regression and Gaussian processes regression. In short, GAMLSS extends generalized additive models to allow fitting probability distributions both inside as well as outside the exponential family. While there are several ways to estimate parameters in GAMLSS, the discussion will be limited to penalized likelihood estimation. For a more recent review of GAMLSS with emphasis on penalized likelihood estimation as well as detailed coverage of the gamlss R package see (Rigby et al. 2017) or (Rigby et al. 2020). Other references discussing alternative estimation include (Schlosser et al. 2019), (Umlauf et al. 2017), (Adam et al. 2017) and (Mayr et al. 2012).

Estimation involves finding all parameters of a specified distribution conditional on regressors, including parameters of higher-order moments, like variance, skewness and kurtosis. Since GAMLSS are additive models, they support fitting flexible basis functions

capable of learning complex shapes while still yielding interpretable relations. For time series specifically, basis functions provide a natural way to generate a decomposition into meaningful unobservable components, like trend, seasonality, change points, etc.

GAMLSS maximizes a penalized likelihood function dependent on the chosen distribution. Here the input are univariate time series $y = (y_t)$, conditional on an unknown finite dimensional vector of functions $\mathbf{\Theta} := (\theta_1, ..., \theta_p)$, with a parametric distribution $P_{\mathbf{\Theta}}$. Discrete realizations y_i are assumed to be dense taken over a common time grid T. Each y_i are padded with missing values as needed. Let $\mathbf{X}_T := (\mathbf{t}_j)$ be a matrix of common regressors extracted over T. For example, this would contain a feature for sequence of time $t \in T$, as well as other extracted calendar-based features like hour of day, day of week, and so on. Additional features could easily be incorporated.

For each $m \in \{1, ..., p\}$, a decomposition into finitely many basis functions under chosen link functions g_m are produced, yielding path representations given by:

$$\eta_{m,t} = g_m[\theta_{m,t}|\mathbf{X}_T] = \sum_{j=1}^{j=d_m} b_{jm}[\mathbf{t}_j] \tag{7}$$

where each basis function can be written in the form $b_{jm} = B_{\mathbf{t}_j m}\mathbf{a}_{jm}$, $B_{\mathbf{t}_j m}$ is an associated basis matrix dependent on time, and $\mathbf{a}_{jm}$ is a vector of coefficients subject to a quadratic penalty $\mathbf{a}_{jm}^T G_{\mathbf{t}_j m}\mathbf{a}_{jm}$.

The matrices $G_{\mathbf{t}_j m}$ are symmetric with a generalized inverse that is a variance-covariance matrix dependent on a vector of hyperparameters used to define penalties λ_{jm}. The penalty vectors need to be provided, though learning can be done to find an optimal value maximizing the marginal likelihood or applying generalized cross-validation (GCV), which is the approach taken here.

Then using the RS-algorithm as proposed in (Rigby and Stasinopoulos 2005), a penalized log-likelihood is maximized as defined by:

$$\ell_{pen}\left(\mathbf{\Theta}; \lambda|y\right) = \sum_{i=1}^{i=N} \log\left[P_{\mathbf{\Theta}}\left(y_i\right)\right] - \frac{1}{2}\sum_{m=1}^{m=p}\sum_{j=1}^{j=d_m} \lambda_{jm}\mathbf{a}_{jm}^T G_{\mathbf{t}_j m}\mathbf{a}_{jm} \tag{8}$$

The RS-algorithm effectively takes an iterative approach to solve the penalized likelihood using cycles of iteratively reweighted least squares with Fisher scoring and applying a modified backfitting algorithm (Buja et al. 1989).

One benefit of working within the penalized likelihood framework is that models can be compared to assess their predictive performance without the need to work with holdout data. Per Equation (3), this requires knowing the effective degrees of freedom (EDF). In the case that a model has a known number of fixed parameters this is straightforward to compute. However, computing the EDF for more complex models is not obvious and often depends on the choice of basis function. This is handled in the proposed GAWS implementation working with linear smoothers by default, specifically B-splines with a fixed degree (1,2 or 3) and a sufficiently large number of knots, where the EDF is then given by the trace of a smoother matrix, see (Eilers and Marx 1996).

5 GAMLSS-Akaike-Weights-Scoring

The GAMLSS-Akaike-Weights-Scoring (GAWS) algorithm is introduced in this section as a solution for either binary classification or ranking of anomalous series within a collection.

5.1 GAWS Algorithm Design

The algorithm is designed to operate on different classes of univariate discrete or continuous time series, where there are sufficiently many high frequency series per each class so that basis function learning is feasible. While it is possible to apply the algorithm to a collection of sparse functional series through a modification using mixed GAMLSS and specifying hierarchies, the implementation described in this paper performs parameter estimation per each time series separately.

Analogous to feature-based models, or selection of kernels for GP learning, a particular choice of both distribution families and basis functions must be made. Naturally it is possible to use a default set of very flexible continuous and discrete distributions with parameters for location, scale, and shape consisting of generic bases, such as penalized splines or Fourier expansions for periodic series, but additional customization reflecting the nuances of a given domain is likely required to optimize both computational performance and accuracy.

A nice feature of the GAWS algorithm is that initialization of the null and alternative model spaces, scoring and detecting anomalous series, and updating the model spaces are separate components that can all be parallelized. From this perspective, the GAWS algorithm offers a solution that can scale even on massive sets of time series by leveraging cloud computing.

The GAWS algorithm proceeds with selecting model families, estimating parameters, performing dimensionality reduction, scoring paths and detecting anomalies.

Figure 1 depicts the computational steps involved.

