2015-12

Load-Adjusted Video Quality Prediction Methods for Missing Data

Ruairí de Fréin
Technological University Dublin, ruairi.defrein@tudublin.ie

Recommended Citation

De Fréin, D. (2015) Load-adjusted video quality prediction methods for missing data, *10th International Conference for Internet Technology and Secured Transactions (ICITST), London, 2015*. doi:10.1109/ICITST.2015.7412111

R. de Fréin, "Load-adjusted video quality prediction methods for missing data," 2015 10th International Conference for Internet Technology and Secured Transactions (ICITST), London, 2015, pp. 314-319.
doi: 10.1109/ICITST.2015.7412111
keywords: {client-server systems;polynomials;quality control;statistical analysis;video on demand;video signal processing;PQ-model;RTP packet rate;client-server system;load-adjusted video quality prediction methods;missing data;network planning;parametric statistical model;polynomial fitting model;robustness;video-on-demand;Kernel;Load modeling;Measurement;Predictive models;Quality assessment;Servers;Video recording;Clouds;Network Analytics;Video-on-Demand},
URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7412111&isnumber=7412034

Load-Adjusted Video Quality Prediction Methods for Missing Data

Ruairí de Fréin† ††
†KTH - Royal Institute of Technology, Stockholm, Sweden
††Waterford Institute of Technology, Ireland
web: https://robustandscalable.wordpress.com

in: 10th International Conference for Internet Technology and Secured Transactions (ICITST-2015), to appear. See also BibTeX entry below.

BibTeX:

```
@article{rdefrein15Load,
author={Ruair\'{i} de Fr\'{e}in$^\dagger$ $^{\dagger\dagger}$},
journal={10th International Conference for Internet Technology and Secured Transactions
(ICITST-2015), to appear},
title={Load-Adjusted Video Quality Prediction Methods for Missing Data},
year={2015},
pages={1--6},
month={Dec.},
note={London, UK},}
```


document created on: December 9, 2015
created from file: rdefreinICITST15_preprint.tex
cover page automatically created with CoverPage.sty
(available at your favourite CTAN mirror)

Load-Adjusted Video Quality Prediction Methods for Missing Data

Ruairí de Fréin
KTH Royal Institute of Technology, Sweden: rdefrein@gmail.com

***Abstract*—A polynomial fitting model for predicting the RTP packet rate of Video-on-Demand received by a client is presented. The approach is underpinned by a parametric statistical model for the client-server system, namely the *PQ*-model. It improves the robustness of the predictor in the presence of a time-varying load on the server. The advantage of our approach is that (1) if we model the load on the server, we can then use this model to improve RTP packet rate predictions; (2) we can predict how the server will behave under previously unobserved loads –a tool which is particularly useful for network planning; and finally (3) the PQ-model provides accurate predictions of future RTP packet rates in scenarios where training data is unavailable.**

***Keywords*—Video-on-Demand, Clouds, Network Analytics.**

I. Introduction

The authors of [1] examined whether a client's service level metrics for Video-on-Demand could be accurately predicted using metrics captured from the kernel of the server servicing the requests. The authors of [2] demonstrated that load-adjusted learning improved the predictor in [1] but did not consider prediction when training data was missing. In this paper we characterize the behaviour of the server irrespective of the number of active user requests on the server. We show that using a system characterization step allows us to improve the prediction estimates. The application of Statistical Learning (SL) for prediction in cloud and network environments is at an early stage. Monitoring and predicting performance metrics for clouds services is a challenging, open problem [3]. Running software systems on general purpose platforms without real-time guarantees, with the expectation that one can safeguard revenues, is dichotomous. The video service level prediction work of [1] is a timely contribution given that Cisco [4] predicts that network traffic volumes in the order of tens of exabytes are not that far off, and 90% will be video related [5]. According to [1] a SL approach, for example Matrix Factorization [6] or Formal Concept Analysis [7], is preferable to developing and fitting complex analytical models for the different layers of soft/hardware in these complex systems. In terms of the practicality of this type of approach, the authors of [8] make the case that modern multi-core (parallel online) learning algorithms are limited by the bandwidth bottleneck, and thus, overly complex SL algorithms may not be suited for real-time cloud services. In terms of related approaches, a method for identifying and ranking servers with problematic behavior is proposed in [9]. The authors use Random Forest classifiers to select candidate servers for modernization. A predictive model is then used to determine the impact of modernization actions. A Support Vector Regression predictor is used in [10] to perform lightweight TCP throughput prediction. Prediction is based on prior file transfer history and measurements of path properties. A method for modeling application servers in order to detect performance degradation due to ageing is presented by [11]. The authors use classification algorithms to perform proactive detection of performance degradation. Finally, the authors attempt to reduce the size of the data-stream that is forwarded to an operations support system by removing uncorrelated noise events in [12]. A heuristic cross-correlation function determines the degree of inter-relationship between the events in the data-stream. Little work has been done on dealing with the effects of adaptive loads on these systems, in particular, on constructing load-aware models with missing data as part of a prediction systems. One of the first approaches by Zhang et al., showed that Tree-Augmented Bayesian networks provide an effective approach for identifying which low-level system properties are correlated with high-level servlice level objective violations in [13] in the presence of changing workloads.

