

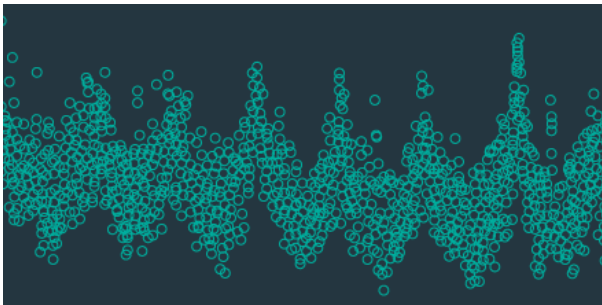


RSS

[\(https://content.pivotal.io/blog.rss\)](https://content.pivotal.io/blog.rss)[Featured \(https://content.pivotal.io/\)](https://content.pivotal.io/) » [Blog \(https://content.pivotal.io/blog\)](https://content.pivotal.io/blog) » Forecasting Time Series Data with Multiple Seasonal...

# Forecasting Time Series Data with Multiple Seasonal Periods

SEPTEMBER 22, 2015

[ANIRUDH KONDAVEETI \(HTTPS://CONTENT.PIVOTAL.IO/AUTHORS/ANIRUDH-KONDAVEETI\)](https://content.pivotal.io/authors/anirudh-kondaiveeti)

Time series data is produced in domains such as IT operations, manufacturing, and telecommunications. Examples of time series data include the number of client logins to a website on a daily basis, cell phone traffic collected per minute, and temperature variation in a region by the hour. Forecasting a time series signal ahead of time helps us make decisions such as planning

capacity and estimating demand. **Previous time series analysis blog posts**

(<https://blog.pivotal.io/tag/time-series-analysis>) focused on processing time series data that resides on Greenplum database using SQL functions. In this post, I will examine the modeling steps involved in forecasting a time series sequence with multiple seasonal periods. The various steps involved are outlined below:

- Multiple seasonality is modelled with the help of fourier series with different periods
- External regressors in the form of fourier terms are added to an ARIMA model to account for the seasonal behavior
- Akaike Information Criteria (AIC) is used to find the best fit model

An example of time series data which shows the number of people logging in a gaming website over two months is shown in Figure 1. The graph shows the number of gamers logging in each hour. The X-axis represents the hour of the day and the Y-axis tracks the number of gamers who are logging in the website in that hour. This time series demonstrates seasonal behavior, with the number of gamers logging in following both daily and weekly seasonality. The highlighted regions in the plot correspond to weekends, during which logins are considerably higher than the weekdays. Within a single day, the number of logins increases towards mid-day and decreases thereafter. We explain the modeling steps to forecast this time series signal using the best line of fit.

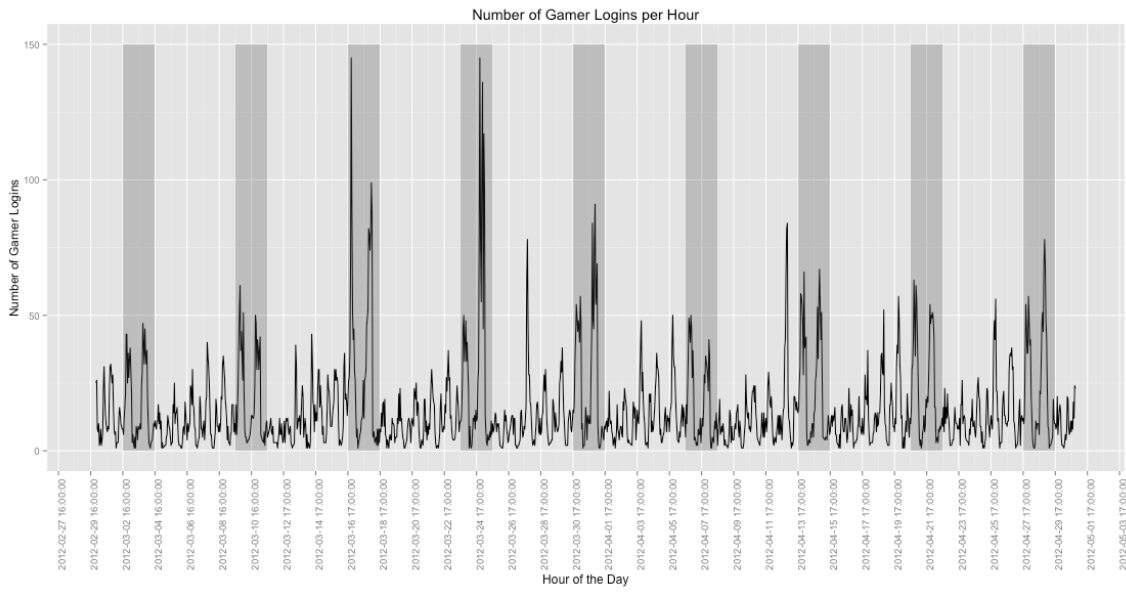


Fig. 1: Time series plot of number of gamer logins per hour

Autoregressive integrated moving average (ARIMA) models are generally used to model time series data, however they do not directly handle seasonality. The ARIMA model regresses the current data value against historical data value(s) in the time series. In order to deal with multiple seasonality, external regressors need to be added to the ARIMA model[1]. To incorporate the multiple seasonality in the gamer login behavior, additional Fourier terms are added to the ARIMA model, where  $N_t$  is an ARIMA process.

