Chapter 6
Service Providers' Long-Term QoS Prediction Model

6.1 Introduction

It is natural for different providers to compete in the cloud market to maximize their profits using their individual economic models. The *performance index* of a provider can be calculated using the information of resource utilization, price fairness, consumers' satisfactions and providers' profits [71]. It is necessary for a provider to compare its performances with the overall performance of the cloud market. For example, a higher performance index of the market indicates that most providers are making profits while maintaining service level agreements. Conversely, a lower performance index of a provider may suggest the provider should find some alternative solutions to become competitive and profitable in the market.

The future performance of peer IaaS providers may play an important role to construct an effective long-term economic model. For example, if it is predicted that the cloud service prices will be reduced in near future, the provider may place an emphasis on new service configurations with added functionality to attract new consumers and to increase its revenue. The quantitative and qualitative composition approaches described in Chaps. 3–5 assume that the provider sets its economic expectations and preferences before the composition. In the real world, these preferences are influenced by the market performance [34]. For example, in 2015 Rackspace Hosting reduced its rates towards the levels charged by other global cloud providers, such as Amazon Web Services (AWS), Microsoft Azure and Google [114].

The future QoS performance of the provider is also important in the long-term composition from the consumers' perspective. The long-term QoS by a service provider may vary for various reasons, e.g. a change in resource allocation policy, resource sharing, multi-tenancy or economic models [68]. Hence, the long-term QoS guarantees from a service provider may not always be available. For

S. Mistry et al., *Economic Models for Managing Cloud Services*,
https://doi.org/10.1007/978-3-319-73876-5_6

example, in Amazon EC2, only the long-term QoS "availability" is advertised [3]. In Windows Azure, the I/O performance is advertised for short-term periods only in comparative terms such as high, moderate and low [92]. A QoS prediction model is thus needed to obtain long-term QoS values before creating the composition.

We propose a *multivariate* prediction model for QoS values. Existing QoS prediction models do not usually consider existing correlations among the QoS attributes in performance history [84, 98, 144, 148, 151, 152]. However, these correlations are prevalent in cloud service composition where a QoS attribute is correlated with one or many other QoS attributes. For example, response times and throughput usually have a strong negative correlation. Hence, when predicting the time-series for a particular QoS attribute, we need to consider the historical time-series of the QoS attribute and any correlated QoS attributes. For example, the predicted values of response times would decrease while the predicted values of throughput increase. Each cloud service provider will generate a series of predicted QoS values for its cloud service through this prediction model, based on the history and the short-term advertisement. *We assume that the past QoS history is provided by the service provider or by any trusted third party that monitors the providers QoS performances over a period.*

The chapter is structured as follows. The QoS prediction framework and the multivariate prediction model are discussed in Sects. 6.2 and 6.3 respectively. The forecasting model, experiments and conclusion are presented in Sects. 6.4, 6.5, and 6.6 respectively.

6.2 The Multivariate QoS Forecasting Framework

The QoS forecasting problem is formulated as follows. Let a cloud composition plan SC consist of k service providers $\{S_i \mid i = 1, 2, .., k\}$. Each service provider supplies QoS history and short-term advertisements in time-series. The QoS model consists of n attributes and corresponding attribute values where Q_{j_t} refers to the value of the QoS attribute j at the time t. Thus, the QoS history of the service provider S_i is defined in a vector $HIS(S_i) = \{\ (Q_{1_t}, Q_{2_t}, \ldots, Q_{n_t}) \mid t \in [1, m]\ \}$ and the short-term advertisement is defined in a vector $ADV(S_i) = \{(Q_{1_t}, Q_{2_t}, \ldots, Q_{n_t}) \mid t \in (m, s]$ and $m < s\}$. The problem is to predict the long-term QoS performance of the composition $PRD(SC) = \{\ (Q_{1_t}, Q_{2_t}, \ldots\ldots, Q_{n_t}) \mid t \in (m, z]$ and $m < s \ll z\}$.