Set-up: In Fig. 1 server resources are shared between multiple clients. The video server (LHS box) delivers video to the target client machine (RHS Upper); however, a number of other clients also use the server's resources. They are represented by the load generator (RHS Lower box). The number of clients using the video server changes with time, which makes predicting the target client's video quality challenging. We desire a model that allows us to characterize the load's effect on the client's video quality if we have knowledge of the kernel metrics of the server delivering the video. Then, irrespective of the number of users on the system, we would also like to be able to predict when the client's video quality will deviate from its ideal performance. In order to match-up the server and client observations, the clocks of the server and client are synchronized using NTP[1a]. Samples from the server and the client's machine are collected every second. The client's service level metric, the RTP packet rate, y_i at time i is captured using VLC media player[1c], which provides Video-on-Demand requests in [1]. Features refer to metrics on the operating system level for example, the number of active TCP connections on the server. The feature set $\mathbf{x}_i[n]$ is constructed using the System Activity Report[1b] (SAR) on the server.

Contributions: (1) A parametric statistical model is contributed, which has sufficient flexibility to describe the behaviour of the server's kernel metrics. This model explicitly accounts for the load on the system (when it does or does not

Dr. de Fréin is also affiliated with TSSG, Waterford Institute of Technology, Ireland. This work was supported by an ELEVATE Irish Research Council International Career Development Fellowship co-funded by Marie Cure Actions award "EOLAS": ELEVATEPD/2014/62.

[1a] http://www.ntp.org; [1b] http://linux.die.net/man/1/sar;
[1c] http://www.videolan.org/vlc

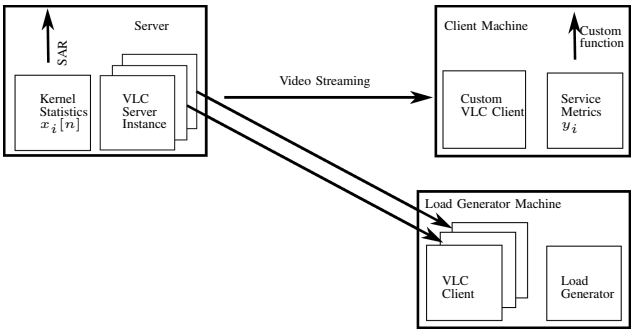

Figure 1. Prediction Scenario: A target client and multiple other clients (represented by the load-generating machine) share the video server's resources.

affect a given kernel metric). In addition, a statistical generative model, which captures the dependence between the server's behaviour and the client's service level metric, is contributed. (2) A polynomial system characterization technique, namely the PQ-model, is proposed. This model accurately models the expected behaviour of the server/client machine under different load conditions, and also, the expected volatility in their behaviour. (3) A prediction model, which incorporates system characterization into the prediction model, is contributed. In Section II we propose a generative model for the observations: the server and client service metrics. We develop a system characterization model in Section III which describes how the server and client's service level metrics behave under different load conditions. We extend this characterization into a prediction model for the client's service level metric. We evaluate these models empirically in Section IV.

II. Parametric Statistical Model

When a client requests Video-on-Demand (VoD), the response of the server, with respect to kernel metric n, *the n-th feature*, to this request for video at time i is expressed as:

$$\mathbf{x}_i[n] = \hat{\mathbf{u}}_i[n] + \boldsymbol{\varepsilon}_i[n], \qquad \text{where } i \in \mathbb{Z}, \mathbf{x}_i[n], \hat{\mathbf{u}}_i[n] \in \Re. \quad (1)$$

The first term, $\hat{\mathbf{u}}_i[n]$ in (Eqn. 1), an indicator function, represents the increase in the usage of the n-th feature for the duration of the client's session. The second term $\boldsymbol{\varepsilon}_i[n]$ captures the non-ideal behaviour of the server's n-th feature. It is this type of error that we would like to be able to predict. When multiple users request VoD the expected usage of the n-th resource is represented by the load signal $l_i[n] = \alpha_n K(i)$. The nonnegative scalar α_n is the usage of the n-th feature by one user and $K(i)$ denotes the number of active user requests at time i. The response of the n-th feature to this load is

$$\mathbf{x}_i[n] = l_i[n] + \sum_{k=1}^{K(i)} \boldsymbol{\varepsilon}_i[n,k] = l_i[n] + \hat{\mathbf{x}}_i[n]. \quad (2)$$

Here the term $\hat{\mathbf{x}}_i[n] = \sum_{k=1}^{K(i)} \boldsymbol{\varepsilon}_i[n,k]$ represents the sum of all of the non-idealities, due to all of the active user sessions on the server at time i, e.g. $K(i)$. The service level metric, the RTP packet rate in this paper, measured by the client of interest is modelled by the linear expression

$$y_i = \sum_n \mathbf{w}[n]\left(l_i[n] + \hat{\mathbf{x}}_i[n]\right) = \alpha_y K(i) + \hat{y}_i. \quad (3)$$

This expression is composed of the sum of a load term $\alpha_y K(i)$, where α_y represents the affect of one user request on the RTP packet rate and $K(i)$ is the load trace. In addition, $\hat{y}_i$ is the aggregate deviations signal, which represents departures from the ideal performance of the system measured by the client. We desire a system that predicts, $\hat{\mathbf{y}}_i$. To do this we need to extract the deviations from the ideal performance component signal, e.g. $\hat{\mathbf{x}}_i[n]$ and $\hat{\mathbf{y}}$, from the load component $l_i[n]$, and predict how the deviations (alone) affect the RTP packet rate. In summary, when the system is behaving in an ideal manner the RTP packet rate at time i, e.g. y_i, is modelled as a weighted sum of the expected usage of each of the server's resources, $y_i = \sum_n \mathbf{w}[n] l_i[n]$. There is no uncertainty in how the server will behave, $\hat{y}_i = 0, \forall i$. All that is required to obtain good estimates of the RTP packet rate, is to measure the usage of each resource type for a given number of co-incidental user requests using training data, and then to read-off what that value should be from a look-up table during the prediction step. If the system is not ideal, some of the server features experience deviations from their ideal behaviour, $\hat{\mathbf{x}}_i[n]$. The goal of this paper is to predict how these deviation signals, $\hat{\mathbf{x}}_i[n]$, affect the RTP packet rate.

