

Datenmodell

Der folgende Abschnitt beschreibt das Datenmodell hinter der graphischen Darstellung des Entscheidungsbaums. Auf das Datenmodell der GUI wird hierbei ganz bewusst verzichtet, da dieses stark von der gewählten Programmiersprache und des entsprechenden Frameworks abhängt. Das in Abbildung XX gezeigte Datenmodell beschreibt somit die Klassen der Kernfunktionalität.

Als oberstes in der Datenhierarchie steht einerseits der Baum (Tree) sowie die Tabelle (Table) sie repräsentieren die Darstellung der Daten als Baum bzw. als Tabelle. Der Kern beider Darstellungen sind die Objekte (Object) selbst. So verfügt die Tabelle direkt über eine Liste sämtlicher Objekte. Ein Objekt entspricht dabei einem Datensatz und somit einer Zeile in der Tabelle. Jedes Objekt besitzt bis zu 16 Attribute. Die Abstrakte Klasse Abttribut fast die Klassen BooleanAttribut, DOubleAttribut, IntegerAttribut sowie StringAttribut zu einem gemeinsamen Nenner zusammen. Jeder der Unterklasse implementiert diese abstrakte Klasse. Jeder der Attribut-Klassen verfügt über den Member value. Dieser Wert entspricht bei jedem der Attribut-Klassen einem anderen Datentyp.

Die Tabelle verfügt neben der Liste von Objekten auch noch über eine Liste die alle Attribute. Anhand dieser Attributliste können neue Objekte auf Validität überprüft werden und erstellt werden. Das Hinzufügen von Objekten übernimmt ebenfalls die Tabelle selbst. Sie bietet daneben noch Funktionalitäten zum Ändern und Löschen von Attributen und Werten.

Der Baum besteht prinzipiell aus einem Verweis auf den ersten Knoten (Node) des Baums. Der Baum insgesamt besteht dann aus einzelnen Knoten (Nodes) welche über beliebig viele Kindknoten (ChildNodes) verfügen können und alle in der Liste objects vorhandenen Objekte auf die Kindknoten verteilen ggf. anhand des Split-Wertes (SplitValue). Der NOde besitzt neben den obligatorischen getter und Setter Methoden auch über die Methoden put und makeSubTree(). makeSubTree entält den algorithmus, der den optimalen Entscheidungsbaum anhand der Entropie berechnet. Put hingegen fügt ein übergebenes Objekt den Knoten hinzu. Dieser entscheidet anschließend in welchen Kindknoten dieses Objekt eingefügt werden soll und fügt es diesem hinzu. So wandert ein neu eingegebenes Objekt den Baum bis zu den Blättern hinab.