The multivariate QoS prediction model (Fig. 6.1) has three main parts: prediction model, forecasting, and the aggregator. The multivariate prediction model takes a QoS history as an input from the service providers. The model is then fed to a prediction module to predict QoS values up to a time specified by end users. Necessary adjustments need to be performed based on the difference between the predicted values and the short-term advertisements published by service providers. In the aggregator, the predicted QoS values of all service providers in the service

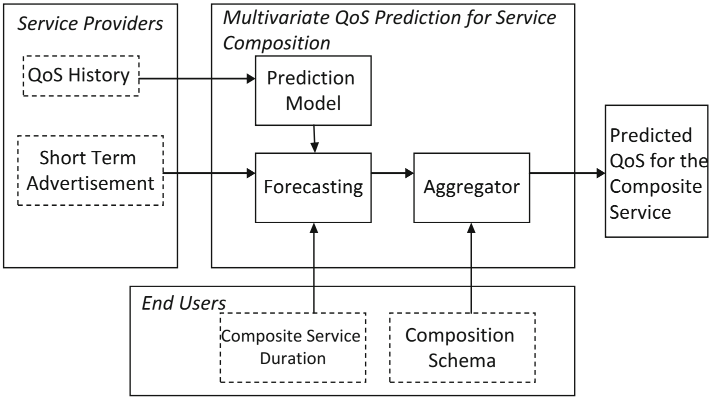

Fig. 6.1 The multivariate QoS prediction framework

composition are aggregated, based on the composition schema. The aggregation rules rely on the QoS attributes. *We only focus on the prediction and forecasting module in the following sections.*

6.3 Multivariate QoS Prediction Model (MQPM)

Given the history of a time-series, a model is required to identify the pattern of the time-series based on observed values [19]. ARIMA and Holt-Winters are widely used to model univariate time-series [35, 75]. ARIMA is an improved form of linear regression and Holt-Winters is an enhanced form of exponential smoothing. If we decompose the multivariate QoS history into individual univariate time-series, each time-series can be fitted into a univariate model. For example, the multivariate QoS history $History(Responsetime_t, Throughput_t)$ is decomposed into two individual univariate time-series, $Responsetime_t$ and $Throughput_t$ in Fig. 6.2. $Responsetime_t$ best fits the ARIMA model and $Throughput_t$ best fits the Holt-Winters model. There are errors between the predicted and observed values (Fig. 6.2). We believe these errors can be further minimized using the correlations among the QoS attributes.

Response time and *Throughput* have a strong negative correlation in Fig. 6.2. An increase in *Response time* is followed by a decrease in *Throughput* and vice-versa. Thus the predicted *Throughput* values obtained from the univariate models can be further decreased if an increasing trend in *Response time* is observed. Another multivariate time-series prediction model, the Vector Autoregressive (VAR) model, assumes constant correlations among QoS attributes. This fails to predict QoS values effectively if there are random changes in trend or seasonality in the QoS history. Therefore we devise a new multivariate QoS prediction model named MQPM.

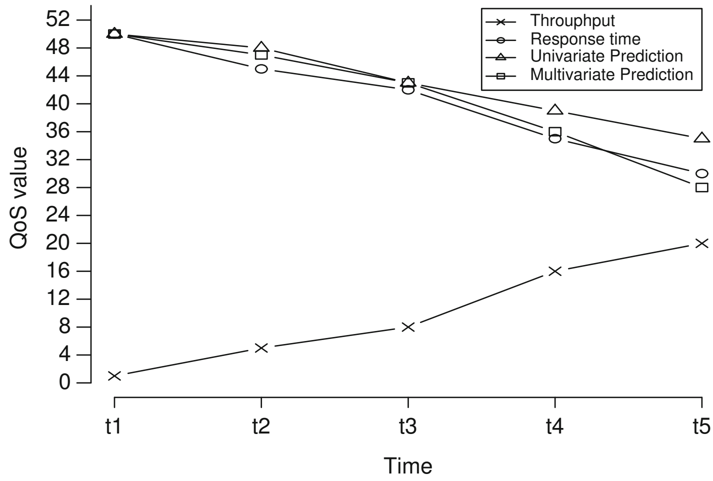

Fig. 6.2 Prediction error reduction using a multivariate analysis

We define some terms related to this model before its formal presentation.