$$y_t = a + \sum_{i=1}^M \sum_{k=1}^{K_i} \left[ \alpha \sin\left(\frac{2\pi kt}{p_i}\right) + \beta \cos\left(\frac{2\pi kt}{p_i}\right) \right] + N_t$$

The seasonal pattern is modeled through the addition of Fourier terms which are used as external regressors. This approach is flexible, allowing you to incorporate multiple periods. For example, if there are 'M' periods ( $p_1, p_2, p_3, p_M$ ) in the data, we would have different fourier series corresponding to each of the 'M' periods. In this particular example, there are two seasonal periods, daily and weekly, with  $p_1$  including 24 hours and  $p_2$  including 168 hours. For each of the periods ' $p_i$ ', the number of fourier terms ( $K_i$ ) are chosen to find the best statistical model for a given set of data. Given a set of models, AIC (which is derived from information theory) estimates the quality of model compared to other models, and thus provides a means for model selection. The value of  $K_i$  is chosen so as to minimize the AIC criteria.

Heat Map of AIC Values for Varying Number of Fourier Terms corresponding to the 2 Periods

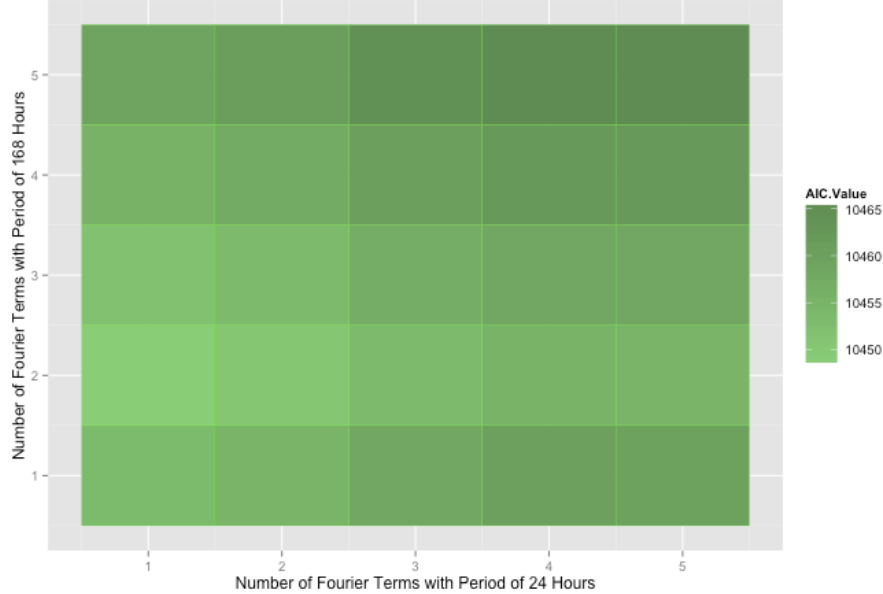


Fig. 2: Heat Map showing the variation of AIC values with number of Fourier terms. It can be seen that the minimum AIC value is at point (1,2)

In order to find the right number of Fourier terms corresponding to each of the periods, the AIC values of the ARIMA model with varying Fourier terms are calculated. The scatter plot for the AIC values with varying number of Fourier terms for the two periods p1 (24 hours) and p2 (168 hours) is shown in Fig. 2. From the figure, it can be seen the best model, which minimizes the AIC criteria, is one which has one Fourier term corresponding to a period of 24 hours and two Fourier terms corresponding to a period of 168 hours with an AIC value of  $\sim 10450$ .