III. System Characterization: PQ-model

We present a polynomial statistical system characterization model. It allows us to answer the questions: (**q1**) "What is the expected effect of k active users on the RTP packet rate of the observed client?"; (**q2**) "What types of fluctuations should we expect in the RTP packet rate for a given load on the system?"; (**q3**) "If we have not observed the RTP packet rate when the system is under the load k can we estimate what is would be?" and finally, (**q4**) "What is RTP packet rate be at time $i+1$ if we observe the kernel features at time $i+1$?" We evaluate the efficacy of the model in the next section using the traces generated in [1]. To characterize the system, using a real-trace as training data, we generate the set of points corresponding to a load of $l_i = \alpha_y k$ for all values of k in the training data. We denote this set $\mathcal{H}(y)|_k$. We model the characteristic behaviour of the server by fitting a P-th degree polynomial through the observations $\mathcal{H}(y)|_k$ as a function of k, e.g. the number of active users on the system.

$$y = \sum_{p=0}^{P} a_p k^p, \text{ and } R_a^2 = \sum_{i=1} \left[y_i - \sum_{p=0}^{P} a_p K(i)^p \right]^2. \quad (4)$$

The residual of this approximation is given by R_a^2 and the notation $K(i)$ indicates the number of active user requests at time i. The trace $K(i)$ is obtained from the TCP socket count of the server. The set of parameters $\theta_y = \{a_0, \ldots a_P\}$ that minimize the residual are obtained by deriving the partial derivatives of (Eqn. 4) and setting them to zero,

$$\frac{\partial R_a^2}{\partial a_m} = -\sum_i \left[y_i - \sum_{p=0}^{P} a_p K(i)^p \right] K(i)^m. \quad (5)$$

We reformulate these systems of equations in matrix form –for notational convenience– and then solve for $\mathbf{a}$.

$$\mathbf{M} = \begin{pmatrix} 1 & K(1) & \ldots & K^P(1) \\ 1 & K(2) & \ldots & K^P(2) \\ \vdots & \vdots & \ddots & \vdots \\ 1 & K(T) & \ldots & K^P(T) \end{pmatrix}, \mathbf{a} = \begin{pmatrix} a_o \\ a_1 \\ \ldots \\ a_P \end{pmatrix}, \mathbf{y} = \begin{pmatrix} y_1 \\ \vdots \\ y_T \end{pmatrix} \quad (6)$$

We obtain the parameters in the set θ_y by computing

$$\mathbf{a} = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\mathbf{y}. \quad (7)$$

We now answer **q1** by plugging the value k into the polynomial and examining the expected RTP packet rate $\bar{y}(k)$

$$\bar{y}(k) = \sum_{p=0}^{P} a_p k^p. \quad (8)$$

Remark: We have presented a polynomial generalization of the system in (Eqn. 3), where the first term is no longer restricted to be linear in form. This added flexibility allows us to model saturation effects using a higher order polynomial $y_i = \bar{y}(K(i)) + \hat{y}_i$. Note that we model discrete data using a normal distribution because (1) in other scenarios, the service metric that is modelled may be real-valued; (2) we choose the normal distribution as a first-approximation, we does not preclude further refinement of our approach using a more suitable distribution; (3) we fit the P-th order polynomial to the server's kernel metrics, some of which are real-valued. To consider **q2**, the fluctuations of the RTP packet rate for a given load on the server, we generate the squared differences signal,

$$\epsilon_{y,i} = (y_i - \bar{y}(K(i)))^2, \quad (9)$$

by subtracting the expected RTP packet rate (Eqn. 8), given the load is $K(i)$, at time i, from the observed RTP packet rate y, and model the signal $\epsilon_{y,i}$ using the Q-th order polynomial

$$\epsilon_y = \sum_{q=0}^{Q} b_q k^q. \quad (10)$$

We solve for the parameters, $\mathbf{b}$, by computing

$$\mathbf{b} = (\mathbf{M}^T\mathbf{M})^{-1}\mathbf{M}^T\epsilon_y, \quad (11)$$

where $\epsilon_y = [\epsilon_{y,1}, \epsilon_{y,2} \ldots \epsilon_{y,T}]^T$ and $b = [b_1 \ldots b_Q]^T$. An estimate of the expected fluctuation of the system, given a k active user requests on the server, is obtained by computing

$$\bar{\epsilon}_y(k) = \sum_{p=0}^{P} b_p k^p \quad (12)$$

We add the parameters $\mathbf{b}$ to the system characterization set $\theta_y = \{a_0, \ldots a_P, b_0, \ldots, b_Q\}$ –a second order characterization of the server. A similar modeling step may be taken for each of the features $\hat{\mathbf{x}}_i[n]$ which results in the characteristic set $\theta_n = \{a_0, \ldots a_P, b_0, \ldots, b_Q\}$, where $\mathbf{a}$ and $\mathbf{b}$ have the same role as before, and the feature index is denoted by the subscript n. In conclusion the entire system is characterized by $\{\theta_y, \theta_1, \ldots \theta_n, \ldots \theta_N\}$.