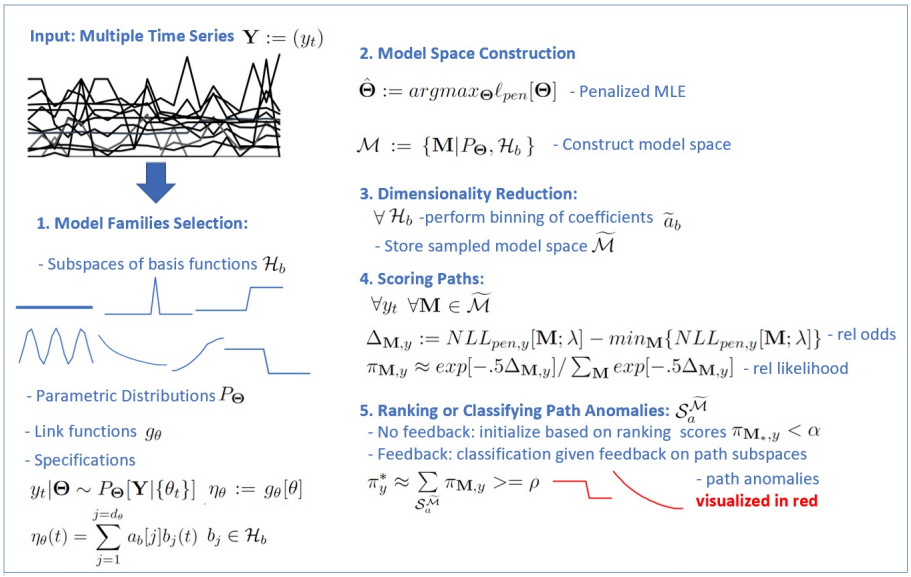

Figure 1: GAWS Framework

5.2 Model Families Selection

The most important factor influencing the performance of GAWS in detecting anomalies is the choice of basis functions and distributions families. Working with a default combination of common basis functions, such as penalized splines has shown to work reasonably well on real world time series as will be shown in Section 8.

However, given the diversity of parametric distributions it is not straightforward nor robust to settle on such a default. Relying on an expert to identify candidate distributions accounting for the problem context and validating against data via diagnostic checks is

recommended to maximize GAWS performance. In the situation where expert elicitation is not practical, a data driven consideration for selecting distributions is to search over a large enough set and retain all that have Akaike weights above a defined tolerance threshold. Alteratively, a forward search could be done, for example, utilizing the fitDist() function in the gamlss package to find the distribution with the lowest GIC among all distributions having valid range, e.g. over the positive real line.

Automation of model family selection is an area for further research.

5.3 Rescaling Time Series

GAWS is flexible enough to quantify multiple types of anomalies, including paths with levels that have extremely high or low values, so-called *magnitude anomalies*, as well as paths with unique or rare characteristics that otherwise have similar ranges of values. The focus of this paper is on quantifying deviations in the characteristics across paths and not magnitudes. Therefore, all paths are rescaled to have the same level. Given non-negative y_t with values typically below an upper bound u, replace y_t with $\exp[w_t]$ as follows:

$w_t := \log(y_t + 1)$

$w_t := (w_t - mean(w_t))/(\max(w_t) - \min(w_t))$

$w_t := w_t + \log(u)$

For time series having valid negative values the log transformation step is skipped.

5.4 Model space Construction

The GAWS algorithm currently operates by estimating GAMLSS for each univariate time series. For each y, and for every model specified $\mathcal{M}_{\mathbf{f}}$ from the selected model families $\mathbf{f}$, compute the basis function expansions for each location, scale, shape using penalized likelihood estimation as detailed in Section 4 to obtain models $\mathcal{M}_{y,\mathbf{f}}$.

This produces a model space $\mathcal{M} := \bigcup_{y,\mathbf{f}} \mathcal{M}_{y,\mathbf{f}}$ which should be suitably large enough to reflect the normal and anomalous paths given enough y and appropriately selected $\mathbf{f}$.

5.5 Dimensionality Reduction

The constructed model space $\mathcal{M}$ might require an excessive amount of storage, particularly when dealing with a massive set of time series or many model families. Moreover, for the purpose of identifying anomalies within each subspace, it's sufficient to cluster or summarize the range of basis coefficients into bins. Thus, a dimensionality reduction step can be employed as necessary to help minimize storage space.

Because of the potential variety of model families, learning a joint distribution across all models is problematic. Therefore, the approach taken here is simple, ignoring the potential gains that could be obtained from applying better multivariate methods by doing it univariately and creating frequency bins for $\mathbf{a}_b|(P_{\mathbf{\Theta}}, \mathcal{H}_b, \nu_{\mathbf{M}})$. Using histograms to represent basis expansions should also not be problematic if using orthogonal basis functions.

To construct the reduced model space first initialize $\widetilde{\mathcal{M}}$ to the empty set, and configure the basis coefficient bins $\mathbf{I}_b = \left\{[i_{b_k}, i_{b_{k+1}})\right\}$. Then for each model family $\mathbf{f} = (P_{\mathbf{\Theta}}, \mathcal{H}_b, \nu_{\mathbf{M}})$, compute the sampling distribution $\widetilde{\mathbf{a}}_b := (\mathbf{I}_b, freq(\mathbf{a}_b \in \mathbf{I}_b))$, and update
$\widetilde{\mathcal{M}} := \widetilde{\mathcal{M}} \cup \{(\mathbf{f}, \widetilde{\mathbf{a}}_b)\}$.

5.6 Scoring Paths

To identify a candidate subset of the null model space $\mathcal{M}_0$ associated to a subset of paths that are normal, denoted $\mathcal{S}_0$, it is important to discriminate among models that are plausible in the sense of having a minimum positive Akaike weight.

It is hoped that the right model families are chosen up front to reflect the majority of patterns seen. However, to guard against outliers influencing the bases or misspecified models, two user driven control parameters are introduced.

Let α define the minimum value such that each Akaike weight $\pi_{\boldsymbol{M},y}$ must exceed to be considered plausible relative to the union of models $\{\boldsymbol{M}_y\}$ estimated for y and other models $\{\boldsymbol{M}_{y_0}\}$. Initialize $\mathcal{M}_0$ to be the set of models $\boldsymbol{M}_{y_0}$ satisfying $\pi_{\boldsymbol{M}_{y_0},y} >= \alpha$.

Define N_{min} to be the minimum number of unique series associated with models $\boldsymbol{M}_{0\boldsymbol{f}}$ needed to differentiate a model family $\boldsymbol{f}$ among extremely infrequent model families.

Let $\mathcal{F}_0$ be the collection of model families satisfying count$(\boldsymbol{M}_y \in \boldsymbol{f}_0) >= N_{min}$, and

update $\mathcal{M}_0 = \mathcal{M}_0 \cap \mathcal{M}_{F_0}$.

A final step removes models associated with anomalous series by thresholding the scores of series π_y relative to $\mathcal{M}_0$. This is necessary if there are enough multiple anomalies of the same class that are plausible for one of the model families, which can happen if a model family has basis functions of high complexity.

For each series y_* satisfying $\pi_{y_*} < \alpha$, update the set of path anomalies $\mathcal{S}_a = \mathcal{S}_a \bigcup \{y_*\}$, remove the associated models $\boldsymbol{M}_{y_*}$ from $\mathcal{M}_0$, and initialize the alternative model space $\mathcal{M}_a := \mathcal{M} \setminus \mathcal{M}_0$.