The R code to tune the number of Fourier terms and find the minimum AIC values is shown below. The variable “xreg1” generates a Fourier series with a period of 24 hours and the number of terms are varied using the variable “i”, from 1 to 5. Similarly, the variable “xreg2” generates a fourier series with a period of 168 hours and the number of terms are varied using the variable “j” from 1 to 5. For each combination of “i” and “j”, an ARIMA model is fit using the variables “xreg1” and “xreg2” as external regressors and the corresponding AIC value is calculated. Once all the AIC values are obtained, the best parameters “i” and “j” are those that correspond to the minimum AIC value.

```

1 aic_vals_temp <- NULL
2 aic_vals <- NULL
3 for(i in 1:5)
4 {
5   for (j in 1:5)
6   {
7     xreg1 <- fourier(1:length(NumEvents),i, 24)
8     xreg2 <- fourier(1:length(NumEvents), j, 24*7)
9     xtrain <- cbind(xreg1, xreg2)
10    fitma1 <- auto.arima(NumEvents, D= 0 , max.P = 0, max.Q = 0, xreg = xtrain)
11    aic_vals_temp <- cbind(i, j, fitma1$aic)
12    aic_vals <- rbind(aic_vals, aic_vals_temp)
13  }
14 }
15 colnames(aic_vals) <- c("FourierTerms24", "FourierTerms168", "AICValue")
16 aic_vals <- data.frame(aic_vals)
17 minAICVal <- min(aic_vals$AICValue)
18 minvals <- aic_vals[which(aic_vals$AICValue == minAICVal),]
19

```

Code 1: Finding the optimum number of fourier terms for a given time series

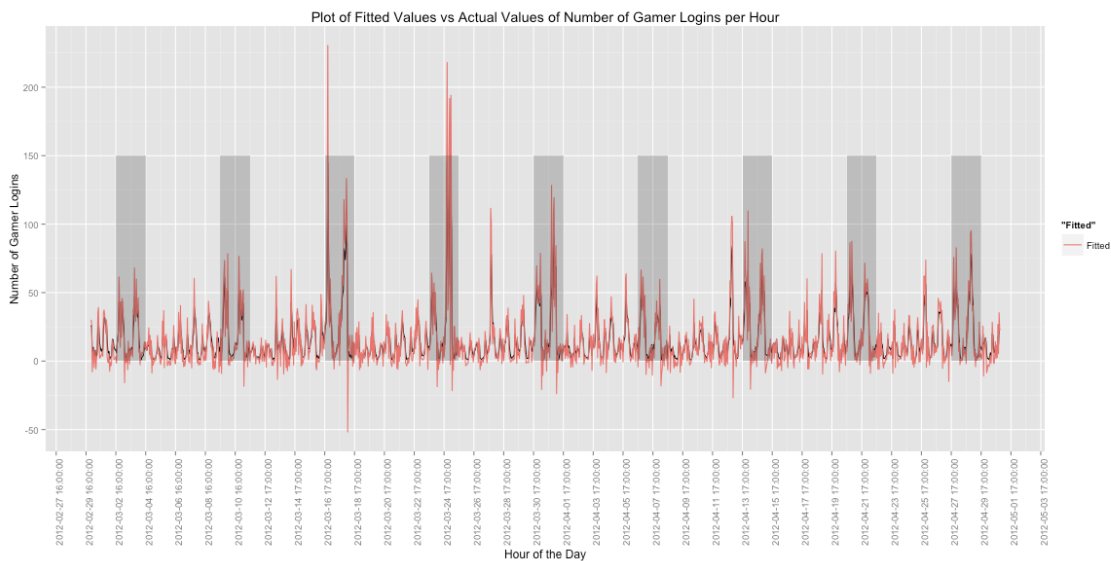


Fig. 3: Plot showing the fitted values for the time series. Red line is the fitted value and the black line is the actual value

The best line of fit for the time series is an ARIMA (4, 0, 3) model , including four autoregressive terms and three moving terms, with one Fourier term corresponding to a period of 24 hours and two Fourier terms corresponding to a period of hours. The plot of the fitted values is shown in Fig. 3, wherein the highlighted gray regions correspond to the weekends, the red line corresponds to the fitted values, and the black line corresponds to the actual values. The coefficients and the standard errors (s.e.) for the various terms are shown below.

Coefficients:													
	ar1	ar2	ar3	ar4	ma1	ma2	ma3	intercept	S1	C1	S1	C1	S2
	1.3013	-0.7254	0.8390	-0.5471	-0.5010	0.2364	-0.4211	15.6296	1.7243	-0.4052	4.8646	3.3971	-0.2570
s.e.	0.0764	0.1275	0.1417	0.0663	0.0854	0.1042	0.0949	0.5986	1.7314	1.7315	0.8709	0.8775	0.9548

The equation corresponding to the final model is

$$y_t = 15.6296 + 1.3013 \times y_{t-1} - 0.7254 \times y_{t-2} + 0.839 \times y_{t-3} - 0.5471 \times y_{t-4} - 0.501 \times z_{t-1} + 0.2364 \times z_{t-2} - 0.4211 \times z_{t-3} + 1.7243 \times \sin\left(\frac{2\pi t}{24}\right) - 0.4052 \times \cos\left(\frac{2\pi t}{24}\right) + 4.8646 \times \sin\left(\frac{2\pi t}{168}\right) + 3.3971 \times \cos\left(\frac{2\pi t}{168}\right) - 0.257 \times \sin\left(\frac{2\pi 2t}{168}\right) - 2.9108 \times \cos\left(\frac{2\pi 2t}{168}\right)$$

where  $y_{t-1}$ ,  $y_{t-2}$ ,  $y_{t-3}$ ,  $y_{t-4}$  correspond to the autoregressive terms and  $z_{t-1}$ ,  $z_{t-2}$ ,  $z_{t-3}$  correspond to the moving average terms. The external regressors correspond to fourier terms with a period of 24 hours (single frequency) and period of 168 hours (two different frequencies).