- **Backshift Operator** (B): In time-series analysis, a backshift operator operates on an element of a time-series to produce previous elements. For example, given a time-series of a QoS attribute as $x = \{ Q_{x_1}, Q_{x_2}, .., Q_{x_t}, .., Q_{x_{t+m}}\}$; the backshift operator is B when $BQ_{x_t} = Q_{x_{t-1}}$ or $B^{-1}Q_{x_t} = Q_{x_{t+1}}$ and $B^kQ_{x_t} = Q_{x_{t-k}}$.
- **Cross Backshift Operator** ($CB_{x,y}$): The cross backshift operator operates on an element of a time-series to produce the previous elements of another time-series. For example, given a time-series of a QoS attribute x as $x = \{Q_{x_1}, Q_{x_2}, \ldots, Q_{x_t}, \ldots, Q_{x_{t+m}}\}$ and a time series of another QoS attribute y as $y = \{Q_{y_1}, Q_{y_2}, \ldots, Q_{y_t}, \ldots . Q_{y_{t+m}}\}$; the cross backshift operator is $CB_{x,y}$ when $CB_{x,y}Q_{x_t} = Q_{y_{t-1}}$ or $CB^{-1}_{x,y}Q_{x_t} = Q_{y_{t+1}}$ and $CB^k_{x,y}Q_{x_t} = Q_{y_{t-k}}$.
- **Correlation Operator** ($CP_{x,y}$): Given $\bar{Q}_x$ and $\bar{Q}_y$ as the mean of time-series x and y, and S_{Q_x} and S_{Q_y} as their standard deviations; the correlation operator between the time-series x and y is a Pearson Correlation:

$$CP_{x,y} = \frac{1}{m-1}\sum_{i=1}^{m}\left(\frac{Q_{x_i} - \bar{Q}_x}{S_{Q_x}}\right)\left(\frac{Q_{y_i} - \bar{Q}_y}{S_{Q_y}}\right) \tag{6.1}$$

- **Root Mean Square Error** (RMSE): RMSE is used to evaluate the performance of a prediction model. Let us assume the time-series of an individual QoS attribute in the QoS history fits many prediction models. If the predicted time-series are $(\hat{Q}_{1_t}, \hat{Q}_{2_t}, \ldots, \hat{Q}_{n_t})$, the prediction error is calculated using Eq. 6.2. A lesser value of RMSE imposes a better prediction model.

$$RMSE(i) = \sqrt{\frac{\sum_{i=1}^{m}(\hat{Q}_{i_t} - Q_{i_t})^2}{m}} \tag{6.2}$$

MQPM uses the cross backshift operator and the correlation operator to reduce RMSE in individual univariate models in two phases. In the first phase, we transform ARIMA and Holt-Winters into the MQPM as stated in Sects. 6.3.1 and 6.3.2 respectively. In the second phase, we devise an algorithm for adaptive prediction error reduction in the MQPM as stated in Sect. 6.3.3.

6.3.1 Transforming ARIMA into the MQPM

Using the cross backshift operator and the correlation operator, we extend the univariate ARIMA(p,d,q) model into a multivariate model in Eq. 6.3. Three parts are integrated in this equation. In the *first* part, the *autoregressive (AR)* depends on the p_n lagged values of the time-series of n_{th} QoS attributes and φ_{n_p} is the coefficient constant. In the *second* part, The *moving average (MA)* depends on the q_n lagged values of the previous prediction errors and θ_{n_p} is the coefficient constant. B_n is a backshift operator for the n_{th} QoS attribute. The *third* part is the multivariate error reduction attributes where k_n is the lagged values of the $\{1, 2, \ldots., n-1\}$ QoS attributes, $CP_{n,i}$ is the correlation operator of the QoS attribute i relative to the QoS attribute n; and $\alpha_{n,i,j}$ is the constant coefficient of the time-series value of the QoS attributes i relative to the time-series value of the QoS attributes n at the j_{th} time slot. In ARIMA, a non-stationary time-series needs to be converted into a stationary time-series by a difference operation. Here, d_n represents the number of times that the difference operation is performed to obtain the stationary time-series.