Setting α and N_{min} depend on the application requirements, the number and complexity of subspaces, number of series, and the amount of noise. Simulations and real world experiments show that setting α at values within the relatively large range of 0.0001 up to 0.02, and N_{min} for values as low as 3 or as high as 15% of the sample size yield competitive performance in the upsupervised setting, even in the presence of moderately noisy time series. See Sections 7 and 8.

5.7 Ranking and Classifying Path Anomalies

Once the model space is constructed and a decomposition available, each new series y is scored against $\mathcal{M}_0$ employing the Akaike weights to generate π_y. A binary classification can then be performed to designate y anomalous if $\pi_y < \alpha$, or the scores can be ranked.

If feedback becomes available, e.g. a domain expert flags candidate anomalous series as a false positive or false negative, the model space could be updated by creating basis function representations for the flagged series and inserting into the model spaces $\mathcal{M}_0$ and $\mathcal{M}_a$ respectively.

Finally, if identifying particular types of anomalies defined based on a set of a priori attributes considered alarming is the objective, then the alternative scores denoted by $\pi_y^* := 1 - \pi_y$ can be controlled to achieve a specified expected precision given a parameter $\rho \in (0, 1)$. For this problem, it is not necessary to construct the entire model space across all series, but simply compute the score for the null model space of model families against the alternative for each series, which will significantly speed up computations.

6 Software Implementation

The R programming language (R Core Team 2020), version R x64 3.5.1 was utilized for analyzing the data described in this paper, running simulations, implementing GAWS, and producing and comparing results.

The specification of penalized b-splines and the computation of penalized likelihood estimation was done using the gamlss package, see (Stasinopoulos et al. 2020).

To compare the performance of GAWS in detecting path anomalies, two other methodologies were considered: Time series feature extraction with PCA and multivariate highest density region based on the R package anomalous, see (Hyndman et al. 2018). The second method uses the R package stray, see (Talagala et al. 2020).

These methods are abbreviated *hdr* and *stray* respectively for reporting results. Note that only the default time series extractor provided by the packages was utilized. It is likely that if custom features were built for the specific simulated data then results would change.

7 Simulation Study

A set of synthetic datasets were generated and the performance of GAWS evaluated to assess how well it does finding different types of path anomalies.

This study implemented GAWS on a single machine. A sample size up to 200 time series per simulation run was chosen to yield a reasonable representation while being able to keep run times to a few hours on a laptop with 4 cores and 16GB RAM running 64 bit Windows. Each time series was measured hourly across 21 days, and composed of location, scale, and shape bases generated from 10 different basis function classes.

The specific distributions, basis functions and parameters chosen in the simulations were based on analyzing real cloud traffic data as outlined in section 8, involving outliers and Non-Gaussian heavy tailed distributions.

The BIC was chosen as the penalty function in the computation of the Akaike weights. While the AIC or some other criterion could have been assessed, BIC was selected given

one goal of the study was to show that GAWS works well discriminating between the null and alternative model spaces with modest sample sizes.

The number of time series in each subspace was randomly assigned and constrained to be at least 10; this minimum sample size seemed reasonable to reflect the sampling variation in the parameters considered and frequency of anomalies tested.

7.1 Simulated Basis Functions and Parameters

Sample paths are defined based on a composition of additive models under log link functions for the Box-Cox Cole and Green (BCCG) distribution. The BCCG distribution as defined in (Cole and Green 1992) is useful for modeling positive continuous data with excess skewness. It is parameterized via location, scale and shape stochastic processes as follows:

$$\begin{aligned} \text{BCCG}[y_t|\mu_t,\sigma_t,\nu_t] &:= \frac{1}{\sqrt{2\pi}\sigma}\frac{y_t^{\nu_t-1}}{\mu_t^{\nu_t}}\exp[-\frac{z_t^2}{2}] \\ if\ \nu_t \neq 0,\ z_t &:= [(\frac{y_t}{\mu_t})^{\nu_t}-1]/(\nu_t\sigma_t),\ \text{otherwise } z_t := \log[\frac{y_t}{\mu_t}]/\sigma_t \end{aligned} \tag{9}$$

Formally, each time series y_t are independent, identically distributed samples from $P_{\mathbf{\Theta}}$, where the function parameters $\mathbf{\Theta} = \{\theta_t = g_\theta^{-1}[\eta_{\theta t}]\}$ are given by log canonical link functions g_θ^{-1} and $\eta_{\theta t} = \sum_{m=1}^{m=M} b_{m,t}$ represented in terms of finitely many prespecified basis functions $b_{m,t}$.

For location parameters, the default basis is constructed from the local level with daily and weekly seasonality simulated from a double seasonal innovations state space model as posed in (Gould et al. 2008). Additional bases include random pulses, autoregressive processes, and linear step functions to capture level shifts.

For scale parameters, constants sampled from a grid of values between 0.05 to 0.25 forms a default basis. The random walk with slow growth is also considered to capture path anomalies in increasing variance. For the BCCG distribution shape parameter, i.e.

skewness, a range of constant values randomly sampled between -0.5 and 0.2 is considered, which maintains right-tail skewed data.

Let $\epsilon_t \sim N(0, \sigma^2)$ be iid Gaussian noise, where for each time series, y_t, $\sigma^2 = cv \cdot (1 + y_1)$, where cv is the coefficient of variation, generated from a truncated log-normal distribution with location $\log(0.05)$, and scale 0.25. These priors were chosen to evaluate a range of very low to medium relative variance in the location parameters. Table 1 shows the collection of bases.

