

Seasonal Forecasting using Regression

One of the most common applications of predictive analytics is to forecast time-based data. This analytic uses R's ordinary least squares regression algorithm to fit the best curve that captures the general trend and season variability of numeric data so it can use to predict future values. The result is a regression model in the form

$$Y = b_{Trend}Trend + \sum (b_{Season_i}Season_i) + b_{Intercept}$$

where

Y is a numeric metric (called the Dependent Variable)

Trend is a numeric metric that's an arithmetic sequence of monotonically increasing values

Season_i is a binary indicator metric derived from **Season**, a numeric or string metric that represents each season. Binary indicators have a value of 1 for the i-th season and are 0 for all other seasons. For n seasons, there are n-1 **X_{Season_i}** variables

b_{Trend}, **b_{Season_i}**, and **b_{Intercept}** are coefficients determined by the regression algorithm.

How to Deploy to MicroStrategy:

Prerequisite: Please follow the instructions in the [R Integration Pack User Guide](#) [1] for configuring your MicroStrategy environment with R and that the R Script functions have been installed in your MicroStrategy project(s). This installation includes the SeasonalForecasting.R script which can be found in the RScripts folder where the R Integration Pack is installed (usually C:\Program Files (x86)\R Integration Pack\RScripts).

- 1) The R Integration Pack installation includes the SeasonalForecasting.R script which can be found in the RScripts folder where the R Integration Pack is installed (usually C:\Program Files (x86)\R Integration Pack\RScripts).
- 2) From the R console, run the SeasonalForecasting.R script to verify the script runs correctly. For details, see the "[Running from the R Console](#)" section below.
- 3) Cut-and-paste the metric expression below in any metric editor. This metric returns the forecast of seasonal values.
- 4) Use the new metric in reports, dashboards and documents.

Metric Expression:

- 1) **Forecast:** Returns the forecasted value as a number

```
RScript<_RScriptFile="SeasonalForecasting.R", _InputNames="target, trend, season",
StringParam9="SeasonalForecast">(target, trend, season)
```

Analytic Signature:

Inputs	Parameters	Outputs	Metric Expression
1) Target Numeric <input type="button" value="v"/> Vector <input type="button" value="v"/> 2) Trend Numeric <input type="button" value="v"/> Vector <input type="button" value="v"/> 3) Season Default <input type="button" value="v"/> Vector <input type="button" value="v"/>	FileName StringParam9 <input type="button" value="v"/> {Default}	Forecast (default) Numeric <input type="button" value="v"/> Vector <input type="button" value="v"/>	<input checked="" type="checkbox"/> Nulls Allowed <input checked="" type="checkbox"/> Check Input Count <input checked="" type="checkbox"/> Enable Sort By {Default=First Input} <input type="checkbox"/> Specify Working Directory E:\R_Off-the-Shelf Output Variable Forecast (default) <input type="button" value="v"/> Metric Expression <input type="button" value="Copy to Clipboard"/> RScript<_RScriptFile="SeasonalForecasting.R", _InputNames="Target, Trend, Season">(Target, Trend, Season)

Input:

Target: Value to forecast (dependent variable). All non-null values are used by the curve fitting algorithms, rows with null values for the Target are excluded are excluded from the machine learning algorithm (these are typically future values to be forecast).

Trend: Numeric metric that's an arithmetic sequence of monotonically increasing values (independent variable).

Season: A numeric or string metric that represents each season (independent variable).

Parameters:

ModelName: Uses StringParam9 with a default of "SeasonalForecast". This parameter specifies the file name (and optionally the path) to use for the files saved by the script. Please note the R Script automatically appends the appropriate file extensions to this file name.

Output:

Forecast: Forecasted values

Additional Results Generated by the R Script:

Two files are stored in the RScripts folder, or the working directory if one is specified by the `_WorkingDir` function parameter:

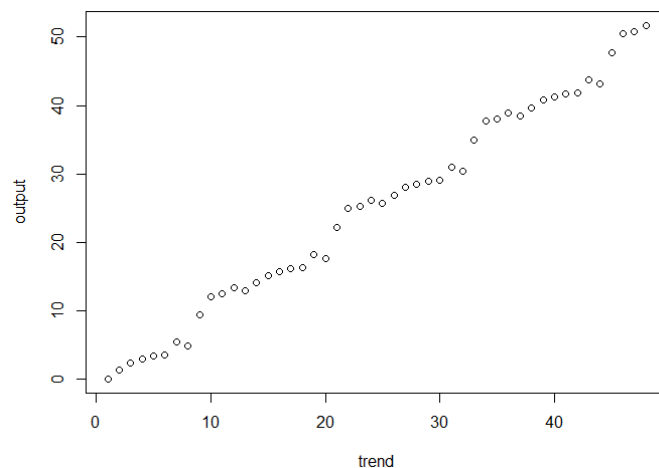
PMML Model: An XML representation of the regression model using the PMML standard. This model can be imported into a predictive metric using MicroStrategy Desktop for deployment to other reports and documents.

.Rdata File: This file persists the state of several objects from the R environment for later inspection and analysis, including `df` (a data frame containing the data used to train the model), `model` (the regression model object) and `output` (the expected value).