To visualize the effect of Fourier terms which are used as external regressors in the model, a plot of these values with time ranging from 1 to 350 hours is shown in Fig. 4. The value in the figure is the sum of sine and cosine terms shown in the equation above for each period. As can be seen from the figure, the red line corresponds to the external regressors that capture the weekly seasonality in the model (a period of 168 hours), and the blue line corresponds to regressors that capture the daily seasonality (a period of 24 hours). The forecast from the time series is thus the sum of the forecasts obtained from the ARIMA (4, 0, 3) model and the additional effect from the fourier signal.

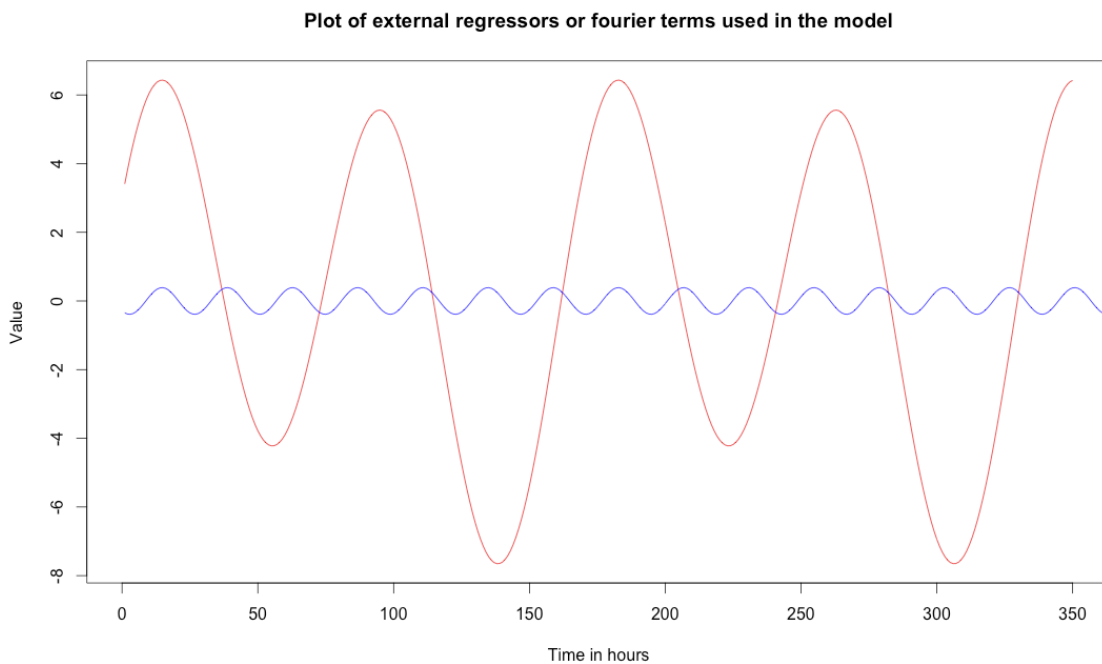


Fig. 4 : Fourier terms used as external regressors in the forecasting model. Red line corresponds to weekly period of  $24 \times 7$  hours and blue line corresponds to a daily period of 24 hours

Once the appropriate model is fit to the time series, the next step is to use this model to predict the future values. The code used for predicting the next ten values is shown below. The data from 1 to 750 hours is used for training, while the data for the next 10 hours from 751 to 760th hour is used for testing. The variable “*xreg1*” in the code below creates fourier terms with a period of 24 hours and “*xreg2*” creates fourier terms with a period of 168 hours. The variable “*xtest*”, which is a combination of terms from the variables “*xreg1*” and “*xreg2*”, is used as external regressor to the predict function with the model built using training data as the input.

```
xreg1 <- fourier(751:760,1, 24)
xreg2 <- fourier(751:760, 2, 24*7)
xtest <- cbind(xreg1, xreg2)
predict(fitma1, newxreg = xtest)
```

Code 2 : Forecasting the time series values using the fitted model

We have outlined the modeling approach to forecast time series with two seasonal periods: daily and weekly. This method can be easily extended if we have more seasonal periods by adding additional Fourier terms corresponding to each period. In general, it is recommended to forecast less than 5 values in advance, because the error in forecasts increases if we forecast far ahead in the future. This is because the ARIMA model uses the past values to predict the future values, and if we predict too much into the future, the error-prone predictions obtained previously would be used as training data.