Remark: We have implicitly assumed that the samples used in the P-th order polynomial are iid. Fitting a Q-th order to the squared residual signal implies that we do not believe that this is the case. Inspection of the data illustrates that the variance of the observations depends on the value of the load, and thus the fit of the P-th order polynomial is affected. The purpose of the system's characterization is not to give an exact characterization, but a good approximation to aid the prediction step. The success of the prediction algorithm in the numerical evaluation section supports our approach.

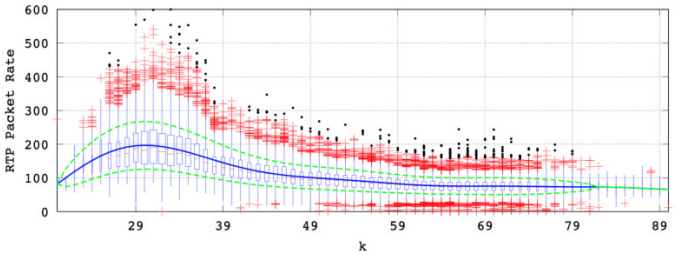

Figure 2. Boxplots and the 1st, 2nd, 3rd quartile of the underlying RTP packet rates as a function of the number of active users sessions k. The expected RTP packet rate $\bar{y}(k)$ [blue] and $\bar{\epsilon}_y(k)$ [green-dash] are also illustrated.

Learning and Prediction: To perform prediction we learn the weights that minimize the squared error in the approximation

$$y_i \approx \mathbf{w}^T\mathbf{x}_i. \quad (13)$$

When, $N = 1$, we model y_i with one feature x_i and take a maximum likelihood approach to obtain the best fit,

$$\eta_i = y_i - wx_i, \text{ where } \eta_i \sim \mathcal{N}(\gamma_i, \kappa_i). \quad (14)$$

If the mean and standard deviation of the random variables y_i and x_i depend on the load, $\mu_y(k), \mu_x(k), \sigma_y(k)$ and $\sigma_x(k)$ respectively, it follows that $\eta_i \sim \mathcal{N}(\mu_y - w\mu_x, \sigma_y^2(k) + w^2\sigma_x^2(k) - 2w\text{Cov}(x_i(k), y_i(k)))$, which gives the likelihood

$$L = \prod_{i=1}^{T} \frac{1}{\sqrt{2\pi\kappa_i^2}} e^{-\frac{\eta_i - \gamma_i}{2\kappa_i^2}}. \quad (15)$$

Remark: Another way of thinking about this likelihood function, is that it is composed of the product of a number of translated and dilated likelihoods; one translation and dilation for each value of the load k. In an attempt to remove the dependence on the load, k, and thus to be able to use all of the samples in prediction, we standardize y_i and x_i. For a fixed value of the weight w, $\eta \sim \mathcal{N}(0, \sigma^2)$ where $\sigma^2 = 1 + w^2 - 2w\rho$ and ρ is the correlation between y_i and x_i. We assume the correlation is the same irrespective of k. The log likelihood is

$$\ln L = \text{Const.} - \frac{1}{2\sigma^2}\sum_i (y_i - wx_i)^2. \quad (16)$$

When $N > 1$, for each value of the load we have the sets

$$\{\mathcal{H}(y)|_k, \mathcal{H}(\mathbf{x}_i[1])|_k, \ldots \mathcal{H}(\mathbf{x}_i[n])|_k, \ldots \mathcal{H}(\mathbf{x}_i[N])|_k\}. \quad (17)$$

For many of the features, the load signal component, $K(i)$, is present. It causes the distribution of the values of the associated RTP packet rate and features to be translated and dilated. We denote estimates of these translations and dilations, $\bar{y}(k)$, $\bar{\epsilon}_y(k)$, $\bar{\mathbf{x}}_i[n](k)$ and $\bar{\epsilon}_n(k)$ respectively. These estimates are obtained by computing the P-th and Q-th order polynomials described above, for set in the set of sets in (Eqn. 17). To learn the mapping between $\mathcal{H}(y)|_k$ and $\mathcal{H}(\mathbf{x}_i)|_k$ for any load value it is necessary to undo this translation and dilation for the RTP packet rate and each of the features in turn. This mapping is achieved for the service level metric and the n-th feature by

$$y_i' \leftarrow \frac{\mathcal{H}(y_i)|_k - \bar{y}(K(i))}{\max(\text{Re}\{\sqrt{\bar{\epsilon}_{y,i}(K(i))}\}, 1)}, \quad (18)$$

$$\mathbf{x}_i'[n] \leftarrow \frac{\mathcal{H}(\mathbf{x}_i[n])|_k - \bar{\mathbf{x}}_i[n](K(i))}{\max(\text{Re}\{\sqrt{\bar{\epsilon}_{n,i}(K(i))}\}, 1)}, \quad (19)$$

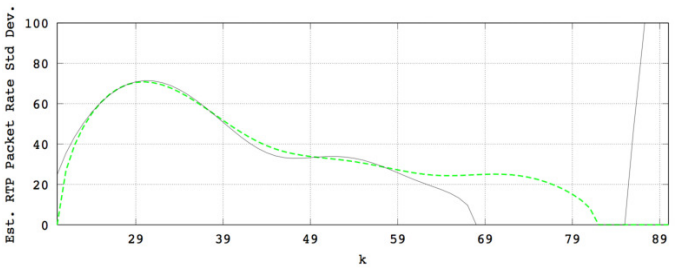

Figure 3. Estimated standard deviation of the RTP packet rate vs. the number of active user sessions k: green dashed line, using all of the data; and black dashed line, in the missing data case.