$$
\begin{aligned}
&(\sum_{i=1,i\neq 1}^{n}\sum_{j=1}^{k_1}\alpha_{1,i,j}(CP_{1,i})(CB^{j}_{1,i}))\hat{Q}_{1_t} + ((1-\sum_{i=1}^{p_1}\varphi_{1_i}B^{i}_{1})(1-B_1)^{d_1})\hat{Q}_{1_t} = (1+\sum_{i=1}^{q_1}\theta_{1_i}B^{i}_{1})\varepsilon_{1_t}\\
&\qquad\qquad\qquad\qquad\qquad\qquad \ldots.\\
&\qquad\qquad\qquad\qquad\qquad\qquad \ldots. \qquad\qquad\qquad\qquad\qquad\qquad (6.3)\\
&(\sum_{i=1,i\neq n}^{n}\sum_{j=1}^{k_n}\alpha_{n,i,j}(CP_{n,i})(CB^{j}_{n,i}))\hat{Q}_{n_t} + ((1-\sum_{i=1}^{p_n}\varphi_{n_i}B^{i}_{n})(1-B_n)^{d_n})\hat{Q}_{1_t} = (1+\sum_{i=1}^{q_n}\theta_{n_i}B^{i}_{n})\varepsilon_{n_t}
\end{aligned}
$$

6.3.2 Transforming Holt-Winters into the MQPM

It is uncertain that attributes in multivariate QoS time-series will always fit in ARIMA individually. For example, in the dataset [67], 80 service providers' QoS history among 94 service providers best fitted in the ARIMA model. The remainder best fitted in Holt-Winters. To reduce the error in Holt-Winters, the multivariate error reduction part in Eq. 6.3 is also used to extend the Holt-Winters model to multivariate models in Eq. 6.4. Here, $\hat{Q}_{n_{t+1}}$ is the predicted value at $t+1$ time slot. At time t, m_{n_t} describes the level, b_{n_t} describes trend and $c_{n_{t-s}}$ describes seasonality where s specifies the seasonal period. x_{n_t} is the multivariate error reduction factor as described in Eq. 6.3. η_n, β_n and γ_n are the constant coefficients for calculating the level, trend and seasonality respectively.

$$
\begin{aligned}
\hat{Q}_{1_{t+1}} &= m_{1_t} + b_{1_t} + c_{1_{t-s}} + x_{1_t} \\
\hat{Q}_{2_{t+1}} &= m_{2_t} + b_{2_t} + c_{2_{t-s}} + x_{2_t} \\
&\cdots\cdot \\
&\cdots\cdot \\
\hat{Q}_{n_{t+1}} &= m_{n_t} + b_{n_t} + c_{n_{t-s}} + x_{n_t} \\
\text{where,} & \\
m_{n_t} &= \eta_n(Q_{n_t} - c_{n_{t-s}}) + (1-\eta_n)(m_{n_{t-1}} + b_{n_{t-1}}) \\
b_{n_t} &= \beta_n(m_{n_t} - m_{n_{t-1}}) + (1-\beta_n)b_{n_{t-1}} \\
c_{n_t} &= \gamma_n(Q_{n_t} - m_{n_t}) + (1-\gamma_n)c_{n_{t-s}} \\
x_{n_t} &= \sum_{i=1, i\neq n}^{n} \sum_{j=1}^{k_n} \alpha_{n,i,j}(CP_{n,i})(CB^j_{n,i})
\end{aligned}
\tag{6.4}
$$

6.3.3 Prediction Error Reduction Algorithm in the MQPM

Given a multivariate QoS history, we first decompose the time-series into individual univariate time-series. Then we fit each time series into an ARIMA model using the Box-Jenkins method [19] and into a Holt-Winters model [29]. If the RMSE of ARIMA is less than the RMSE of Holt-Winters, we will choose Eq. 6.3, otherwise we choose Eq. 6.4. In Eq. 6.3, the optimal values of AR and MA parts are calculated by the Box-Jenkins method. If Eq. 6.4 is used, the optimal values of coefficients are calculated by the Holt-Winters method. To have a reduced RMSE in either the Box-Jenkins or the Holt-Winters method, we need to find the optimal values of k_n and $\alpha_{n,i,j}$ of the multivariate error reduction factor in either equation for the n_{th} QoS attribute of the QoS history.

