Poisson Regression

Updated: July 13, 2015

Creates a regression model that assumes data has a Poisson distribution

Category: Machine Learning / Initialize Model / Regression (https://msdn.microsoft.com/en-us/library/azure/dn905922.aspx)

Module Overview

You can use the **Poisson Regression** module to create a regression model that can be used to predict numeric values, typically counts. You should use this model only if the values you are trying to predict fit these conditions:

- The response variable has a Poisson distribution (http://en.wikipedia.org/wiki/Poisson_distribution).
- Counts cannot be negative. The method will fail outright if you attempt to use it with negative labels.
- Given that a Poisson distribution is a discrete distribution, it is not meaningful to use this method with non-whole numbers.
- If your target isn't a count, Poisson regression is probably not an appropriate method.

After you have configured the model, you must train the model using a dataset that contains an example of the value you want to predict (the numeric label), using either Train Model (https://msdn.microsoft.com/en-us/library/azure/dn906044.aspx) or Sweep Parameters (https://msdn.microsoft.com/en-us/library/azure/dn905810.aspx). The trained model can then be used to make predictions using Score Model (https://msdn.microsoft.com/en-us/library/azure/dn905995.aspx).

Understanding Poisson Regression

Poisson regression is a special type of regression analysis that is typically used to model counts. For example, Poisson regression would be useful in these scenarios:

- Modeling the number of colds associated with airplane flights
- Estimating the number of emergency service calls during an event, or estimating customer inquiries relating to a promotion
- Creating contingency tables

Because the response variable has a Poisson distribution, the underlying assumptions about probability distribution are different from least-squares regression and Poisson models must be interpreted differently from other regression models.

See the Technical Notes section for more details of the formulas.

How to Configure a Poisson Regression Model

1. Specify how you want the model to be trained, by setting the **Create trainer mode** option.

    - **Single Parameter**

      If you know how you want to configure the model, you can provide a specific set of values as arguments. You might have learned these values by experimentation or received them as guidance.

    - **Parameter Range**

      If you are not sure of the best parameters, you can find the optimal parameters by specifying multiple values and using a parameter sweep to find the optimal configuration.

      Sweep Parameters (https://msdn.microsoft.com/en-us/library/azure/dn905810.aspx) will iterate over all possible combinations of the settings you provided and determine the combination of settings that produces the optimal results. You can use the model trained using those parameters, or you can make a note of the parameter settings to use when configuring a learner.

2. Continue to set other parameters that affect the behavior of the regression model. See the section for details.Bookmark link 'bkmk_Options' is broken in topic '{"project_id":"37f8d135-1f1d-4e57-9b7d-b084770c6bf5","entity_id":"80e21b9d-3827-40d8-b733-b53148becbc2","entity_type":"Article","locale":"en-US"}'. Rebuilding the topic '{"project_id":"37f8d135-1f1d-4e57-9b7d-b084770c6bf5","entity_id":"80e21b9d-3827-40d8-b733-b53148becbc2","entity_type":"Article","locale":"en-US"}' may solve the problem.

3. If you set the **Create trainer mode** option to **Single Parameter**, add a tagged dataset and train the model by using the Train Model (https://msdn.microsoft.com/en-us/library/azure/dn906044.aspx) module.

   If you set the **Create trainer mode** option to **Parameter Range**, add a tagged dataset and train the model using Sweep Parameters (https://msdn.microsoft.com/en-us/library/azure/dn905810.aspx).

Options

You can use these parameters to affect the model training process:

Create trainer mode

Choose the method used for configuring and training the model:

- ***Single Parameter***

  Select this option to configure and train the model with a single set of parameter values that you supply.

  If you choose this option, you should train the model by using the Train Model (https://msdn.microsoft.com/en-us/library/azure/dn906044.aspx) module.

- ***Parameter Range***

  Select this option to use the Range Builder and specify a range of possible values. You then train the model using a parameter sweep, to find the optimum configuration.

⚠ **Warning**

- If you pass a parameter range to Train Model (https://msdn.microsoft.com/en-us/library/azure/dn906044.aspx), it will use only the first value in the parameter range list.
- If you pass a single set of parameter values to the Sweep Parameters (https://msdn.microsoft.com/en-us/library/azure/dn905810.aspx) module, when it expects a range of settings for each parameter, it ignores the values and using the default values for the learner.
- If you select the **Parameter Range** option and enter a single value for any parameter, that single value you specified will be used throughout the sweep, even if other parameters change across a range of values.

Optimization tolerance

Type a value that defines the tolerance interval during optimization.

The lower the value, the slower and more accurate the fitting.