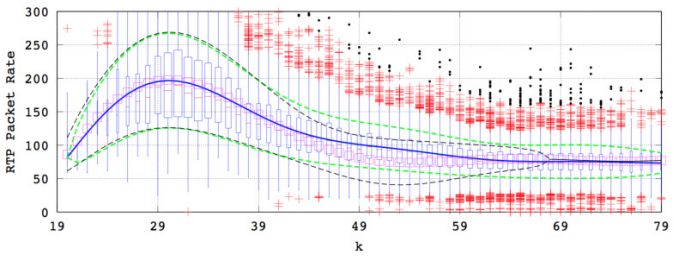

Figure 4. Predictions of the typical RTP packet rate and fluctuations in the RTP packet rate for a range of unseen values of the load $41 \leq k \leq 59$. The expected RTP packet loss for the missing data case, $\bar{y}(k)$, is indicated by pink squares. The expected fluctuations, $\bar{\epsilon}_y(k)$ are indicated by black dashed lines.

and produces standardized random variables, y'_i and $\mathbf{x}'_i[n]$. The operator $\text{Re}\{\cdot\}$ returns the real part of its argument. To perform prediction, we learn the weights, $\mathbf{w}$, that minimize:

$$L' = \prod_{i=1}^{T} \frac{1}{\sqrt{2\pi\kappa_i^{2'}}} e^{-\frac{\eta'_i - \gamma'_i}{2\kappa_i^{2'}}}, \text{ where } \eta'_i = y'_i - \mathbf{w}^T \mathbf{x}'_i. \quad (20)$$

The argument, $\eta'_i \sim \mathcal{N}(\eta'_i, \kappa_i^{2'})$, is normally distributed, with mean $\eta'_i = \boldsymbol{\mu}_{y'} - \mathbf{w}^T \boldsymbol{\mu}_{x'} = 0$, and variance $\kappa_i^{2'} = \sigma^2$, the variance of the linear combination of the features. Once we have determined the weights that minimize the objective, $\mathbf{w}^\star$ and obtained an estimate $y'_{i+1} = \mathbf{w}^\star \mathbf{x}'_{i+1}$ we can undo the effect of standardization by applying the inverse mapping,

$$y_{i+1} = y'_{i+1} \max(\text{Re}\{\sqrt{\bar{\epsilon}_{y,i}(K(i+1))}\}, 1) + \bar{y}(K(i+1)), \quad (21)$$

which scales the standardized estimate y'_{i+1}, by the standard deviation estimate obtained from the training data $\sqrt{\bar{\epsilon}_{y,i}(K(i+1))}$ and adds the mean offset $\bar{y}(K(i+1))$ for a load of value $K(i+1)$.

IV. Numerical Evaluation

We evaluate the PQ-model in two types of scenario: (1) We observe the performance of the system for every load value in the range, $19 \leq k \leq 90$. There is no missing data in the traces. The purpose of this evaluation is to demonstrate the accuracy of the system characterization step. We establish our ability to answer **q1** and **q2**. (2) We are given limited training data. We do not observe the behaviour of the RTP packet rate for certain values of the load –the missing data problem– we demonstrate our ability to answer **q3**. Finally, we evaluate **q4**, the improvements gained in predicting the RTP packet rate by including the system characterization step described above in the prediction process. We pre-process the traces by removing all non-numeric and constant valued features from the set of server features. As a result of this pre-processing step we use $N = 231$ features. We draw $\approx 50k$ samples under different load conditions from the server and the client machine and use a subset of these samples as training data and the remaining samples subset is used as test data.

(1) Supervised System Characterization: We characterize the performance of the server using the PQ-model above. We illustrate the results for $\{P, Q\} = \{40, 24\}$ and the observations of the client's RTP packet rate. We do not include our analysis of each of the server's features, but instead focus on the client's RTP packet rate due to space constraints. In order to select the best $\{P, Q\}$ we evaluate the quality of the fit achieved using different values of P independently of Q. Once P is chosen we choose the Q that achieves the best fit. Fig. 2 illustrates y_i, where a box is constructed for each value of the load. The load increases from $k = 19$ to $k = 90$ on the x-axis. The boxplots provide a benchmark for the PQ-model. We overlay a P-th order polynomial fit for the expected/ideal RTP packet rate, $\bar{y}(k)$, for each k using a blue line (in Fig. 2).