Basis Name	**Specification**	**Parameter Constraints**
Local Level	$L_t := L_{t-1} + \alpha\epsilon_t$	α randomly between 0 to 0.15 L_0 is sampled from a truncated log-normal with mean log(500) and range between 350 and 650
Double Seasonal	$S_t := S^{(1)}_{t-m1} + S^{(2)}_{t-m2}$ $S^{(1)}_t := S^{(1)}_{t-m1} + \gamma_1\epsilon_t$ $S^{(2)}_t := S^{(2)}_{t-m2} + \gamma_2\epsilon_t$	time periods m1:=7, m2:=24 γ_1, γ_2 randomly between 0.001 and 0.1 $S^{(1)}_0 := L_0 \cdot [0.27, 0.25, 0.24, 0.21, 0.12, -0.52, -0.57]$ $S^{(2)}_{0,n} := L_0 \cdot \sum_k c_k sin(2\pi kn/m2) + d_k cos(2\pi kn/m2)$ $n = 0, ..., 23$, $c_1 = 0.1, c_2 = -0.2$ $d_1 = -0.5, d_2 = -0.2$
Random Pulse	$\exists \chi_t \sim$ Bernoulli$[\pi_t]$ and δ_t $x_t := \delta_t$ if $\chi_t = 1$ and 0 otherwise	π_t randomly selected between 0 to .01 $\delta_t = L_t \times r$, r randomly between 3 and 6
AR(P)	$x_t := \sum_{i=1}^{i=P} \phi_i x_{t-i} + \varepsilon_t$	P sampled from a zero-adjusted Poisson location = 0.2, proportion of zeros = 0.75, ϕ_i randomly between 0.05 and 0.25
Random Walk Drift	$x_t := x_{t-1} + b + \epsilon_t$	$b = r \cdot L_0$, r randomly chosen between 0.0001 and 0.002
Linear Step	$\exists \tau_1 \leq \ ... \leq \tau_N$, and $\exists \delta_1, ..., \delta_N$ $x_t := \sum \delta_i 1_{[\tau_i, \tau_{i+1})}(t) + \epsilon_t$	N randomly set to 0 with probability 0.9 and 1 otherwise τ_i is a randomly sampled integer between 50 and 450, and $\delta_i = L_i r$ is a random pulse, r between 0.3 to 0.7 or 1.4 to 2

Table 1: Simulated Bases

7.2 Location, Scale, and Shape Estimates

Given a time series y_t, models are estimated using GAMLSS penalized likelihood as previously detailed, with location parameters consisting of penalized linear B-splines per (Eilers and Marx 1996) for modeling trend, penalized cubic cyclic splines applied to hour of day and day of week regressors, and regressors for pulses. Scale and shape parameters include intercepts without any regressors, and B-splines applied to time. Paths parameterized by $\boldsymbol{\eta}_y := (m_y, s_y, \nu_y)$ are given as follows:

Location: $m_y(t) = \beta_y^0 + f_{trend}(t) + f_{c1}(x_{t,dow}) + f_{c2}(x_{t,hour}) + \sum_{k=1}^{k=N} \delta_k \mathbf{1}_{t_k}(t)$; where $x_{t,dow} \in \{1, 2, ..., 7\}$, $x_{t,hour} \in \{0, 1, ..., 23\}$, and pulses δ_k occurring at unknown t_k, for $k = 1, ..., N$.

Scale and ***Shape***:

$s_y(t), \nu_y(t) = \beta_y^0 + f_{trend}(t)$, where $f_{trend} = \sum c_i B_{i,n}(t)$, $B_{i,1}(t) = 1$ if $t \in [t_i, t_{i+1}]$, and 0 otherwise, and $B_{i,k+1}(t) = \frac{t-t_i}{t_{i+k}-t_i} B_{i,k}(t) + \frac{t_{i+k+1}-t}{t_{i+k+1}-t_{i+1}} B_{i+1,k}(t)$ given knots $t_0, t_1, \ldots, t_K$.

7.3 Generating Path Anomalies

Real anomalous time series are rare and can contain features that have most of the characteristics of the normal time series but with subtle differences or unusual variations, such as periodicity being masked by an extreme temporary shift. The approach taken to simulate path anomalies was to randomly sample from multiple time series with random walk or downward shift for location, linear increasing scale, or shape parameter between -1 and -0.5, and combine their values to form new composite time series. This yielded paths that do not probabilistically belong to any of the defined subspaces, and exaggerates the characteristics taken from the normal but less frequent paths. A sample size of anomalies tested was 1, 5, and 10 per simulation. See Figure 2 for a plot of the time series with path anomalies.

7.4 Performance Measure

The primary measure used to evaluate performance of detecting anomalies is the F score, a weighted average of precision and recall defined by 2·(Precision·Recall)/(Precision+Recall),

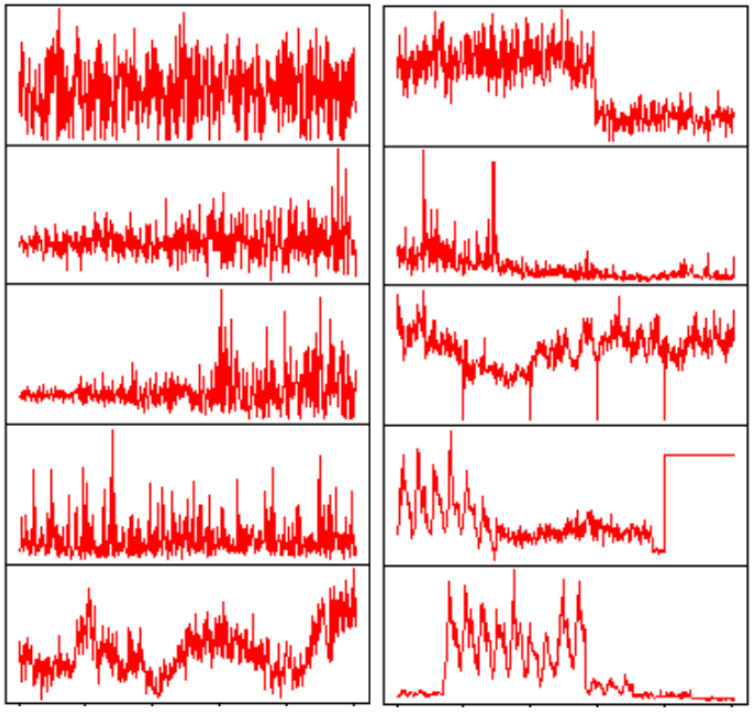

Figure 2: Simulated Path Anomalies

where Precision $= TP/(TP + FP)$, and Recall $= TP/(TP + FN)$. Both precision and recall values are computed over the top 10 ranked unusual time series.

Additionally, a measure of excess rank is introduced. If p is the real frequency of anomalies, and N is the number of time series, let R_N be the position of each anomaly out of N found by sorting the estimated likelihood scores in decreasing order. Then the excess rank is the average difference between frequency and max ranked proportion, i.e. $E[p - max(R_N)/N]$. If likelihood scores are properly calibrated then the excess rank should be zero. Note that a weighted average excess rank could also be computed if a measure of rarity of each anomaly was available, but for this simulation each are viewed as equally likely.

7.5 Results

Normal sample paths were simulated from BCCG processes with constant scale and shape, split into low value ranges (between 0.05 and 0.1 for scale and between -0.3 and 0.2 for

shape) and mid value ranges (0.1 to 0.2 for scale, -0.5 to -0.25 for shape). Location parameters were generated from a mix of basis function classes, including constant level, multiple seasonality, stationary local level with random pulses, AR(p), and linear upward shifts.