6.3.3.1 Calculating k_n for the n_{th} QoS Attribute

Let us assume that $n = \{Q_{n_1}, Q_{n_2}, Q_{n_3} \ldots\ldots Q_{n_m}\}$ and $i = \{Q_{i_1}, Q_{i_2}, Q_{i_3} \ldots\ldots Q_{i_m}\}$ are two time-series of the n_{th} and i_{th} QoS attributes in the history respectively. To calculate a k lagged coefficient of i for n, we first transform n and i into

$n_{m-k} = \{Q_{n_t} \mid k < t \leq m\}$ and $i_{m-k} = \{Q_{i_t} \mid 1 \leq t < (m-k)\}$ respectively, and then calculate the correlation operator $CP_{n_{m-k}, i_{m-k}}$ using Eq. 6.1. As it is a Pearson correlation, values above 0.40 can be treated as candidate lagged values. In this process we calculate all the lagged values (k) for QoS attributes 1 to $n-1$ for the n_{th} QoS attribute.

6.3.3.2 $\alpha_{n,i,j}$ for the n_{th} QoS Attribute

The values of $\alpha_{n,i,j}$ should be sufficiently adaptable to reduce the RMSE from the first phase of ARIMA and Holt-Winters for the n_{th} attribute of the QoS history. Let us assume the i_{th} attribute is the only correlated attribute to the n_{th} QoS attribute and there is a first lagged operation of i to n. Using Eq. 6.3 or 6.4 in the second phase, the predicted values of the n_{th} QoS attribute can be written as:

$$Q^2_{n_t} = Q^1_{n_t} + \alpha_t (CP_{n,i}) Q_{i_{t-1}} \tag{6.5}$$

As α_t is used to reduce the RMSE in the second phase; we should find optimal values of α_t and also predict the time-series of α_t. We first calculate α_t from the observed values. To do so, let us assume there is no prediction error in Eq. 6.5. Thus, in the second phase, the predicted values become the observed values $Q^2_{n_t} = Q_{n_t}$. Substituting Q_{n_t} in Eq. 6.5, we get values for α from Eq. 6.6.

$$\alpha_t = \frac{Q_{n_t} - Q^1_{n_t}}{(CP_{n,i}) Q_{i_{t-1}}} \quad \mid 1 \leq t \leq m \tag{6.6}$$

As α_t can be considered as a time-series, we can model α_t prediction $\hat{\alpha}_t$ using the univariate ARMIA or Holt-Winters model in Eq. 6.7.

$$\hat{\alpha}_t = \left\{ \begin{array}{l} \text{ARIMA(p,d,q) on } \alpha_t \text{ , where} \\ \text{RMSE(ARIMA(p,d,q)}(\alpha_t)) < \text{RMSE(Holt-Winters}(\alpha_t)) \\ \text{Holt-Winters}(\alpha_t) \text{ , where} \\ \text{RMSE(ARIMA(p,d,q)}(\alpha_t)) \geq \text{RMSE(Holt-Winters}(\alpha_t)) \end{array} \right\} \tag{6.7}$$

To find the prediction pattern of α coefficients for all correlated attributes, we devise an iteration on sorted QoS attributes as described in Algorithm 6.