(1) The P-th order polynomial is approximately equal to the median. The difference between the median and the P-th order polynomial is explained by the median's robustness to outliers. There are more *too-high* outliers in the range $19 \leq k \leq 30$ than *too-low* outliers when $k > 30$. (2) The relationship between the load and the client's RTP packet rate is not linear. The linearity assumption is reasonable in the range $19 \leq k \leq 30$. When the load is above $k = 30$, the server becomes overloaded and the number of RTP packets that the user receives begins to decrease. The linear model in (Eqn. 3) does not accurately characterize the system. Fig. 2 gives evidence that a higher order relationship gives a better system characterization. The Q-th order polynomial, that models the deviation signal, $\epsilon_{y,i}$, is illustrated as an upper and lower range for the P-th order polynomial, $\bar{y}(k) \pm \bar{\epsilon}_y(k)$, e.g. two dashed green lines in Fig. 2 (and in isolation in Fig. 3). The range of fluctuation in the number of RTP packets the target client receives initially increases as the number of clients increases. This relationship is approximately linear in the range $19 \leq k \leq 30$. However, when $k > 30$ the fluctuation decreases. If the client receives fewer packets because the server is saturated, it is reasonable to assume that the range of fluctuations of the packet count will be similarly reduced. (1) The Q-th order polynomial accounts for the increase in fluctuation when the load is approximately 30 active user requests. Fig. 3 illustrates that the fluctuation of the RTP packet rate is 70 packets when the server is serving 30 active user requests. It is 20 RTP packets when the server is serving 19 active user requests. The Q-th order polynomial is potentially an over-estimate of the fluctuation; this observation is based on the fact that the 1st and 3rd quartiles are in general closer to the median than the fluctuation estimate. (2) The standard deviation of the RTP packet rate is clearly a function of the load on the system, an assertion that contradicts the claim that the standard deviation of the RTP packet rate is the same, irrespective of the load on the system [1]. The PQ-model allows us to rule out the use of a simple linear model.

Missing data System Characterization: We address **q3** by establishing our ability to estimate how the RTP Packet will behave when a load we have not observed before is placed on the server. This type of prediction problem is of interest from

TABLE I. COMPARISON OF ESTIMATES OF $\bar{y}(k)$, WHEN $k = 40$ IS INCREASED BY 2.5-37.5% CONSECUTIVELY WHEN (1) NO SAMPLES ARE AVAILABLE FOR THESE LOADS AND (2) SAMPLES ARE AVAILABLE.

%	2.5	7.5	12.5	17.5	22.5	27.5	32.5	37.5
Missing	133.7	119.9	110.6	105.9	104.4	104.5	104.1	102.1
All	136.9	124.5	114.9	107.9	102.8	98.8	95.3	91.7

TABLE II. COMPARISON OF ESTIMATES OF $\bar{\epsilon}_y(k)$, WHEN $k = 40$ IS INCREASED BY 2.5-37.5% CONSECUTIVELY WHEN (1) NO SAMPLES ARE AVAILABLE FOR THESE LOADS AND (2) WHEN SAMPLES ARE AVAILABLE.

%	2.5	7.5	12.5	17.5	22.5	27.5	32.5	37.5
Missing	45.4	38.3	32.8	30.1	29.8	30.7	31.3	30.6
All	47.9	42.6	38.3	35.1	33.0	31.4	30.2	28.8

the perspective of network planning and resource allocation. From the point of view of the network manager, the ability to estimate how a client will be affected if an existing server is not supplemented by another server, when the load increases by 10%, for example, is of interest. We assume that the service level metric, y_i, is only captured when the load on the system is in either of the ranges $20 \leq k \leq 40$ or $60 \leq k \leq 90$. Instead of using all 51043 samples from the traces to characterize the system, we use 33294 samples. The goal is to estimate how the client's service level metric will typically behave in the range $41 \leq k \leq 59$. In addition we want to estimate what types of fluctuations to expect when the server is under this type of load. The PQ-model computed for the entire data-set (along with the boxplots) are illustrated as before (blue line and green dashed lines) in Fig. 4. To evaluate the predictive power of the PQ-model, a P-th order polynomial is fit to the data-set which is missing samples for loads in the range $41 \leq k \leq 59$. Pink squares denote the estimated ideal behaviour of y_i in the ranges $20 \leq k \leq 40$ and $60 \leq k \leq 90$, and also, in the range $41 \leq k \leq 59$ where we have no observations of y_i. The pink squares are aligned with the P-th order polynomial computed using the entire data-set and also the median of the boxplots. The parameters used to fit these polynomials are $P = 9$ and $Q = 57$. These parameters are chosen by evaluating the quality of the fit of the P-th order polynomial to the samples, drawn when $20 \leq k \leq 40$ or $60 \leq k \leq 90$, and then, the quality of the fit of the Q-th order polynomial to the same samples. The underpinning assumption is that if the fit is good when the load is $20 \leq k \leq 40$ or $60 \leq k \leq 90$, it will also be good when the load is $41 \leq k \leq 59$.