Six different experiments were run, each generating multiple replications of 200 series during each run, randomly sampling from both the different bases and anomalous series, per each frequency of anomalies tested.

Experiment 1 consisted solely of paths from a single model family generated from multiple seasonality bases with low and high constant values for scale and shape. It was designed to establish a baseline for relatively simple series.

Experiment 2 consisted of 4 distinct model families from the multiple seasonality, local level random pulses, AR(p), and linear upward shift bases, with 6 different subspaces. This produces a fairly complex mix of time series with different characteristics.

Experiment 3 was set up to assess how the methods would perform in the presence of a much larger frequency of anomalies. In addition to the 10 defined anomalies, 10 other anomalies were sampled from a basis consisting of 3 change points and a temporary upward shift. A constant location, with low scale and shape parameters from a BCCG distribution formed the normal series.

Experiment 4 was designed to test that GAWS with a BCCG distribution family but generic basis function class consisting only of penalized B-splines for all parameters would not overfit. Again a constant location, scale and shape model family generated the normal series.

Experiment 5 is similar to 4, but includes all model families from experiment 2 in the generation of the normal series, as well as an additional local level.

Finally, experiment 6 was added to show that GAWS could yield very poor results if the wrong model families are entertained. Random walks bases from Gaussian and Student t distribution families were considered, with again a constant location, scale and shape model family forming the normal time series.

A total of 15 distinct model families were considered to configure the GAWS algorithm for the main three experiments E1,E2,E3. Fitting models for each series included com-

binations of local level, double seasonality, random pulse, AR(p), random walks, linear shifts, and penalized B-spline trends, using the Gamma and BCCG distribution families. Additional spurious models were purposely included to evaluate the robustness of Akaike weight scoring.

Note that time series were re-scaled to have the same mean. This was done because the focus is not on quantifying magnitude differences but rather understanding which relative paths are unusual.

Table 2 summarizes the overall average performance for each of the experiments ran. Note that the relative F-score is defined as F-score + correction, where correction:$= 1 - 2 \cdot \frac{count(anomaly)/rth)}{1+count(anomaly)/rth}$, where rth is the maximum rank set to 10 here. The correction is added to account for the maximum possible F-score that can be attained only selecting the top 10. Thus, if there is only 1 anomaly but it is ranked in the top 10 then the relative f-score is 1, instead of an f-score of 0.182.

				Relative F-Score			Abs Excess Rank		
ExperimentName	#Series	#Subspace	#Anomaly	gaws	hdr	stray	gaws	hdr	stray
E1: Rank Anomaly|Seasonality	200	2	1	1.0000	0.9982	1.0000	0.0000	0.0080	0.0000
			5	1.0000	0.9133	1.0000	0.0000	0.1519	0.0150
			10	1.0000	0.8100	1.0000	0.0000	0.1265	0.0000
E2: Rank Anomaly|Seasonality, Pulses, AR, Shift	200	6	1	1.0000	0.9091	0.9618	0.0000	0.1844	0.0197
			5	1.0000	0.6667	0.7967	0.0000	0.4414	0.0445
			10	1.0000	0.6000	0.6433	0.0000	0.5043	0.0833
E3: Rank Anomaly|Constant and 10 distinct/10 similar temp-shift anomalies	200	1	20	0.9500	0.7500	0.8000	0.0050	0.6550	0.1200
E4: Rank Anomaly|Constant and GAWS|penalized splines	200	1	10	1.0000	0.8000	1.0000	0.0000	0.6500	0.0000
E5: Rank Anomaly|Level, Seasonality, Pulses, AR, Shift, and GAWS|penalized splines	200	8	10	0.9474	0.4667	0.5260	0.0050	0.6553	0.2444
E6: Rank Anomaly|Constant and misspecified GAWS|RW NO/TF family	200	1	10	**0.0000**	0.8000	1.0000	**0.9000**	0.6500	0.0000

Table 2: Average Performance per Experiment and Anomaly Frequency

In summary, across the 5 main experiments E1-E5, GAWS performance often exceeds and is never worse than both the stray and hdr algorithms. GAWS is much better in E2 for 5 and 10 anomalies, as well as in E5. GAWS yielded an overall mean relative F-score of 0.987, a 0.13 improvement compared to stray and 0.22 over hdr. In terms of the mean absolute excess rank metric, GAWS had a mean of 0.001, versus stray at 0.0585, and hdr at 0.375. In particular, GAWS had perfect ranking except in experiment 5, where it only took 1 additional rank to catch all anomalies.

Perhaps it is not surprising that GAWS yields such nearly perfect results given the model space constructions are representative of the data, with each model family having adequate samples for GAMLSS to learn basis functions.

As a side note worth calling out, stray is always better than the hdr method, with superior results for the excess rank metric. It does appear that performance for both the hdr and stray algorithms do degrade as the number of anomalies increases in the data. Further experiments would need to be ran to better understand if this is only specific to the type of anomalies and default extracted features considered here.

On the other extreme, GAWS yields inferior results in experiment 6, though this is purely contrived. The GAWS algorithm by default uses the more flexible penalized spline bases, which was shown to yield competitive performance on this simulated data. However, the takeaway from E6 should be that using poor choices of bases can render GAWS useless, which of course is not only the case for this algorithm but any that rely on sensible model or feature choices.

8 Application: Anomalous Cloud Traffic

Monitoring traffic for cloud products across multiple data centers and software applications is an important task. As engineers develop and ship new features, update virtual machine configurations, and attempt to improve the cloud experience for customers, inadvertently bugs and other issues can arise and impact performance. Furthermore, because of maintenance, or capacity shortages, traffic can get redirected from the original intended data center to another where excess capacity is available, thus adding noise to the data.

There are too many complex time series for subject matter experts to manually visualize and identify what to investigate, so an automated solution that ranks the most unusual series is useful. Thus, an analysis was initiated to understand how well GAWS could identify previously known anomalous series.

A data set consisting of 138 per 30-minute time series with 960 distinct time points reporting the total number of sessions across multiple data centers and different software applications was considered. There were 4 known odd series flagged, and engineers were interested in evaluating which of these could be captured by an anomaly detector looking at the top 10 ranking. An assessment was carried out to comparing rankings from GAWS and stray.

To use the GAWS algorithm, a collection of basis functions and distribution families need to be specified. Performing some exploratory data analysis, and looking at plots for a sample of time series revealed clear multiple and changing seasonality, changing level/slow trend, infrequent spikes, and variability much higher during peak times, see Figure 3. Note that due to Microsoft confidentiality, all references to product information and times are removed in the plots, and actual time series values are transformed while preserving non-negativity and perturbed by adding some Gaussian noise.

