Stepwise Logistic Regression

Stepwise Logistic Regression is a variant on classical Logistic Regression in which variables are only included in the model if they have a significant effect on the variable we are trying to predict.

For instance, when trying to predict whether a customer will respond to an offer, we can try to include many variables such as the type of offer, gender of the customer, frequency with which the customer makes purchases, and other demographic and psychographic variables. But rather than using all these variables to try to generate a prediction, we only want to include those that have a demonstrated effect on the customer's decision. Stepwise Logistic Regression automatically determines which variables from our dataset have an effect on a customer's response and returns predictions based on those variables.

The Stepwise Logistic Regression script uses backward elimination based on the Akaike Information Criterion (AIC) [3]. By using "backward elimination", all variables are initially included in the model and are removed one by one until only the significant variables remain.

For more detailed information on Stepwise Regression, please consult the Stepwise Regression Wikipedia page [4].

This R Script has two functional modes:

- The user opts for stepwise logistic regression.
- The user opts not to perform stepwise logistic regression. In such a case, standard logistic regression is performed and all independent variables are included in the model.

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).

1) Download the StepwiseLogistic.R file from the R Script Shelf [2] and place it 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 StepwiseLogistic.R script to verify the script runs correctly. For details, see the "**Running from the R Console**" section below.
3) Cut-and-paste the appropriate metric expression below in any MicroStrategy metric editor. Map the arguments to the appropriate MicroStrategy metrics. A description of each output is shown below.
4) Use the new metric in reports, dashboards and documents.

Metric Expressions:

The metric expressions shown here assume that the StepwiseLogistic.R file has been downloaded to the server. If using the URL-based approach where the StepwiseLogistic.R file is accessed directly via URL, please consult the R Script Shelf [2].

1) **Probability:** Returns the probability of an event occurring:

If using R Integration Pack V 1.0 with 27 pre-defined parameters:

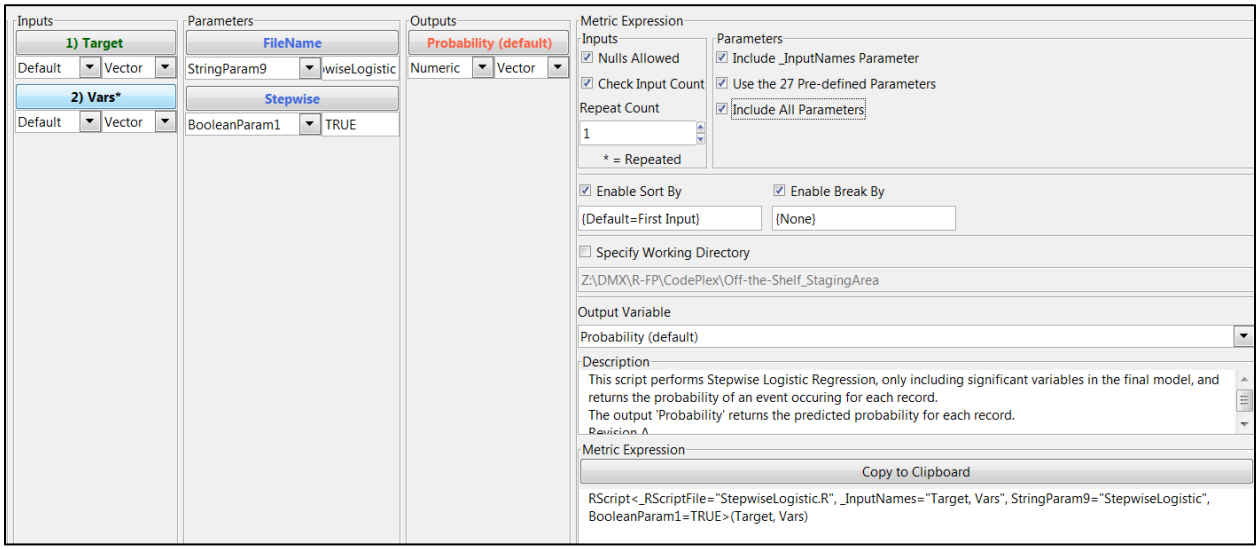
```
RScript<_RScriptFile="StepwiseLogistic.R", _InputNames="Target, Vars",
StringParam9="", BooleanParam1=TRUE>(Target, Vars)
```

© 2015 MicroStrategy, Inc.

If using R Integration Pack V 2.0 with named parameters:

```
RScript<_RScriptFile="StepwiseLogistic.R", _InputNames="Target, Vars",
_Params="FileName='', Stepwise=TRUE">(Target, Vars)
```

Analytic Signature:



Inputs:

Target: The variable that we are trying to predict.

