Ordinal Regression

Updated: July 18, 2015

Creates an ordinal regression model

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

Module Overview

You can use the **Ordinal Regression** module to create a regression model that can be used to predict ranked values. An example of ranked values might be survey responses that capture user's preferred brands on a 1 to 5 scale, or the order of finishers in a race.

Understanding Ordinal Regression

Ordinal regression is used when the label or target column contains numbers, but the numbers represent a ranking or order rather than a numeric measurement. Predicting ordinal numbers requires a different algorithm than predicting the values of numbers on a continuous scale, because the numbers assigned to represent rank order do not have intrinsic scale. For example, to predict students' test scores, you would use a standard regression model, because students' test scores vary on a continuous scale and can be measured. However, to predict their class ranking, you would use an ordinal regression model.

For more information about the research behind this algorithm, see this paper (downloadable PDF): http://papers.nips.cc/paper/3125-ordinal-regression-by-extended-binary-classification.pdf (http://papers.nips.cc/paper/3125-ordinal-regression-by-extended-binary-classification.pdf)

Configuring an Ordinal Regression Model

1. Add the **Ordinal Regression Model** module to your experiment.

2. Add one of the binary classification models and connect it as an input to the **Ordinal Regression Model** module.

Warning

To solve a ranking problem as a series of classification problems, the algorithm creates a series of extended training examples and trains against that extended set. This operation can be computationally expensive.

3. You don't need to set additional parameters on the **Ordinal Regression Model** itself – the algorithm has been pre-configured with the most effective parameters for solving a ranking problem.

4. Add a Train Model (https://msdn.microsoft.com/en-us/library/azure/dn906044.aspx) module and connect the **Ordinal Regression Model**. For the training dataset, connect a dataset that contains examples of the ordinal values you want to predict.

Warning

For purposes of processing, the ranks are assumed to have the order *1 to K*, where 1 is lowest rank, and *K* is the highest rank. However, the Train Model (https://msdn.microsoft.com/en-us/library/azure/dn906044.aspx) module will work even if the semantics of your scale are reversed. For example, if in your original survey, 1 was the highest score and 5 is the lowest, it does not affect the processing of the model.

5. In the Train Model (https://msdn.microsoft.com/en-us/library/azure/dn906044.aspx) module, select the column that contains the rank values.

   The rank values must be numerical values, but they need not be integers or positive numbers, as long as they represent a sequence.

6. After you train the model, you can connect it to the Score Model (https://msdn.microsoft.com/en-us/library/azure/dn905995.aspx) module to make predictions.

   Alternatively, the untrained model can be passed to Cross-Validate Model (https://msdn.microsoft.com/en-us/library/azure/dn905852.aspx) for cross-validation against a labeled data set.

Examples

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

- In Step 2C of the Predictive Maintenance (https://gallery.azureml.net/Experiment/68b4a27dc53d426e902025e77a393702) sample, **Ordinal Regression** is used to rank values output by a classification model, on the assumption that the value reflects the severity of the failure classification.

Technical Notes

You can use any numeric column as the target of an ordinal regression model, but in practice you should use only data that represents some sort of order or ranking. The intervals between ranks are assumed to be unknown and the size of the interval does not matter to the model; however, the model assumes that the sequence of ranks follows the natural ordering of numbers.

The model does not assign any meaning to a particular scale. In other words, you might have one model that assumes 1 is best and 10 is the worst, and another model that assumes 10 is the best and 1 is the worst.

The ordinal regression algorithm used in this learner is implemented by extended binary classification, as described by the paper titled *Ordinal Regression by Extended Binary Classification*, by Ling Li and Hsuan-Tien Lin, in NIPS 2006.

Algorithm

The training set *(X,Y)* consists of input vectors *x* and labels *y*, which are ranks *{1,2, … , K}*

The ranks have been assumed to have the order *1,2,3, … , K* where *K* is the best or highest rank and 1 is the lowest or the worst rank.

The crux of the algorithm lies in modifying the given input features *X* and labels *Y* to extended examples, and then using a binary classifier to solve the ordinal regression problem.

The binary classifier is trained to give a yes/no answer to the question, "Is the rank greater than r?"

For example, for each case in the training set there are *K-1* extended examples, and the maximum observed rank is *K*. The extended features are formed by appending the i^{th} row of a *K-1 x K-1* identity matrix to the input features for all *i*. The labels are given *+1* for the first *r-1* rows if its rank is *r* and *-1* to the rest.

Sample Calculations

Let x^1 be the training feature whose rank is 3, where the maximum observed rank is 5. The extended examples corresponding to this feature are as follows:

Case	Test	Resulting label
$X_1 1000$	Is rank greater than 1?	Yes, therefore: *+1*
$X_1 0100$	Is rank greater than 2?	Yes, therefore:

		+1
$X_1 0010$	Is rank greater than 3?	No, therefore no additional feature
$X_1 0001$	Is rank greater than 4?	No, therefore no additional feature

Expected Inputs

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

Outputs

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