The massively parallel processing (MPP) capabilities of Pivotal Greenplum Database and HAWQ can be used to forecast multiple time series at different nodes in a scalable and parallel manner. As an example, consider predicting the cell phone outages by analyzing the time series data of cellphone usage from various data centers. A separate forecasting model can be built for each data center using Procedural Language R (PL/R) to monitor each data center independently. To illustrate with an example, consider the table below called *input\_table*, which consists the time series of performance values aggregated in the form of an array for different data centers.

Data_Center_ID	Time_Index	Performance_Values
D1	[1, 2, 3, 4, 5, ..]	[23.4, 10.4, 3.2, 4.8, 2.2, ..]
D2	[1, 2, 3, 4, 5, ..]	[5.5, 10.7, 11.1, 3.7, 10.3, ..]
D3	[1, 2, 3, 4, 5, ..]	[13.2, 18.4, 31.2, 47.8, 22.1, ..]
D4	[1, 2, 3, 4, 5, ..]	[34.4, 1.7, 13.8, 5.6, 7.9, ..]
.	.	.
.	.	.

input\_table consisting of time series of performance values of different data centers

The PL/R function below can be used to find the optimum number of Fourier terms required for model selection for each of the data centers. All the computations are performed in-database. As it can be seen from the code below (Code 3), the PL/R function is similar to Code 2 with an SQL like wrapper.

```

1  create or replace function calculate_optimum_fourier_terms(performance_values numeric[])
2  returns integer[] as
3  $$
4      NumEvents <- performance_values
5      aic_vals_temp <- NULL
6      aic_vals <- NULL
7      for(i in 1:5)
8      {
9          for (j in 1:5)
10         {
11             xreg1 <- fourier(1:length(NumEvents),i, 24)
12             xreg2 <- fourier(1:length(NumEvents), j, 24*7)
13             xtrain <- cbind(xreg1, xreg2)
14             fitma1 <- auto.arima(NumEvents, D= 0 , max.P = 0, max.Q = 0, xreg = xtrain)
15             aic_vals_temp <- cbind(i, j, fitma1$aic)
16             aic_vals <- rbind(aic_vals, aic_vals_temp)
17         }
18     }
19     colnames(aic_vals) <- c("FourierTerms24", "FourierTerms168", "AICValue")
20     aic_vals <- data.frame(aic_vals)
21     minAICVal <- min(aic_vals$AICValue)
22     minvals <- aic_vals[which(aic_vals$AICValue == minAICVal),]
23     returnval <- c(minvals$FourierTerms24, minvals$FourierTerms168)
24     return(returnval)
25 $$
26 language plr;

```

Code 3: PL/R function calculate\_optimum\_fourier\_terms to find the optimum number of fourier terms

The above function *calculate\_optimum\_fourier\_terms* can be used to find the optimum number of fourier terms for each of the time series from the data centers, by executing the function on each row of the input\_table as below. The results are stored in a table called *optimum\_parameter\_vals*.

```

1  create table optimum_parameter_vals as
2  select
3      Data_Center_ID,
4      calculate_optimum_fourier_terms(Perfomance_Values) as optimum_vals
5  from    input_table
6  distributed by (Data_Center_ID);

```

Code 4: Create table optimum\_parameter\_vals to store the optimum number of fourier terms for each time series

The table *optimum\_parameter\_vals* created contains the optimum number of fourier terms for the time series from each data center. A sample of the table *optimum\_parameter\_vals* is shown below:

Data_Center_ID	Time_Index	Performance_Values	Optimum_Vals
D1	[1, 2, 3, 4, 5, ..]	[23.4, 10.4, 3.2, 4.8, 2.2, ..]	[1, 2]
D2	[1, 2, 3, 4, 5, ..]	[5.5, 10.7, 11.1, 3.7, 10.3, ..]	[2,1]
D3	[1, 2, 3, 4, 5, ..]	[13.2, 18.4, 31.2, 47.8, 22.1, ..]	[2,2]
D4	[1, 2, 3, 4, 5, ..]	[34.4, 1.7, 13.8, 5.6,7.9 , ..]	[3,1]
.	.	.	.
.	.	.	.

optimum\_parameter\_vals table consisting of the optimum fourier terms for each time series