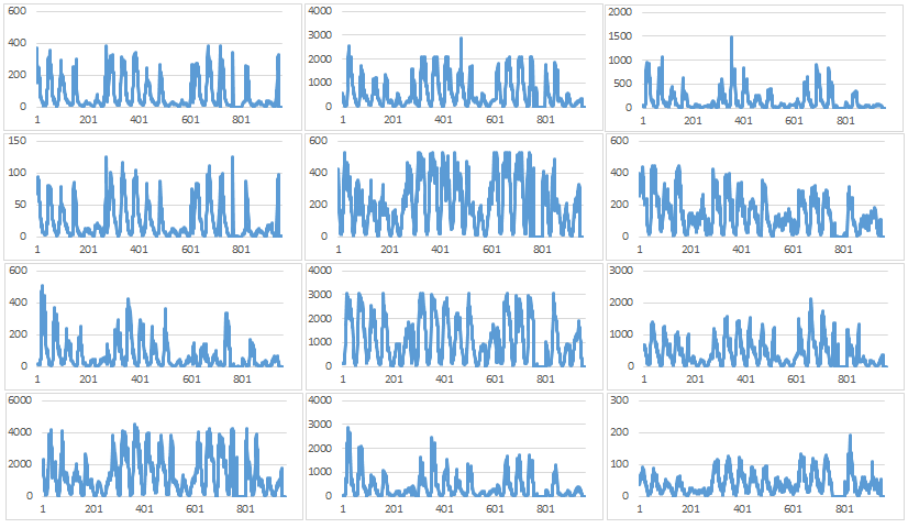

Figure 3: Sample Cloud Traffic Time Series 30-Minute Intervals

Given the time series were positive skewed large volume counts, several continuous distribution families were entertained including log-normal, log-t, log-skewed-t, gamma, and BCCG with log link. Fitting initial GAMLSS models to the sample of time series with these distribution families was carried out using the following basis functions: for the location parameter, fourier expansions were used to represent seasonality, penalized b-spline of degree 1 captured any changing local level/trend, and binary indicators included to handle significant random pulses. For the scale parameter, penalized splines dependent on seasonality and level basis were implemented. For the shape parameter, a constant basis as well as spline conditioned on the level of the series were considered.

A random sample of 20 series confirmed that the model specification with log-t family was reasonable. Figure 4 depicts random fluctuations around a center line with a few points

bordering or outside the 95 percent confidence band of the worm plot, that is, a detrended Q-Q plot as described in (Buuren and Fredriks, 2001) applied to the quantile residuals, see (Dunn, 1996).

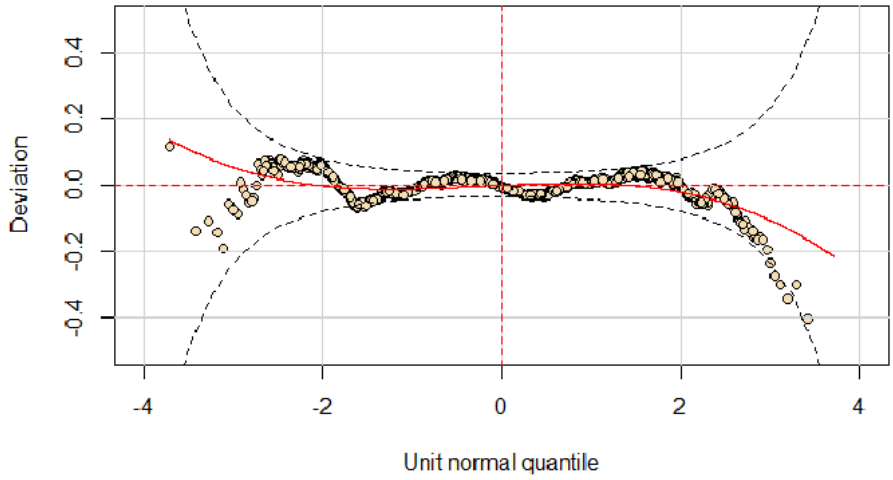

Figure 4: Log-t Distribution Family Worm Plot Quantile Residuals

Figure 5 visualizes how the scale changes as a function of the relative peak season, and how shape changes given level. Scale is nonlinearly increasing as a function of season, whereas shape jumps upward and climbs as the level increases. Modeling of conditional moments in terms of basis functions is easily done leveraging GAMLSS, and is integral to how the GAWS algorithm can detect many types of unusual paths.

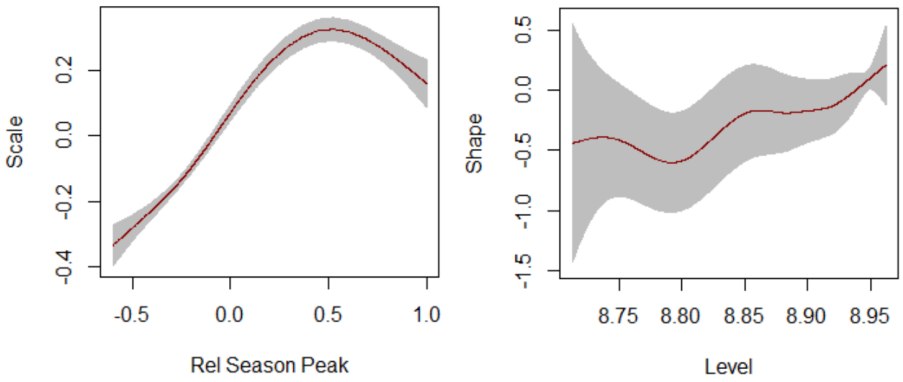

Figure 5: Sample Cloud Traffic Scale and Shape Basis Functions per Time

With the basis functions defined above, both GAWS and stray were run and their

ranking output is compared in Table 3. The results displayed are based on the BIC, though AIC was also tested and yielded the same ranking on this data. GAWS was able to detect all 4 anomalies in the top 4 ranking as indicated by bold red, with one FP for series identifier 130. The stray algorithm was only able to catch two anomalies in the top 5 ranking, and required going to rank 18 to catch all 4. It is worth noting that stray detected series identifiers 127, 128, 129 as the most anomalous, which were from a specific data center with slightly different seasonal variation that had less noise but otherwise were not remarkable. Perhaps with custom feature engineering the stray algorithm could have caught all 4 anomalies, though the key takeaway is that GAWS achieved perfect results using a fairly generic class of basis functions.

