

Two-Class Bayes Point Machine

Updated: July 6, 2015

Creates a Bayes point machine binary classification model

Category: Machine Learning / Initialize Model / Classification (<https://msdn.microsoft.com/en-us/library/azure/dn905808.aspx>)

Module Overview

You can use the **Two-Class Bayes Point Machine** module to create an untrained binary classification model.

After you have configured the model, you must train the model on a labeled dataset using the Train Model (<https://msdn.microsoft.com/en-us/library/azure/dn906044.aspx>) module. You can then pass the trained model 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.

Understanding Bayes Point Machines

The Bayes Point Machine is a Bayesian approach to linear classification. It efficiently approximates the theoretically optimal Bayesian average of linear classifiers (in terms of generalization performance) by choosing one "average" classifier, the Bayes Point. Because the Bayes Point Machine is a Bayesian classification model, it is not prone to overfitting to the training data.

For more information, read the original research paper: Bayes Point Machines (<http://research.microsoft.com/apps/pubs/default.aspx?id=65611>).

This implementation improves on the original algorithm in several ways:

- It uses the expectation propagation message-passing algorithm. For more information, see A family of algorithms for approximate Bayesian inference (<http://go.microsoft.com/fwlink/?LinkId=511015>).
- It does not require parameter sweeping.
- It does not require data to be normalized.

These improvements make the Bayes Point Machine classification model more robust and easier-to-use, and you can bypass the time-consuming step of parameter tuning.

For more information, see Chris Bishop's post on the Microsoft Machine Learning blog: Embracing Uncertainty - Probabilistic Inference (<http://blogs.technet.com/b/machinelearning/archive/2014/10/30/embracing-uncertainty-probabilistic-inference.aspx>),

How to Configure the Two-Class Bayes Point Machine

You can use the following parameters to customize the behavior of the Bayes Point Machine algorithm:

Number of training iterations

Type a number to specify how often the message-passing algorithm iterates over the training data.

The higher the number of training iterations, the more accurate the predictions; however, training will be slower.

For most datasets, the default setting of 30 training iterations is sufficient for the algorithm to make accurate predictions. Sometimes accurate predictions can be made by using fewer iterations. For datasets with highly correlated features, you might benefit from more training iterations. Typically, the number of iterations should be set to a value in the range 5 – 100.

Include bias

Select this option if you want a constant feature or bias to be added to each instance in training and prediction.

Including a bias is necessary when the data does not already contain a constant feature.

Allow unknown values

Select this option to create a group for unknown values. If you deselect it, the model can accept only the values that are contained in the training data. In the former case, the model might be less precise for known values, but it can provide better predictions for new (unknown) values.

Examples

To see how the Bayes Point Machine is used in machine learning, see these sample experiments in the Model Gallery (<http://gallery.azureml.net/>):

- The Compare Binary Classifiers (<http://go.microsoft.com/fwlink/?LinkId=525729>) sample demonstrates the use of various two-class classifiers.

Module Parameters

Name	Range	Type	Default	Description
Number of training iterations	>=1	Integer	30	Specify the number of iterations to use when training
Include bias	Any	Boolean	True	Indicate whether a constant feature or bias should be added to each instance
Allow unknown values in categorical features	Any	Boolean	True	If True, creates an additional level for each categorical column. Any levels in the test dataset that are not available in the training dataset are mapped to this additional level.

Output

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

See Also

Machine Learning / Initialize Model / Classification (<https://msdn.microsoft.com/en-us/library/azure/dn905808.aspx>)

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