6.4 Forecasting from the MQPM

We need the long-term forecast using the QoS history. Using the proposed multi-variate analysis, the QoS history is modelled as $\{\hat{Q}_{1_t}, \hat{Q}_{2_t}, \ldots, \hat{Q}_{n_t}\}$. As the model uses observed values, we can only forecast one step ahead, i.e. $\hat{Q}_{n_{t+1}}$. End users

Algorithm 6 Finding $\hat{\alpha}_t$ for all the co-related attributes of the n_{th} QoS attribute

Require: The multivariate time-series of n QoS attributes $(Q_{1_t}, Q_{2_t}, \ldots, Q_{n_t})$ and the set of correlation operators between the n_{th} and other QoS operators $\{CP(n,1), CP(n,2), \ldots, CP(n, n-1)\}$

Ensure: The set of predicted adaptive coefficient $\hat{\alpha}_t$ for the n_{th} QoS attribute from $\{1, 2, .., n-1\}$ QoS attributes

```
a[n-1] is the descending-order-sorted array of attributes based on CP(n, i)
Q^1_{n_t} is the first-phase prediction on the n_th QoS attribute
for i := 1 to n − 1 do
    Get α_t using Equation 6.6
    Store predicted α̂_t for attribute i using Equation 6.7
    Q^1_{n_t} = Q^1_{n_t} + α_t(CP_{n,i})Q_{i_{t−1}}
end for
```

require a long-term forecast, i.e. l-step forecast. If the QoS history is modelled with Eq. 6.3, we formulate the l-step forecast for the n_{th} QoS attribute using the recursive procedure [47] in Eq. 6.8.

$$\hat{Q}_{n_{t+l}} = \sum_{i=1}^{p_n} \varphi_{n_i} \hat{Q}_{n_{t+l-i}} + \sum_{i=1, i \neq n}^{n} \sum_{j=1}^{k_n} (\hat{\alpha}_{n,i,t+l-j}) \hat{Q}_{i_{t+l-j}} CP(n,i) \tag{6.8}$$

The l-step forecast of the coefficient adaptive factor $(\hat{\alpha}_{n,i,t+l-j})$ is calculated using the prediction procedure in Sect. 6.3.3.2. The prediction of the correlated attributes also uses Eq. 6.8. All the QoS attributes are predicted in parallel operations. Thus the predicted values of $(\hat{Q}_{1_{t+l}}, \hat{Q}_{2_{t+l}}, \ldots, \hat{Q}_{n_{t+l}})$ are only computed when $(\hat{Q}_{1_{t+l-1}}, \hat{Q}_{2_{t+l-1}}, \ldots, \hat{Q}_{n_{t+l-1}})$ have already been computed. A similar procedure is followed in Eq. 6.4. As Eq. 6.4 uses recursive values of previous forecasts, they can be treated as the observations in the l-step forecast.

The short-term advertisements of a service provider may change the behavior of forecasting. Let us assume that $(\hat{Q}_{1_{t+s}}, \hat{Q}_{2_{t+s}}, \ldots, \hat{Q}_{n_{t+s}})$ is a s-step ahead short-term forecast and $(Q_{1_{t+s}}, Q_{2_{t+s}}, \ldots, Q_{n_{t+s}})$ is the short-term advertisement from the service provider. If the difference between the RMSE of the predicted forecast and the short-term advertisement is unacceptable, there could well be a change in the service provider's delivery system. In fact advertisements often tends to show improved QoS. If a user believes that the RMSE difference is acceptable, the user can use Eq. 6.8 for forecasting; otherwise, the user may adjust the forecast with a level-adjusting factor. The resulting l-step QoS forecast adjusted by a short-term advertisement is presented in Eq. 6.9.

$$\begin{aligned} \hat{Q}_{n_{t+l}} &= \sum_{i=1, i \neq n}^{n} \sum_{j=1}^{k_n} (\hat{\alpha}_{n,i,t+l-j}) \hat{Q}_{i_{t+l-j}} CP(n,i) + \gamma \\ \gamma &= \sum_{i=1}^{p_n} \varphi_{n_i} \hat{Q}_{n_{t+l-i}} + \frac{\sum_{i=1}^{s} |Q_{n_{t+i}} - \hat{Q}_{n_{t+i}}|}{s} \end{aligned} \tag{6.9}$$

Table 6.1 Summary of the real-world QoS dataset

QoS attribute	Values
Min response time (r_{min})	13 ms
Max response time (r_{max})	1538 ms
Avg response time (r_{avg})	337 ms
Min throughput (th_{min})	0.0
Max throughput (th_{max})	5.0
Avg throughput (th_{avg})	3.2
Avg correlation operator $CP(r, th)$	0.62

6.5 Experiments and Results

A set of experiments have been conducted to assess the performance of the proposed prediction model. We compare the proposed multivariate prediction model with MQPM, ARIMA, Holt-Winters, and VAR. All experiments have been conducted on computers with Intel Core i7 CPU (2.13 GHz and 4 GB RAM).

