Documentation: MIP Function — Predictive Clustering Trees

The predictive clustering tree algorithm for predicting structured target variables.

Metadata

Category	MIP function, data analysis algorithm
Maintainers	Martin Breskvar Bernard Ženko
Homepage	http://source.ijs.si/hbp/clus/wikis/home
Documentation	http://source.ijs.si/hbp/mipfunctions/raw/master/doc/r-clus-trees.pdf http://source.ijs.si/hbp/clus/wikis/documentation
Support	http://source.ijs.si/hbp/clus/issues
Source Code	http://source.ijs.si/hbp/clus.git
License	GNU GPL
Current Version	2.12
All Versions	2.12

Description

Predictive clustering combines aspects from both predictive modeling and clustering. Predictive clustering trees (PCTs) partition the set of examples into subsets in which examples have similar values of the target variable, while clustering produces subsets in which examples have similar values of the descriptive variables. The task of predictive clustering is to find clusters of examples which have similar values of both the target and the descriptive variables.

While most decision tree learners induce classification or regression trees, PCTs generalize this approach and represent trees that are interpreted as cluster hierarchies. Depending on the learning task at hand, different goal criteria are to be optimized while creating the clusters, and different heuristics will be suitable to achieve this. Classification and regression trees are special cases of PCTs, and by choosing the right parameter settings PCTs can closely mimic the behavior of tree learners such as CART or C4.5. However, its applicability goes well beyond classical classification or regression tasks: PCTs have been successfully applied to many different tasks including multi-task learning (multi-target classification and regression), structured output learning, multi-label classification, hierarchical classification, and time series prediction. Next to these supervised learning tasks, PCTs are also applicable to semi-supervised learning, subgroup discovery, and clustering.

This MIP function implements the PCT algorithm.

The development of the algorithm and its implementation was not paid by HBP.

References

H. Blockeel, L. De Raedt, and J. Ramon. Top-down induction of clustering trees. In J. W. Shavlik, editor, Proceedings of the Fifteenth International Conference on Machine Learning (ICML 98), pages 55–63, San Francisco, CA, USA, July 1998. Morgan Kaufmann.

Usage

Input: A set of examples with known values for descriptive and target variables.

Parameters: Descriptive variables, target variables, minimum number of examples in each leaf.

Output: A predictive model written as a combination of a linear equation and decision rules.

Example

We can illustrate the use of the algorithm on the ADNI data.

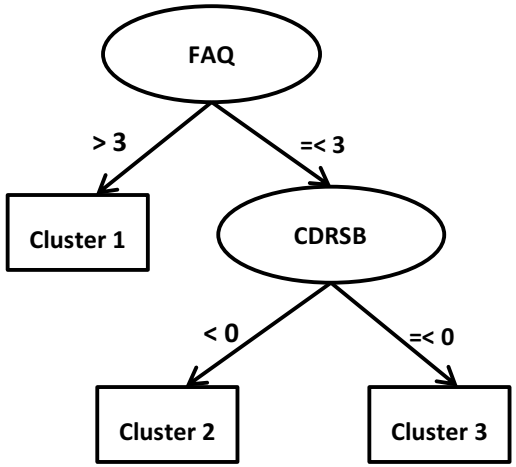
Task:

Given a data set consisting of examples (i.e., patients) described in terms of several descriptive variables and labelled with several target variables we want to learn a PCT that will predict the values of target variables from the values of descriptive variables. In addition, we want to identify groups of patients that are similar in terms of all the target variables and desribe them in terms of descriptive variables.

Input data:

Given a subset of ADNI data with 916 patients described with 34 different variables, collected at baseline evaluation, we split the variables in two groups. In the first group we have descriptive variables: APOE4, Ventricles, Hippocampus, WholeBrain, Entorhinal, Fusiform, MidTemp, ICV, FDG, AV45, CDRSB, ADAS13, MMSE, RAVLT_immediate, RAVLT_learning, RAVLT_forgetting, RAVLT_perc_forgetting, FAQ, MOCA. In the second group we have the target variables: EcogPtMem, EcogPtLang, EcogPtvisspat, EcogPtPlan, EcogPtOrgan, EcogPtDivatt, EcogPtTotal, EcogSPMem, EcogSPLang, EcogSPvisspat, EcogSPPlan, EcogSPOrgan, EcogSPDivatt, EcogSPTotal. We set the minimum number of examples in each cluster in a way to get three leafs or clusters.

Example output:



Cluster 1: (2.4, 1.9, 1.6, 1.7, 1.7, 2.0, 1.9, 3.1, 2.3, 2.2, 2.4, 2.6, 2.7, 2.6) # 264
Cluster 2: (2.2, 1.8, 1.4, 1.4, 1.5, 1.9, 1.7, 1.9, 1.5, 1.3, 1.4, 1.4, 1.7, 1.5) # 389
Cluster 3: (1.7, 1.4, 1.2, 1.2, 1.3, 1.5, 1.4, 1.3, 1.1, 1.1, 1.1, 1.2, 1.3, 1.2) # 263

The resulting model is a predictive clustering tree with three leafs. Each of the leafs can be interpreted as a cluster of examples that is described with associated conditions. The three clusters contain 264, 389 and 263 patients, respectively.