This result has practical significance. Consider the following resource allocation problem: if we have a good estimate of how the RTP packet rate will behave when $19 \leq k \leq 40$, e.g. $\bar{y}(k)$, we are then interested in predicting how the RTP packet rate is affected if the load is 2.5-37.5% larger. Accurate estimates of $\bar{y}(k)$ facilitate improved network planning; one action, for example, if our predictor tells us that the RTP packet rate will be too low, is to deploy additional VoD servers. Does our PQ-model give us a sufficiently good estimate when data is missing to justify the cost of purchasing and deploying an additional server? Table I compares the $\bar{y}(k)$ estimates obtained from a P-th order polynomial when no data is missing, namely the row "All", with a P-th order polynomial obtained obtained when 65% of the samples are missing, namely "Missing", and none of the samples used lie in the range of loads of interest, $41 \leq k \leq 59$. To put these estimates in context, an increase of 2.5% in the load corresponds to a load of $k = 41$ user requests; an increase of 37.5% corresponds to a load of $k = 55$ users. Table I demonstrates that the P-th order polynomial which is fit to 65% gives estimates of the P-th order polynomial, which is fit to all of the data, which are within 1-10 packets. Moreover, we increased the load by up to 50% and the results give approximately the same accuracy. Black dashed lines in Fig. 4 illustrate the estimate of a Q-th order fit to estimate the fluctuations of the server over the entire range: $20 \leq k \leq 40$ and $60 \leq k \leq 90$, and also the range of loads for which we have no samples, $41 \leq k \leq 59$. The estimate is closely aligned with the Q-th order polynomial fit using the entire data-set in the range $20 \leq k \leq 53$. The RTP packet rate fluctuation estimate is heavily influenced by outliers above $k \geq 53$. This is explained by the fact that we have an uneven number of samples for every value of the load. Table II demonstrates that once again the difference in the estimates of the fluctuations in the RTP packet rates are approximately 1-3 RTP packets.

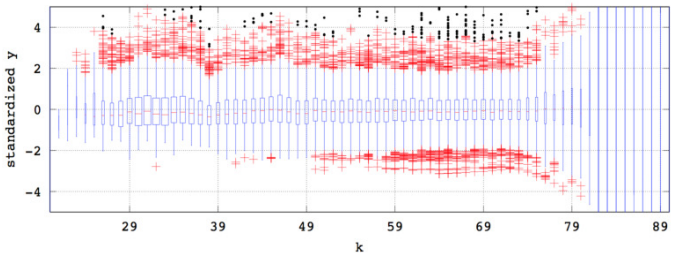

Figure 5. Effect of standardization on y_i. Irrespective of the load on the system, the samples are drawn from approximately the same distribution.

In summary, the PQ-model may be used to predict the typical RTP packet rate and the potential range of fluctuations in the RTP packet rate of a target client even when a significant proportion, 35% in the example given above, of the data is missing, and in this particular case, no observations are available for the load value in question. For example if the number of active users is increased from 40 users to 51 users, the RTP packet rate of the client is predicted to fall from a level of 144 Packets to 104.5 ± 30.7 (when all of the data is used to fit the model) or to 98.8 ± 31.4 (when some the data is missing for the load values $41 \leq k \leq 59$). The difference between the typical RTP packet rate using the missing data estimate 104.5 and the entire data-set 98.8 is small enough for us to have confidence in making a good management decision, e.g. deploying an additional server, based on the missing data estimate. The two estimates of the potential fluctuations in the RTP packet rate are within 1 packet of each other.

Remark: It is important to note that some of the features assume discrete levels approximately. Fitting a smooth polynomial function may not be the best approach in these cases; however, exhaustively selecting functions to model features may not be practicable and fitting a polynomial is computationally cheap and fast. Secondly, some of the transitions between the different levels of the x-boxplots may be quite sharp. A polynomial fit may not accurately capture these transitions. Numerical evaluations demonstrate that it is a good trade-off between accuracy and speed.

Predicting Missing Data: We determine the accuracy of the predictor for certain load values when training data corresponding to these load values is missing. The goal is to predict the value of the y_{i+1} using the features observed $\mathbf{x}_{i+1}$, when observations for the load at time $i+1$ have not previously been observed. That is, we evaluate the accuracy of the predictor

TABLE III. LOAD VALUES k_{target} TESTED (ROW 1), THE NUMBER SAMPLES PREDICTED (ROW 2) AND THE MEAN OF y_i (ROW 3).

41	42	43	44	45	46	47	52	53	54	55	58	59
823	866	944	891	931	875	789	962	1006	984	999	1041	968
123	121	117	114	112	105	100	93	93	90	88	82	80

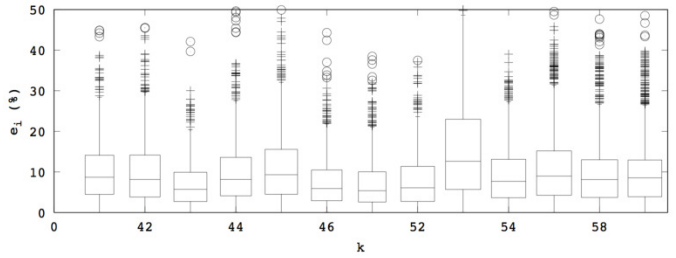

Figure 6. Boxplots (one for each value of k_{target}) of the percentage absolute error for each predicted RTP packet rate.

derived by minimizing the likelihood function L'. The flow of control of these experiments is summarized as follows. We observe features from the server and the client's service level metric at all times indices where the load is in the set $\mathcal{K} = \{k|20 \leq k \leq 90\} \setminus k_{\text{target}}$. The index k_{target} is the value of the load which corresponds to the RTP packet rates we are trying to predict. The load values tested are tabulated in row 1 of Table III. Row 2 lists the number of samples predicted for each of these load values. We do not have any training data (features and service-level metrics) for any of the load values k_{target}. We fit a P-th and Q-th order polynomial to the features and service level metric for all samples corresponding to the set $\mathcal{K}$. These samples are the training data. This is a missing-data problem as the P-th and Q-th order polynomials are learned for all load values except for k_{target}. We standardize this training data and then learn the weights $\mathbf{w}$ that minimize the likelihood function L'. We use these weights to generate predictions of the RTP packet rate when the load is k_{target}, using the features observed at these times. In order to use the weights, we standardize the features using the P-th and Q-th order polynomial computed for the missing-data problem above. For each of the sets of features at each sampling point, we generate a standardized prediction for the value of the RTP packet rate. We then dilate and translate (using Eqn. 21) the predictions using the P-th and Q-th order polynomial computing for the RTP packet rate above (for samples when the load is $k \in \mathcal{K}$).