6.5.1 Data Description

We evaluate the proposed approach using real cloud service data [67]. These real data include two time-series data (i.e. response time and throughput) for 100 cloud services. In each service, the QoS history is 6 months old (28 time slots). We discard six service providers' data as there are zero throughput and zero response times for most of the time in the time-series. The summary of the dataset is given in Table 6.1.

The response time and the throughput have a strong correlation in the dataset. To support the decision-making, we add the attribute of cost into the dataset which has no correlation with response time or throughput. The costs of services stay at a stable level for a time interval and reduce over the period based on the market situation [45]. We generate the cost with random distribution and reduce it over random time periods.

6.5.2 Comparison Among the Prediction Models

In this experiment, we compare the performance of the proposed multivariate model (MQPM) with ARIMA, Holt-Winters, and VAR. We use the statistical package R to generate the predicted values of the models. Using Eq. 6.2, the RMSE is calculated based on the observed and predicted values. Our target is to improve the prediction, i.e. a reduction in the RMSE. The results of the RMSE in *throughput* and *response time* for the services are presented in Figs. 6.3 and 6.4 respectively. We

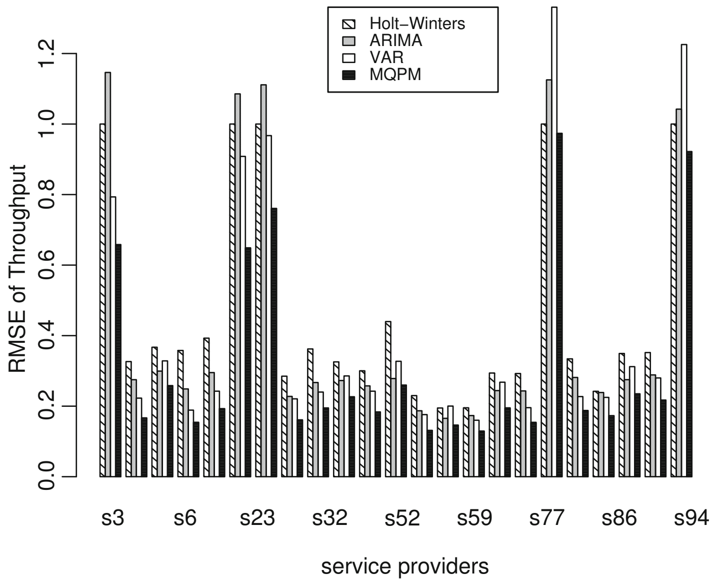

Fig. 6.3 RMSE of throughput in different service providers

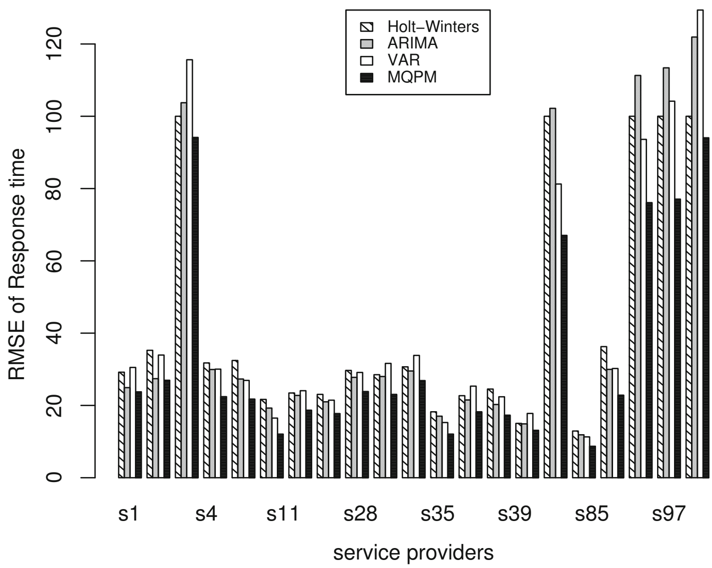

Fig. 6.4 RMSE of response time in different service providers

Table 6.2 Summary of the performance of prediction models

Prediction model	RMSE		
	Avg. throughput	Avg. response time	Avg. cost
MQPM	0.32	59	236
ARIMA	0.43	89	245
Holt_winters	0.55	75	265
VAR	0.53	82	257
RMSE reduction by MQPM	26%	21.3%	0.04%

randomly select 40 service providers and show the RMSEs of the prediction models on those providers in Figs. 6.3 and 6.4. We find that the proposed model, MQPM has lower RMSEs than the other models for each service provider. The average RMSEs for the prediction models are depicted in Table 6.2. We see that the MQPM reduces RMSEs significantly in *throughput* and *response time*. As the cost is not correlated with other attributes, the proposed model performs as a univariate model here.