GAWS Method		
seriesID	Rank	Score
139	1	4E-76
140	2	1E-29
141	3	9E-28
142	4	1E-17
130	**5**	**2E-11**

STRAY Method		
seriesID	Rank	Score
128	1	1.12729
127	2	0.65633
129	3	0.65633
141	4	0.14415
140	5	0.07168
142	8	0.0619
139	18	0.03914

Table 3: Comparison of Ranking of Cloud Traffic Time Series, with Anomalies in red

9 Discussion

In this paper, the problem of detecting relatively anomalous time series among collections was framed and probabilistically quantified. Viewing a time series as a sample path from an unknown random function, a relative likelihood score is applied to establish a measure of rarity. Namely, a path is identified as anomalous if it is implausible to be represented by models belonging to a sufficiently large class comprised of well-specified probability distribution families and basis functions that capture the typical data generating process. From this perspective, anomalous series have path representations that belong to an alternative, low-dimensional and well-separated subspace.

A fairly general statistical framework was then formalized for detecting path anomalies based on model embeddings and Akaike weights following a similar philosophy as discussed in (Viele 2001). A mathematical justification was also presented, explaining why the proposed Akaike weight scores are at least asymptotically unbiased referencing previous known results connecting KL divergence type measures and information criterion.

The GAWS algorithm was then introduced as a data-driven solution for ranking and classifying anomalous series, utilizing generalized additive models for location, scale, and shape to produce flexible model embeddings, and computing Akaike weight scores via penalized likelihood. GAWS provides interpretable inference and can be implemented in a scalable way.

In addition to the supporting asymptotic theory guaranteeing that the Akaike weight scores will converge to a point mass distribution centered at zero for time series with path anomalies under suitable regularity conditions and proper choice of model families, it was also empirically demonstrated that the proposed GAWS method can yield excellent accuracy in detecting anomalies among complex classes of series. Specifically, looking at both multiple simulations and two real datasets, GAWS achieved very high precision and recall, and always yielded as good and often much better results than other methods, including the recently proposed stray algorithm in (Talagala et al. 2019). However, it is acknowledged that no general claims can be made regarding the performance of GAWS across all possible situations, and certainly further studies are warranted to continue assessing the strengths and limitations of GAWS given particular time series.

While GAWS provides a powerful toolbox for detection of anomalous time series, it does have some known limitations. The penalized likelihood formulation, while computationally appealing, is restricted to models where the effective degrees of freedom is known. Moreover, there needs to be ample representative data to obtain trustworthy likelihood estimates. It is expected that GAWS would not perform as well dealing with collections of only short time series, where majority are both highly sparse and heterogenous.

Like other feature-based methods, GAWS is only useful if a reasonable set of model families are entertained. Utilizing generic bases such as penalized splines to construct an

embedding can work quite well given enough data as was demonstrated in the simulation study, but specifying particular location, scale and shape parameters is likely needed to help reduce both false positives and false negatives in practice.

Although GAWS was designed for the purpose of anomaly detection, an extension to functional clustering seems plausible, where the location, scale and shape basis expansions together with an information criterion could be utilized to form fuzzy clusters and hence provide an alternative way to generate mixture models via the Akaike weights. Further work is required to assess the feasibility of this capability.

Additional research is also needed to extend the GAWS framework before it can be useful in certain settings. For example, enhancements would need to be made for GAWS to be suitable for streaming data, where both the model space and parameters are dynamically updated. Another important consideration is automatic pooling to learn parameters across groups of series given sparse data, and improve estimation and inference working with hierarchical or multivariate time series. Integrating and evaluating an alternative estimation procedure, such as boosting for datasets requiring specification of many basis function classes, would be worthwhile for both computational speed and practical construction. Incorporating mixtures of model families for complex datasets where unimodal parametric distributions are insufficient would make the GAWS algorithm even more extensible. On a final note, it would be interesting to re-frame GAWS within a Bayesian framework, putting a prior on the choice of model space and computing scores via Bayesian model averaging.

SUPPLEMENTAL MATERIALS

All simulations, data and code have been made publicly available in a zipped file that can be found at the link below, with an accompanying readme.txt describing how to run all scripts.
https : *//github.com/colesodja/GAWS_LOCAL*

Data: The saved time series simulations, pre-processed hourly pedestrian counts and transformed anonymized cloud traffic data are available as dataframes under a folder called data.

Simulations: If there is interest in generating additional sample paths then all the code to do so is located under a folder called sim.

Scripts: R code to re-produce results running the algorithms on saved datasets exists under the folder named scripts.

Source Code: R functions for all GAMLSS models, penalized likelihood Akaike weight calculations, and the local GAWS algorithm is stored under the folder called src. Note that there are dependencies on other packages that must be installed. A script **1.install.packages.r** is included to check and if missing install all necessary packages.

References

[Abraham et al. 2003] Abraham, Cornillon, Matzner-Løber, Molinari (2003). Unsupervised Curve Clustering using B-Splines *Scandinavian Journal of Statistics* 30(3), 581–595.

[Adam et al. 2017] Adam, Mayr, Kneib, (2017). Gradient boosting in Markov-switching generalized additive models for location, scale and shape arXiv, 1710.02385

[Akaike 1973] Akaike (1973). Information theory and extension of the maximum likelihood principle *2nd International Symposium on Information Theory*, 267–281.

[Akaike 1981] Akaike (1981). Likelihood of a model and information criteria *Journal of Econometrics*, 16:1, 3–14.

[Beggel et al. 2018] Beggel, Kausler, Schiegg, Pfeiffer, Bischl (2018). Time series anomaly detection based on shapelet learning. Computational Statistics, Springer Berlin Heidelberg.

[Berk 1966] Berk (1966). Limiting behavior of posterior distributions when the model is incorrect. The Annals of Mathematical Statistics, 37:1, 51-58.

[Berlinet and Agnan 2004] Berlinet, Agnan (2004). Reproducing Kernel Hilbert Spaces in Probability and Statistics. Springer Science + Business Media, LLC.

[Brockhaus et al. 2017] Brockhaus, Fuest, Mayr, Greven (2017). Signal regression models for location, scale and shape with an application to stock returns. Journal of the Royal Statistical Society, Applied Statistics Series C, 67:3, 665–686.

[Buja et al. 1989] Buja, Hastie, Tibshirani (1989). Linear Smoothers and Additive Models. The Annals of Statistics, 17:2, 453-510.

[Burnham and Anderson 2002] Burnham, Anderson (2002). Model Selection and Multimodel Inference: A Practical Information-Theoretic Approach. 2nd Edition, Springer-Verlag, New York.