Once the optimum number of Fourier terms for the time series are obtained, the following function *forecast\_time\_series* can be used to fit the appropriate model. This uses the optimum number of Fourier terms and then forecasts the time series signal using the learned model. The function takes the time series signal, optimum parameters (Fourier terms), and the number of desired forecasted points as the inputs and outputs of the forecasted values.

```

1  create or replace
2  function
3  forecast_time_series(performance_vals numeric[], optimum_params integer[], num_forecasts integer)
4  returns numeric[]
5  as
6  $$
7      # Fitting the model on training data
8      NumEvents <- performance_vals
9      tslength <- length(NumEvents)
10     tslengthfinal <- tslength + num_forecasts
11     parameter1 <- optimum_params[1]
12     parameter2 <- optimum_params[2]
13     xreg1 <- fourier(1:tslength,parameter1, 24)
14     xreg2 <- fourier(1:tslength, parameter2, 168)
15     xtrain <- cbind(xreg1, xreg2)
16     fitma1 <- auto.arima(NumEvents, D= 0 , max.P = 0, max.Q = 0, xreg = xtrain)
17
18     # Predicting using test data
19     xtest1 <- fourier((tslength + 1): tslengthfinal, parameter1, 24)
20     xtest2 <- fourier((tslength + 1): tslengthfinal, parameter2, 168)
21     xtest <- cbind(xtest1, xtest2)
22     predicted_vals <- predict(fitma1, newxreg = xtest)
23     return(predicted_vals)
24  $$
25

```

Code 5: Function *forecast\_time\_series* to learn and forecast the time series signal

The above function *forecast\_time\_series* can be used to obtain the next 5 values for each of the time series signal from the different data centers, by executing it on each row of the input data. The results are stored in *forecasted\_vals* table.

```

1  create table forecasted_vals as
2  select
3      Data_Center_ID,
4      Time_Index,
5      Performance_Vals,
6      forecast_time_series(Performance_Vals, optimum_vals, 5) as Forecasted_Vals
7  from    optimum_parameter_vals
8  distributed by (Data_Center_ID);
9

```

Code 6 : Create table *forecasted\_vals* to find the next 5 forecasted values for each time series.

The table *forecasted\_val* created contains the next 5 forecasted values for the time series from each data center. A sample of the table *forecasted\_val* is shown below. The column *Forecasted\_Vals* consists the forecasted values from the time series.



Data_Center_ID	Time_Index	Perfomance_Values	Forecasted_Vals
D1	[1, 2, 3, 4, 5, ..]	[23.4, 10.4, 3.2, 4.8, 2.2, ..]	[3.6, 1.2, 22.4, 4.8, 9.2]
D2	[1, 2, 3, 4, 5, ..]	[5.5, 10.7, 11.1, 3.7, 10.3, ..]	[6.5, 2.9, 4.1, 8.0, 4.7]
D3	[1, 2, 3, 4, 5, ..]	[13.2, 18.4, 31.2, 47.8, 22.1, ..]	[15.9, 19.3, 11.6, 10.9, 17.2]
D4	[1, 2, 3, 4, 5, ..]	[34.4, 1.7, 13.8, 5.6,7.9 , ..]	[5.9, 3.3, 1.9, 22.8, 19.8]
.	.	.	.
.	.	.	.

forecasted\_vals table with the forecasted values for each time series

In this blog post, we have shown how to use the Pivotal Greenplum Database to develop a univariate time series forecasting model in a scalable and parallelizable fashion. In the next blog post, we will examine forecasting a more advanced time series signal involving multiple variables using Vector Autoregressive (VAR) models for specific IT operations applications, such as virtual machine capacity planning.

- [1] <http://robjhyndman.com/hyndsight/longseasonality/>  
(<http://robjhyndman.com/hyndsight/longseasonality/>)
- [2] [http://doc.madlib.net/master/group\\_\\_grp\\_\\_arima.html](http://doc.madlib.net/master/group__grp__arima.html)  
([http://doc.madlib.net/master/group\\_\\_grp\\_\\_arima.html](http://doc.madlib.net/master/group__grp__arima.html))
- [3] <https://blog.pivotal.io/author/caleb-welton> (<https://blog.pivotal.io/author/caleb-welton>)

### About the Author

Anirudh Kondaveeti is a Principal Data Scientist and the lead for Security Data Science at Pivotal. Prior to joining Pivotal, he received his B.Tech from Indian Institute of Technology (IIT) Madras and Ph.D. from Arizona State University (ASU) specializing in the area of machine learning and spatio-temporal data mining. He has developed statistical models and machine learning algorithms to detect insider and external threats and "needle-in-hay-stack" anomalies in machine generated network data for leading industries.



More Content by Anirudh Kondaveeti (<https://content.pivotal.io/authors/anirudh-kondaveeti>)

Previous

<https://content.pivotal.io/blog/operations-has-plenty-to-do-in-a-cloud-native-enterprise>

Operations Has Plenty To Do In ...

In an age when the role of IT Operations is shif...

Next

<https://content.pivotal.io/blog/dealing-with-the-stuff-that-makes-all-the-money-the-legacy-journey-part-3>

Dealing With The Stuff That Mak...

Part three of the Cloud Native Journey series i...

19 Comments

Pivotal Blog

3 brookm291

Recommend

Share

Sort by Best

Join the discussion...