6.5.3 Effect of the Time-Series Length on MQPM

In this experiment, we discuss the effect of QoS history length on the proposed multivariate prediction model. The performance of the proposed model relies on the adjusting correlation factor α. The more accurately α is predicted, the more accurate the prediction model is. To assess the effect of QoS history length on MQPM, we need to generate different time-series that vary in length but retain similar correlations among the QoS attributes. There are two variation methods: we may either extend or divide the QoS history. As our multivariate time-series dataset is comprised only of 28 real readings, the lengths of the divided time-series are too small to affect the prediction. On the other hand, the correlations among the QoS attributes may not exist if the dataset is extended at random. Therefore, we generate a new set of time-series with varying lengths by extending a random original QoS history ten times, using the ARIMA model. This is because ARIMA can retain the correlations in the simulated dataset. In each extension, we perform the RMSE of MQPM and plot them in Fig. 6.5. We observe that the RMSE of the original time series reduces in each extension. In Fig. 6.5, the RMSE is reduced by about 50% after the original time series is extended by ten times. From this experiment, we conclude that MQPM performs better when the number of time slots in the QoS history becomes larger.

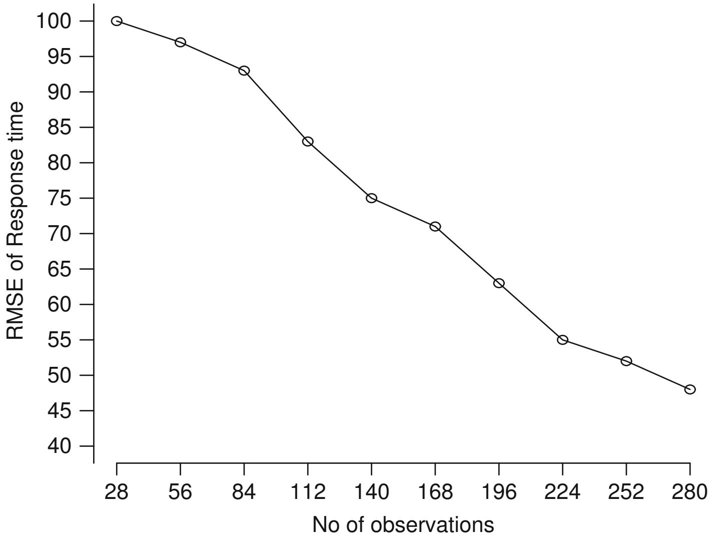

Fig. 6.5 Effect of time-series length on RMSE

6.6 Conclusion

We have used QoS history and short-term advertisements to predict the long-term QoS behavior of service providers. Correlations among the QoS attributes are used to improve the prediction error rate. In this chapter, we do not consider the trust issues in the QoS history and the short-term advertisements. Incorporating the trust issue into the proposed model is a subject for possible future work.