Vars: The independent variables that are considered when creating the model and generating the prediction. Since the Vars argument is a repeated input, it can be mapped to any number of MicroStrategy metrics. In this way, we enable our algorithm to consider any number of variables when generating predictions.

Note: **Repeated inputs must all be the same type (i.e. all variables must be numeric or all variables must be strings)**

Note: **The _InputNames parameter must be the same length as the number of inputs. For instance, if there are 5 independent variables, then the _InputNames argument must reflect all six inputs (Target + 5 independent variables). Spaces will be automatically removed from the _InputNames.**

Parameters:

FileName: Uses StringParam9 with a default of "StepwiseLogistic". This parameter specifies the name of both the .Rdata file used to persist relevant objects from the R environment such as the model and the data used to create it, allowing for additional inspection and analysis, and the PMML file that gets generated as part of script execution. Please note the R Script automatically appends the ".Rdata" file extension to this file name. If a value of "" is passed in, then no Rdata file is created.

Stepwise: Uses BooleanParam1 with a default of TRUE. If this value is set to TRUE, then stepwise logistic regression will be performed and only the variables with a significant effect will be considered when generating predictions. If this value is set to FALSE, then traditional logistic regression will be performed and all variables will be included in the model.

© 2015 MicroStrategy, Inc.

Outputs:

Probability: A numeric value representing the probability of the event occuring.

Additional Results Generated by the R Script:

Two files are stored in the working directory:

Rdata File: This file persists the state of several objects from the R environment for later inspection, analysis, and reuse, including df (a data frame containing the data read in from MicroStrategy), model (the glm model object), and Probability (the probability of the event occurring for that record).

Pmml File: This file contains PMML, an XML representation of our logistic regression model. This file can be imported into MicroStrategy, which will create predictive metrics in MicroStrategy that can be used to score records contained in MicroStrategy.

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. 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 will either fail with an error message, or return nulls 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) If Stepwise is set to TRUE, then the script will attempt to install the required R package.
   MASS: This R package contains functions that support the execution of Stepwise Regression.
   If the package is not successfully installed, you can install using the R console using the command:
   ```
   install.packages("MASS", repos="http://cran.rstudio.com/")
   ```

3) If a mix of string and numeric variables is passed in to the Vars argument, the report will fail with an error message indicating that a variable with unexpected type was passed in. This can be remedied by using all strings or all numeric variables corresponding to the Vars argument.

4) If an error message similar to "Duplicated name X in data frame" appears, that means that the _InputNames parameter does not have length corresponding to the number of inputs. Please ensure that the number of names being passed in to _InputNames matches the number of inputs.

© 2015 MicroStrategy, Inc.

Example (using MicroStrategy Tutorial):

In order to combat an escalation in the number of customers who are churning, or leaving the company in the midst of their contracts, a large Telco organization wants to create and deploy a Logistic Regression model that predicts which customers are most likely to churn. By proactively identifying at-risk customers, the organization can implement different measures aimed at assuaging these disgruntled customers.

When creating the model, the organization would like to consider the effect that the following variables have on whether a customer is likely to churn or not. However, only variables that have a demonstrated effect on churning should be retained in the model.

a) Average minutes used during off-peak hours.
b) Average minutes used during peak hours.
c) Number of Dropped Calls.
d) Number of months that customer has been active.
e) Number of Helpdesk Calls placed.
f) Remaining monetary value of customer contract.

To do so, the retailer employs stepwise logistic regression to determine which variables have a significant effect on profit. This can be done by creating and saving a report with the following design:

Rows (Attributes):

1) **Customer**

Columns (All Metrics which are found in the folder **Public Objects -> Reports -> MicroStrategy Platform Capabilities -> MicroStrategy Data Mining Services -> Support Objects-> Telco Metrics**):

2) **TelcoChurn**
3) **AvgMinOffPeak**
4) **AvgMinPeak**
5) **DroppedCalls**
6) **ActiveMonths**
7) **HelpdeskCalls**
8) **RemainingValue**

Filter (Filter found in **Public Objects -> Reports -> MicroStrategy Platform Capabilities -> MicroStrategy Data Mining Services -> Sampling)**:

9) **Add n-th record sample filter**

Run this report with a prompt answer of **7**.