**ved piyush** • 9 months ago  
Hello Anirudh,  
  
I don't think the equation you show for the final model in this article is one that we will get by using the fourier terms as xreg in the auto.arima function of R. Please see the answer by Richard Hardy in link below, I think auto.arima does regression with ARMA errors so the ARIMA(4,0,3) coefficients that you get should only be for the error structure



and not for the series  $y_t$  itself. (like how Richard shows in the answer, these ARMA coefficients that you get correspond to the coefficients in  $u_t$  from Richard's answer.)

<https://stats.stackexchange...>

Thanks,  
Ved

^ | v • Reply • Share ›



**Sushant Singh** • 10 months ago

Hi , kindly let me know the package you have used for fourier function .  
Thank You !

^ | v • Reply • Share ›



**Ioanna** • a year ago

Dear Anirudh,

Thank you for sharing this post, it is really helpful. Since I am not experienced on time-series analysis, I would like to ask if it is possible for you to upload the whole R code, so that I fully understand how to handle these data.

Best,  
Ioanna

^ | v • Reply • Share ›



**Mohammad Mubarah** • a year ago

Can You Tell us please where we can get the data?

^ | v • Reply • Share ›



**Evert van Doorn** • a year ago

Dear Anirudh,

Thank you for this clear and concise write-up! I am trying to use Python to perform a similar analysis. As Python does not offer the same, convenient Fourier fitting functions, I was hoping you could specify the content of the Fourier terms which you include in the model.

Based on your output of the coefficients and standard errors, you seem to include the sines and cosines for each Fourier term  $k$  within your Fourier  $p$  subscript  $i$  series separately. Does this mean the R function, which you use to generate the Fourier terms, yields a separate series, with length equal to your number of observations, for each frequency (or Fourier term) you include in the Fourier series based on the AIC?

Your answer would really help me in finalizing the code for my project - I'm hoping you are willing to take the time to look into this!

Bests,

Evert

^ | v • Reply • Share ›



**Michael** • 2 years ago

Dear Anirudh,

Thank you for sharing this interesting blog. I would like to ask if it would be possible to get a copy of the data used in the example above to play and learn from?

Kind regards,  
Michael

^ | v • Reply • Share ›



**Michael** • 2 years ago

Hi Anirudh,

Thank you for sharing this very interesting information in this blog. I would like to ask if it would be possible to get a copy of the dataset you use "NumEvents" in order to learn and play with?

Kind Regards,  
Michael

^ | v • Reply • Share ›



**kishore** • 2 years ago

```
> for (i in 1:5){
+ for (j in 1:5){
+ xreg1<-fourier(1:2025,i,24)
+ xreg2<-fourier(1:2025,j,24*5)
+ xtrain<-bind(xreg1,xreg2)
```

```
+ xtrain<-cbind(xreg1,xreg2)
+ fitma1<-auto.arima(final_fitted1,D=0,max.P=0,max.Q=0,xreg=xtrain)
+ aic_vals_temp<-cbind(i,j,fitma1$aic)
+ aic_vals<-rbind(aic_vals,aic_vals_temp)
+ }
+ }
```

Error in `fourier(1:2025, i, 24)` : unused argument (24)

could some one please tell me what is wrong with the extra argument

^ | v • Reply • Share ›



**Rakesh** • 2 years ago

Hi,

I have a problem where time series is highly seasonal in nature. Day to day basis, a cycle starts but the start is randomly spread from 8-10 AM. After the start, more or less, the shape of time series is same through out day.

How to handle such cases.

Thanks in Advance

Rakesh

^ | v • Reply • Share ›



**J M** • 2 years ago

Hi,

What does your NumEvents look like?

eg: `NumEvents <- ts(...)`

^ | v • Reply • Share ›



**Sayantan Sengupta** • 2 years ago

hi Anirudh,

What would you do, if u have a data, whose periodogram shows component values of the order of  $10^5$  to  $10^8$  for all the frequencies. How could u model all the frequencies then? Or is there any better approach to handle this kind of scenario?

I have to fit a (S)ARIMA model to the data..that's my objective..

^ | v • Reply • Share ›



**Yash Shah** • 2 years ago

Excellent post! I have always wondered how I could do multiple seasonality . Most cases I have applied TSA, i have gotten away with a one season component which can be done simply by " `Arima(flow, order = c(4,0,3), seasonal = list(order = c(0,1,1), period = 24))`"

^ | v • Reply • Share ›



**Hasan** • 2 years ago

I am using the `fourier()` function to build fourier series terms for seasonality and then add it to an ARIMA series like mentioned above.

However, it seems the fourier function has now changed with forecast version 5.9+? The x value can no longer be `(1:length(ts))` because it would have frequency 1 and would error out with "K must be not be greater than period/2". So we need to pass 'x' as the time series itself with frequency equal to the 'period' ie. (I have data every 10 minutes) `24 * 6` for daily patterns and `7 * 24 * 6` for weekly patterns . (Setting the frequency in the 'ts()' function however results in oversampling the data by the frequency value and scaling the length of the time series) Is this understanding correct or am I missing something ?