L1 regularization weight, L2 regularization weight

Type values to use for L1 and L2 regularization.

Regularization adds constraints to the algorithm regarding aspects of the model that are independent of the training data. Regularization is commonly used to avoid overfitting. In this module, you can apply a combination of L1 and L2 regularizations.

- L1 regularization is useful if the goal is to have a model that is as sparse as possible.

  L1 regularization is done by subtracting the L1 weight of the weight vector from the loss expression that the learner is trying to minimize. The L1 norm is a good approximation to the L0 norm, which is the number of non-zero coordinates.

- L2 regularization prevents any single coordinate in the weight vector from growing too much in magnitude. L2 regularization is useful if the goal is to have a model with small overall weights.

By combining L1 and L2 regularizations, you can impose a penalty on the magnitude of the parameter values. The learner tries to minimize the penalty, in a tradeoff with minimizing the loss.

Memory size for L-BFGS

Specify the amount of memory to reserve for model fitting and optimization.

L-BFGS is a technique used for optimization. The technique is based on the Broyden–Fletcher–Goldfarb–Shanno (BFGS) algorithm, and specifies a limited amount of memory (L) to compute the next step direction.

By changing this parameter, you can affect the number of past positions and gradients that are stored for computation of the next step.

Recommendations

It is recommended that you use Normalize Data (https://msdn.microsoft.com/en-us/library/azure/dn905838.aspx) to normalize the input dataset before using it to train the regressor.

Examples

For examples of how Poisson regression is used in machine learning, see these sample experiments in the Model Gallery (http://gallery.azureml.net/).

- Sample 6: Train, Test, Evaluate for Regression: Auto Imports Dataset (https://gallery.azureml.net/Experiment/670fbfc40c4f44438bfe72e47432ae7a): This experiment compares the outcomes of two algorithms: **Poisson Regression** and Decision Forest Regression (https://msdn.microsoft.com/en-us/library/azure/dn905862.aspx).

- The Preventive Maintenance (https://gallery.azureml.net/Experiment/a677f8eececf40eaa158699a2b27e3c8) sample is an extended walkthrough that uses **Poisson Regression** to assess the severity of failures predicted by a decision forest model.

Technical Notes

Poisson regression is classically used to model count data (the canonical example being calls to a switchboard in a given day), assuming that the label has a Poisson distribution.

For this algorithm, we assume that an unknown function, denoted *Y*, has a Poisson distribution. The Poisson distribution is defined as follows:

Given the instance $x = (x_0, \ldots, x_{d-1})$, for every $k=0,1, \ldots$, the probability that the value of the instance is *k* is:

Given the set of training examples, the algorithm tries to find the optimal values for $\theta_0, \ldots,\theta_{D-1}$ by trying to maximize the log likelihood of the parameters. The likelihood of the parameters $\theta_0, \ldots,\theta_{D-1}$ is the probability that the training data was sampled from a distribution with these parameters.

The log probability can be viewed as $logp(y = y_i)$

The prediction function outputs the expected value of that parameterized Poisson distribution, specifically: $f_{w,b}(x) = E[Y|x] = e^{w^{T}x+b}$.

For more information, see the Wikipedia entry for Poisson regression.

Module Parameters

Name	Range	Type	Default	Description
Optimization tolerance	>=double.Epsilon	Float	0.0000001	Specify a tolerance value for optimization convergence. The lower the value, the slower and more accurate the fitting.
L1 regularization weight	>=0.0	Float	1.0	Specify the L1 regularization weight. Use a non-zero value to avoid overfitting the model.
L2 regularization weight	>=0.0	Float	1.0	Specify the L2 regularization weight. Use a non-zero value to avoid overfitting the model.
Memory size for L-BFGS	>=1	Integer	20	Indicate how much memory (in MB) to use for the L-BFGS optimizer. With less memory, training is faster but less accurate the training.
	any	Integer		

Random number seed				Type a value to seed the random number generator used by the model. Leave blank for default.
Allow unknown categorical levels	any	Boolean	true	Indicate whether an additional level should be created for each categorical column. Any levels in the test dataset not available in the training dataset are mapped to this additional level.

Outputs

Name	**Type**	**Description**
Untrained model	ILearner interface (https://msdn.microsoft.com/en-us/library/azure/dn905938.aspx)	An untrained regression model

See Also

Machine Learning / Initialize Model / Regression (https://msdn.microsoft.com/en-us/library/azure/dn905922.aspx)
A-Z List of Machine Learning Studio Modules (https://msdn.microsoft.com/en-us/library/azure/dn906033.aspx)