[Buuren and Fredriks 2001] Buuren, Fredriks (2001). Worm plot: A simple diagnostic device for modelling growth reference curves. Statistics in Medicine, 20:8, 1259–1277.

[Cole and Green 1992] Cole, Green (1992). Smoothing reference centile curves: the LMS method and penalized likelihood. Statistics in Medicine 11:10, 1305-1319.

[Dunn and Smyth 1996] Dunn, Smyth (1996). Randomized Quantile Residuals. Journal of Computational and Graphical Statistics, 5:3, 236-244.

[Eilers and Marx 1996] Eilers, Marx (1996). Flexible smoothing with B-splines and penalties. Statistical Science, 11:2, 89-102.

[Gil and Romo 2014] Gil, Romo, (2014). Shape outlier detection and visualization for functional data: the outliergram. Biostatistics, 15:4 603-619.

[Gould et al. 2008] Gould, Koehler, Ord, Snyder, Hyndman, Araghie (2008). Forecasting time series with multiple seasonal patterns. European Journal of Operational Research, 191:1, 207-222.

[Gupta et al. 2014] Gupta, Gao, Aggarwal, Han (2014). Outlier detection for temporal data: A survey *IEEE Transactions on Knowledge and Data Engineering* **32**, 26(9), 2250–2267.

[Hyndman and Shang 2010] Hyndman, Shang, (2010). Rainbow Plots, Bagplots, and Boxplots for Functional Data. Journal of Computational and Graphical Statistics, 19:1, 29-45.

[Hyndman et al. 2015] Hyndman, Wang, Laptev, (2015). Large-Scale Unusual Time Series Detection. IEEE International Conference on Data Mining Workshop.

[Hyndman et al. 2018] Hyndman, Wang, Laptev (2018). Anomalous time-series R Package. R package version 0.1.0. https://github.com/robjhyndman/anomalous

[Kolassa 2011] Kolassa, (2011). Combining exponential smoothing forecasts using Akaike weights. International Journal of Forecasting 27, 238–251.

[Konishi and Kitagawa 1996] Konishi, Kitagawa (1996). Generalised Information Criteria in Model Selection. Biometrika, 83, 875–890.

[Kokoszka and Reimherr 2017] Kokoszka, Reimherr (2017). Introduction to Functional Data Analysis. CRC Press.

[Mayr et al. 2012] Mayr, Fenske, Hofner, Kneib, Schmid, (2012). Generalized additive models for location, scale and shape for high dimensional data - a flexible approach based on boosting. Journal of the Royal Statistical Society, Series C, 61, 403–427.

[Pimentel et al. 2013] Pimentel, Clifton, Tarassenko, (2013). Gaussian process clustering for the functional characterisation of vital-sign trajectories. IEEE International Workshop on Machine Learning for Signal Processing (MLSP).

[Ramsay and Silverman 2005] Ramsay, Silverman (2005). Functional Data Analysis. Springer Series in Statistics.

[Rao and Wu 1989] Rao, Wu, (1989). A strongly consistent procedure for model selection in a regression problem. Biometrika 76, 369-374.

[Rigby and Stasinopoulos 1996] Rigby, Stasinopoulos, (1996). Mean and Dispersion Additive Models. Computational Aspects of Smoothing, Contributions to Statistics.

[Rigby and Stasinopoulos 2005] Rigby, Stasinopoulos, (2005). Generalized additive models for location, scale and shape (with discussion). Journal of the Royal Statistical Society, Applied Statistics Series C, 54:3, 507-554.

[Rigby et al. 2017] Rigby, Stasinopoulos, Heller, Voudouris, De Bastiani, (2017). Flexible Regression and Smoothing: Using GAMLSS in R. New York: Chapman and Hall/CRC.

[Rigby et al. 2020] Rigby, Stasinopoulos, Heller, De Bastiani, (2020). Distributions for Modeling Location, Scale, and Shape: Using GAMLSS in R. New York: Chapman and Hall/CRC.

[R Core Team 2020] R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/

[Salvador and Chan 2005] Salvador, Chan (2005). Learning states and rules for detecting anomalies in time series. Appl Intel 23:3, 241–255.

[Schmutz et al. 2018] Schmutz, Jacques, Bouveyron, Cheze, Martin, (2018). Clustering multivariate functional data in group-specific functional subspaces. hal-01652467v2.

[Schlosser et al. 2019] Schlosse, Hothorn, Stauffer, Zeileis (2019). Distributional Regression Forests for Probabilistic Precipitation Forecasting in Complex Terrain. Annals of Applied Statistics, 13(3), 1564–1589

[Shi and Choi 2011] Shi, Choi, (2011). Gaussian Process Regression Analysis for Functional Data. Chapman & Hall/CRC.

[Stasinopoulos et al. 2020] Stasinopoulos, Rigby, Voudouris, Akantziliotou, Enea, Kiose (2020). gamlss: Generalised Additive Models for Location Scale and Shape. R package version 5.1-7. https://cran.r-project.org/web/packages/gamlss/index.html

[Talagala et al. 2019] Talagala, Hyndman, Miles, Kandanaarachchi,Muñoz (2019). Anomaly Detection in Streaming Nonstationary Temporal Data. Journal of Computational and Graphical Statistics, 1-21.

[Talagala et al. 2020] Talagala, Hyndman, Miles (2020). Anomaly Detection in High Dimensional Data. Journal of Computational and Graphical Statistics, 1-32.

[Talagala et al. 2020] Talagala, Hyndman, Miles (2020). stray: Anomaly Detection in High Dimensional and Temporal Data. R package version 0.1.1. https://cran.r-project.org/web/packages/stray/index.html

[Tzeng et al. 2018] Tzeng, Hennig, Li, Lin, (2018). Dissimilarity for functional data clustering based on smoothing parameter commutation. Statistical Methods in Medical Research, 27(11), 3492–350.

[Twomey et al. 2019] Twomey, Chen, Diethe, Flach (2019). Anomaly detection in star light curves using hierarchical Gaussian processes. Proceedings of the Twenty-sixth European Symposium on Artificial Neural Networks, 615–620.

[Umlauf et al. 2017] Umlauf, Klein, Zeileis, (2017). BAMLSS: Bayesian Additive Models for Location, Scale, and Shape (and Beyond). Journal of Computational and Graphical Statistics, 27:3, 612-627.

[Viele 2001] Viele (2001). Evaluating Fit in Functional Data Analysis Using Model Embeddings. The Canadian Journal of Statistics / La Revue Canadienne De Statistique, 29:1, 51–66.