^ | v • Reply • Share ›



**Pramod Gupta** • 2 years ago

Hi Anirudh,

Thank you very much for a nice article. I am also interested in time series analysis. Could you please let me know how to take care of missing values. Let us say I am looking into data for six months and somehow I am missing data for week or a month. I to take care of it while fitting ARIAM model. I will really appreciate your quick response. Can you refer me to some good reference also.

Regards,

Pramod

^ | v • Reply • Share ›



**Anirudh Kondaveeti** → Pramod Gupta • 2 years ago

Hi Pramod

Generally speaking, handling missing values is a vast topic. However, assuming that your data has

considerable amount of data before the week/month corresponding to the missing value, one simple thing that can be done is: Use the time-ordered data prior to the missing value as training data to fit an ARIMA model, and use this model to forecast the missing values. Now, once we have the forecasted missing values, use the whole data to train a new ARIMA model. Caution here is though that the resultant model is prone to error in the forecasted missing values. Hope that helps. Thanks!

^ | v • Reply • Share ›



**Naga Karu** • 2 years ago

Hi Anirudh

Thank you for that extremely clear write up on adding the fourier seasonality terms. I was wondering if you could please kindly share the R code for the entire analysis? I am having trouble figuring out how you're bringing the underlying data into xreg1 and xreg2 (I can only assume NumEvents is the actual data - apologies, new to R here).

If you please share the underlying R code that will be much appreciated.

Thank you once again for the write up

^ | v • Reply • Share ›



**J M** → Naga Karu • 2 years ago

I would also appreciate the full code.

^ | v • Reply • Share ›



**Anirudh Kondaveeti** → Naga Karu • 2 years ago

Hi Naga Karu

Thanks for your comment. The underlying data is in NumEvents as you point out. The code for creating xreg1 and xreg2 and using them to build the model is in the blog itself.

```
xreg1 <- fourier(1: length(NumEvents), i, 24)
xreg2 <- fourier(1: length(NumEvents), j, 24*7)
xtrain <- cbind(xreg1, xreg2)
```

## Related content in this Stream



Data in Aisle 3



Pairing for Product Managers



Getting Started with Spring Cloud Services on Pivotal Application Service



30 Years of Tech Transition

LEARN	BUILD	CONNECT	COMPANY
(HTTPS://PIVOTAL.IO/LEARN)	(HTTPS://PIVOTAL.IO/BUILD)	(HTTPS://PIVOTAL.IO/CONNECT)	(HTTPS://PIVOTAL.IO/ABOUT)
Topics	Tutorials	Advocates	About (https://pivotal.io/about)
(https://pivotal.io/topics)	(https://pivotal.io/platform/pcf-tutorials/getting-started-with-pivotal-cloud-foundry?utm_source=uberflipfootermenu&utm_medium=header&utm_campaign=getting-started-with-pivotal-cloud-foundry)	(https://pivotal.io/advocates)	Careers (https://pivotal.io/careers)
Products & Services	tutorials/getting-started-with-pivotal-cloud-foundry?	#ImAPivot Community	Contact (https://pivotal.io/contact)
(https://pivotal.io/products)	pivotal-cloud-foundry?	(https://pivotal.io/imapivot)	Executive Team
Industries	utm_source=uberflipfootermenu&utm_medium=header&utm_campaign=getting-started-with-pivotal-cloud-foundry	Events (https://pivotal.io/events)	(https://pivotal.io/team)
(https://pivotal.io/industries)	PWS (http://run.pivotal.io)	Product Consultation	Investors
Resources (/resources)	PCF Dev (https://pivotal.io/pcf-dev)	(https://pivotal.io/office-hours)	(https://pivotal.io/investors)
Webinars (/webinars)	dev)	Blog (/)	Legal (https://pivotal.io/legal)
Customers	Documentation	Podcasts (/podcasts)	Locations
(https://pivotal.io/customers)	(http://docs.pivotal.io/)	Social (/social)	(https://pivotal.io/locations)
Training	Downloads	Newsletter	Partners (https://pivotal.io/partners)
(https://pivotal.io/training)	(https://network.pivotal.io/)	(https://pivotal.io/newsletter-subscription)	Press Center
Platform Acceleration Lab	Knowledge Base		(https://pivotal.io/press-center)
(https://pivotal.io/platform-acceleration-lab)	(https://discuss.pivotal.io)		
	Support (https://support.pivotal.io)		
	Pivotal Services Marketplace		
	(https://pivotal.io/platform/services-marketplace)		

Build software people love

SpringOne Platform

Watch the recordings [→](#)

(https://springoneplatform.io)

(https://twitter.com/pivotal)

in

(https://www.linkedin.com/company/31)

(https://medium.com/built-

to-adapt)

(https://www.youtube.com/c/pivotal)

(https://www.instagram.com/pivotalsof

f

(https://www.facebook.com/pivotalsoft