We evaluate the accuracy of these, I, predictions by computing the absolute difference between the true RTP packet rate (not standardized) and the predicted value of the RTP packet rate which has the inverse mapping of the standardization applied to it (Eqn. 21). We then normalize the absolute difference by the average true RTP packet rate and scale this fraction by 100 in order to obtain a percentage, $e_i = 100I(|y_i - y_i'|)/(\sum_{i=1}^{I} y_i)$. Fig. 6 illustrates boxplots of the precentage asbolute error for each of the predicted RTP packet rate values. For almost all load values, the error in the prediced RTP packet rate for 50% of the predictions is less than 10% of the mean RTP packet rate for y_i. For convenience, the mean of y_i for each value of the load is listed in row 3 of Table III. If the mean RTP packet rate is 123 packets per second for a given load value, greater than 50% of the predictions made have an error which is less than 12 RTP packets. This result is remarkable given that we have no training data for each of the load values for which predictions were made. Moreover, the range of means in Table III is 43 RTP packets and the range of RTP packet rates observed is considerably higher. In summary we have demonstrated that accurate RTP packet rate predictions can be obtained even if there is no training data for certain load values by using the PQ-model to characterize the system. In future work we will develop strategies to counter-act the sensitivity of the prediction algorithm to the inverse mapping described in (Eqn. 21) and we will evaluate the PQ-model and prediction in a wider range of prediction scenarios.

V. CONCLUSIONS

We proposed a generative model for the RTP packet rate received by a target client who has requested VoD from a server shared with other users. Unlike previous approaches, (1) it does not assume that there is a linear relationship between the RTP packet rate received and the load on the server; (2) it accurately predicts ranges for the range of RTP packet rate the client will receive, even that observations for that value of the load have not been experience by the system. Remarkably, good estimates are computed for these statistics, even when the load is increased by 50%. (3) It accurately predicts RTP packet rates in the absence of training data. We contribute evidence that a linear model is inappropriate; we propose a number of tools for network planning; and finally, for service level prediction when there is missing data.

REFERENCES

[1] R. Yanggratoke, J. Ahmed, J. Ardelius, C. Flinta, A. Johnsson, D. Gillblad, and R. Stadler, "Predicting real-time service-level metrics from device statistics," *IFIP/IEEE Int. Sym. Int. Net. Man.*, pp. 1–8, 2015.

[2] Ruairí de Fréin, "Take off a load: Load-adjusted video quality prediction and measurement," *IEEE Int. Conf. Depend., Autonom. and Sec. Comp.*, pp. 1–9, 2015.

[3] Ruairí de Fréin, C. Olariu, Y. Song, R. Brennan, P. McDonagh, A. Hava, C. Thorpe, J. Murphy, L. Murphy, and P. French, "Integration of QoS Metrics, Rules and Semantic Uplift for Advanced IPTV Monitoring," *J. of Net. and Sys. Man.*, pp. 1–36, 2014.

[4] Cisco, "Cisco visual networking index, Global IP Traffic Forecast, 2011-2016.," *white paper*, 2011.

[5] A. Begen, T. Akgul, and M. Baugher, "Watching video over the web Part 2: Applications, standardization, and open issues," *IEEE Int. Comp.*, vol. 15, no. 3, pp. 59–63, 2011.

[6] Ruairí de Fréin, "Ghostbusters: A parts-based NMF algorithm," in *IET Irish Sig. Sys. Conf. (ISSC)*, 2013, pp. 1–8.

[7] Ruairí de Fréin, "Formal concept analysis via atomic priming," in *Formal Concept Analysis*, vol. 7880 of *LNCS*, pp. 92–108. Springer, 2013.

[8] M. Zinkevich, A. Smola, and J. Langford, "Slow learners are fast," in *NIPS*, 2009, pp. 2331–9.

[9] J. Bogojeska, D. Lanyi, I. Giurgiu, G. Stark, and D. Wiesmann, "Classifying server behavior and predicting impact of modernization actions," in *IEEE Conf. Net. Serv. Man. (CNSM)*, 2013, pp. 59–66.

[10] M. Mirza, J. Sommers, P. Barford, and X. Zhu, "A Machine Learning Approach to TCP Throughput Prediction," *Networking, IEEE/ACM Trans.*, vol. 18, no. 4, pp. 1026–39, 2010.

[11] A. Andrzejak and L. Silva, "Using machine learning for non-intrusive modeling and prediction of software aging," in *IEEE Net. Op. Man. Symp. (NOMS)*, 2008, pp. 25–32.

[12] F. Zaman, S. Robitzsch, Zhuo W., J. Keeney, S. van der Meer, and G. Muntean, "A heuristic correlation algorithm for data reduction through noise detection in stream-based communication management systems," in *IEEE Net. Op. Man. Symp. (NOMS)*, 2014, pp. 1–8.

[13] S. Zhang, I. Cohen, M. Goldszmidt, J. Symons, and A. Fox, "Ensembles of models for automated diagnosis of system performance problems," in *Proc. Int. C. Dependable Systems and Networks*, 2005, pp. 644–653.