Now, insert a derived metric named “Churn Probability” with the following definition:

If using R Integration Pack V 1.0 with 27 pre-defined parameters:

```
RScript<[BooleanParam1]=True, [StringParam9]="StepwiseLogistic",
[_RScriptFile]="StepwiseLogistic.r", [_InputNames]="TelcoChurn,
AvgMinOffPeak,AvgMinPeak,DroppedCalls,ActiveMonths,HelpdeskCalls,RemainingValue">(TelcoCh
urn,AvgMinOffPeak,AvgMinPeak,DroppedCalls,ActiveMonths,HelpdeskCalls,RemainingValue)
```

If using R Integration Pack V 2.0 with named parameters:

```
RScript<[_RScriptFile]="StepwiseLogistic.r", [_InputNames]="TelcoChurn,
AvgMinOffPeak,AvgMinPeak,DroppedCalls,ActiveMonths,HelpdeskCalls,RemainingValue",
[_Params]="FileName='StepwiseLogistic', Stepwise=TRUE">(TelcoChurn, AvgMinOffPeak,
AvgMinPeak, DroppedCalls, ActiveMonths, HelpdeskCalls, RemainingValue)
```

© 2015 MicroStrategy, Inc.

You should obtain the following results when sorting on Churn Probability descending and formatting Churn Probability as a percentage:

Customer	Metrics	Churn Probability	TelcoChurn	AvgMinPeak	AvgMinOffPeak	DroppedCalls	HelpdeskCalls	ActiveMonths	RemainingValue
Tomsick	Hammond	88.64%	0	5,077	3,893	7	6	52	$0
Sanchez	Carol	85.83%	1	5,924	3,625	1	8	32	$10,458
Aivazian	Greg	85.18%	0	5,662	3,944	1	9	11	$8,589
Abrams	Fredrick	84.56%	1	5,650	3,716	4	7	9	$9,461
Robare	Henry	82.47%	0	4,406	6,515	4	8	9	$7,802
Harrell	Sidsel	80.04%	1	4,239	5,609	6	6	13	$7,602
Hermann	Kyle	79.19%	1	4,223	3,375	3	7	27	$0
Rizzo	Marek	77.76%	0	3,373	8,667	5	6	40	$5,281
Steinbinder	Marcelino	77.20%	1	5,731	2,320	1	6	14	$4,664
Schrock	Desree	76.98%	1	4,463	2,335	4	6	12	$3,718
Spence	Gerriet	76.39%	1	5,910	5,255	5	1	62	$8,556
Lange	Monte	74.31%	1	3,909	2,424	5	5	22	$503
Beavers	Rurik	72.43%	1	3,545	5,053	6	3	55	$9,090
Borg	Derrin	72.41%	1	4,236	3,380	6	4	8	$6,593
Reber	Andra	72.29%	1	4,671	4,351	6	3	16	$4,703
Reeve	Catarina	71.74%	1	3,787	4,511	3	5	39	$4,475
Moorby	Mitzi	71.67%	1	4,719	4,141	3	3	54	$0
Soliveras	Francis	68.59%	1	5,867	3,592	2	2	35	$8,351
Invie	Elmer	67.90%	0	3,950	4,913	0	6	31	$9,655
Acuna	Andre	67.69%	0	5,202	6,737	5	2	9	$9,711
Buechner	Shareen	66.58%	1	4,416	2,615	6	2	15	$3,050
Hubbard	Adrian	65.30%	1	4,072	2,196	2	5	7	$4,140
Nicholson	Gwin	63.91%	1	5,346	5,031	6	0	16	$6,057
Domini	Wilbur	63.13%	0	4,027	5,918	4	3	15	$6,327
Andrade	Florence	63.03%	0	4,130	4,293	5	2	20	$2,052

Now, navigate to your default R scripts folder (usually C:/Program Files (x86)/R Integration Pack/R Scripts) and locate the StepwiseLogistic.Rdata file that was created. Double click on the file and open it with an R program. Once it is open enter the command "**model**". The result should be:

```
> model

Call:  glm(formula = TelcoChurn ~ AvgMinPeak + DroppedCalls + ActiveMonths +
    HelpdeskCalls, family = "binomial", data = df)

Coefficients:
  (Intercept)     AvgMinPeak   DroppedCalls   ActiveMonths  HelpdeskCalls
   -1.7369735      0.0002754      0.1176709      0.0081297      0.1910426

Degrees of Freedom: 1427 Total (i.e. Null);  1423 Residual
Null Deviance:      1642
Residual Deviance: 1569         AIC: 1579
```

Observe that the stepwise logistic regression model found that just four of the seven independent variables were significant. The stepwise procedure determined that **AvgMinOffPeak** and **RemainingValue** did not have a significant effect on whether a customer decides to churn.

References:

1) MicroStrategy R Integration Pack User Guide: https://rintegrationpack.codeplex.com/documentation
2) R Analytic Shelf: https://rintegrationpack.codeplex.com/wikipage?title=R%20Script%20%22Shelf%22&referringTitle=Home#
3) Akaike Information Criterion Wikipedia Page: http://en.wikipedia.org/wiki/Akaike_information_criterion
4) Stepwise Regression Wikipedia Page: http://en.wikipedia.org/wiki/Stepwise_regression

© 2015 MicroStrategy, Inc.