Running from the R Console:

In addition to processing data from MicroStrategy during execution of a report or dashboard, the R script is also configured to run from the R console. Running the script for the R Console verifies that the script is functioning as expected, a good practice when initially deploying this analytic to a new system (for more details, see “Configuring dual execution modes” in [1]).

When run from the R Console, if the script is executing properly, a “Success!” message will appear in the console. Also, executing the command `plot(Trend, Forecast)` from the R Console will generate an image similar to this one where the data points trend up and to the right, if the script is executing properly:



If a “Success!” message does not appear, then please note the error in order to take appropriate action. For common pitfalls, please consult the [Troubleshooting](#) section below.

Troubleshooting:

This section covers certain situations you might encounter but it’s not intended as a comprehensive list of possible errors.

- 1) If an error occurs, the report may fail with an error message, or nulls returned as the output. In these cases, please refer to the `RScriptErrors.log` file generated for further guidance and the `DSSErrors.log`. Please consult the User Guide [1] and the R documentation for additional guidance.

2) The script will attempt to install one required R package:

pmml: This R package allows the regression model to be output as PMML.

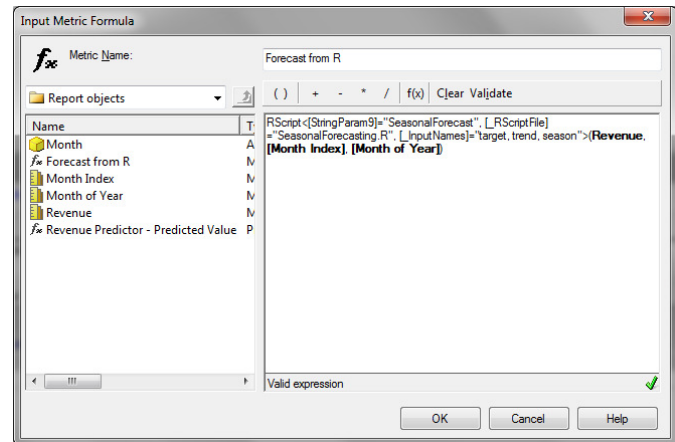
If the package is not successfully installed, you can install using the R console using the command:

```
install.packages("pmml", repos="http://cran.rstudio.com/")
```

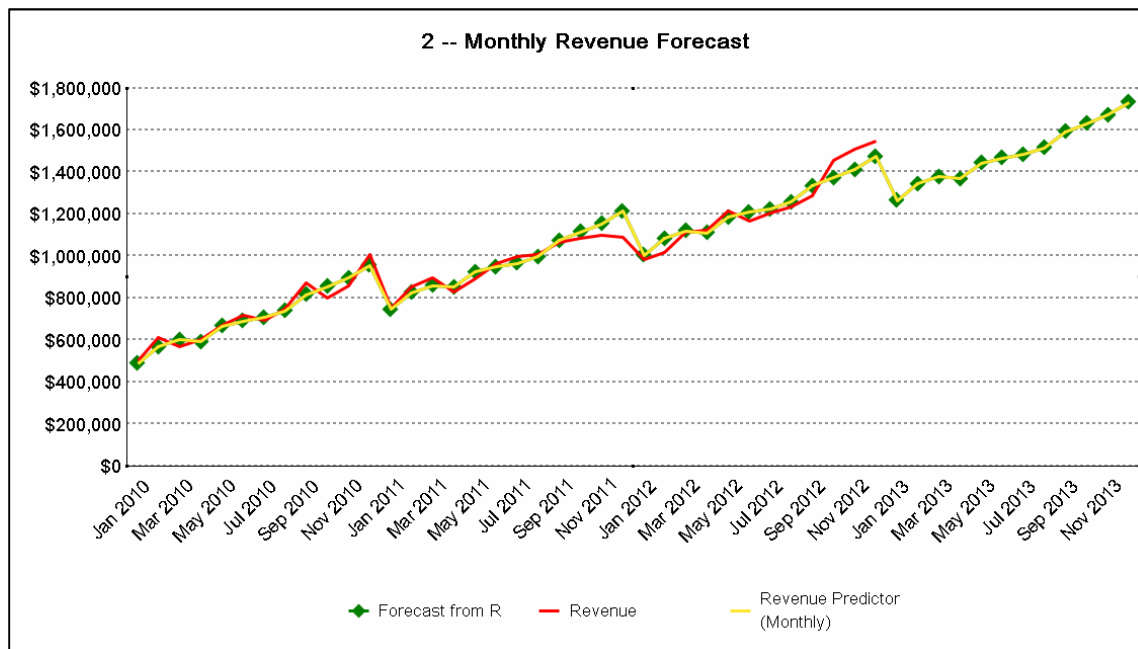
Example:

Using the “2 – Monthly Revenue Forecast” report from the MicroStrategy Tutorial (see Tutorial\Public Objects\Reports\MicroStrategy Platform Capabilities\MicroStrategy Data Mining Services\Linear Regression\Monthly), insert a new metric called Forecast from R (or another name you choose), paste in the Seasonal Forecasting metric expression and assign the inputs as follows:

- Target = Revenue
- Trend = Month Index
- Season = Month of Year



Click ok and you should see that the new metric's results from R exactly match those from the Revenue Forecast (Monthly) predictive metric:



References:

- 1) MicroStrategy R Integration Pack User Guide:
<https://rintegrationpack.codeplex.com/documentation>