Accurate Decision Trees for mining high-speed Data Streams

Stefan Heinje

Inhalt

- Ergänzungen zu VFDT
- Umsetzung in VFDTc
  - Information Gain Ratio als Heuristik
  - Numerische Attribute
  - Naive Bayes in Blättern
- Vergleich
- Verbesserungsvorschläge VFDTc
- Fazit
- Quellen

Ergänzungen zu VFDT

- Konkrete Umsetzung als Algorithmus
- Effiziente Speicherung numerischer Attribute
- Bisher ungenutzte Information in Blättern zur Klassifizierung nutzen

Information Gain Ratio als Heuristik

Informationsgewinn = Entropieverlust

Entropie:

$$H(X) = -\sum_{k \in K} \frac{|X_k|}{|X|} \log_2 \frac{|X_k|}{|X|}$$

Information Gain:

$$info(X, A_j) = H(X) - \sum_{a \in A_j} \frac{|X_{j,a}|}{|X|} \cdot H(X_{j,a})$$

bevorzugt Attribute mit vielen möglichen Werten!

Information Gain Ratio als Heuristik

Splitinformation:

$$split(X, A_j) = -\sum_{a \in A_j} \frac{|X_{j,a}|}{|X|} \log_2 \frac{|X_{j,a}|}{|X|}$$

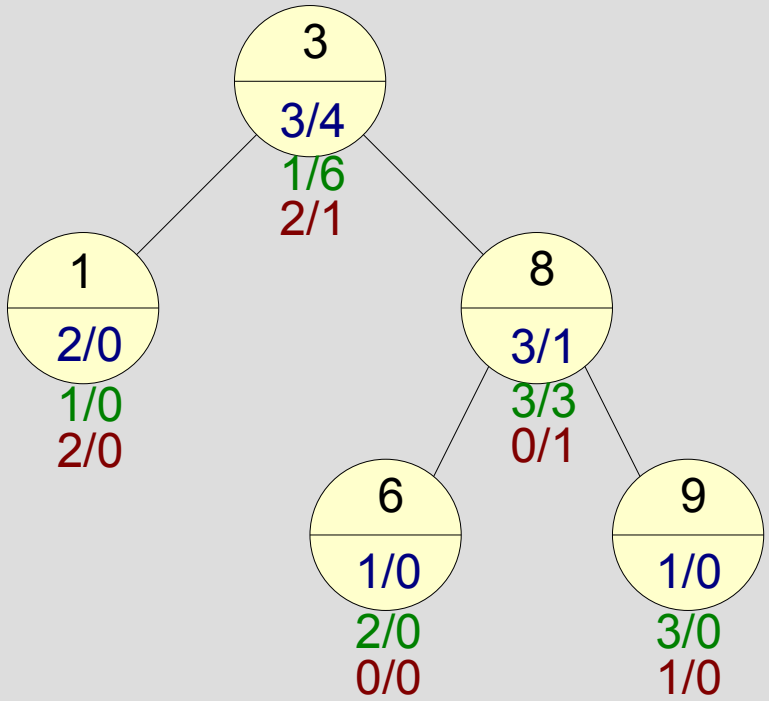
steigt mit Anzahl der Werte des Attributs

Information Gain Ratio:

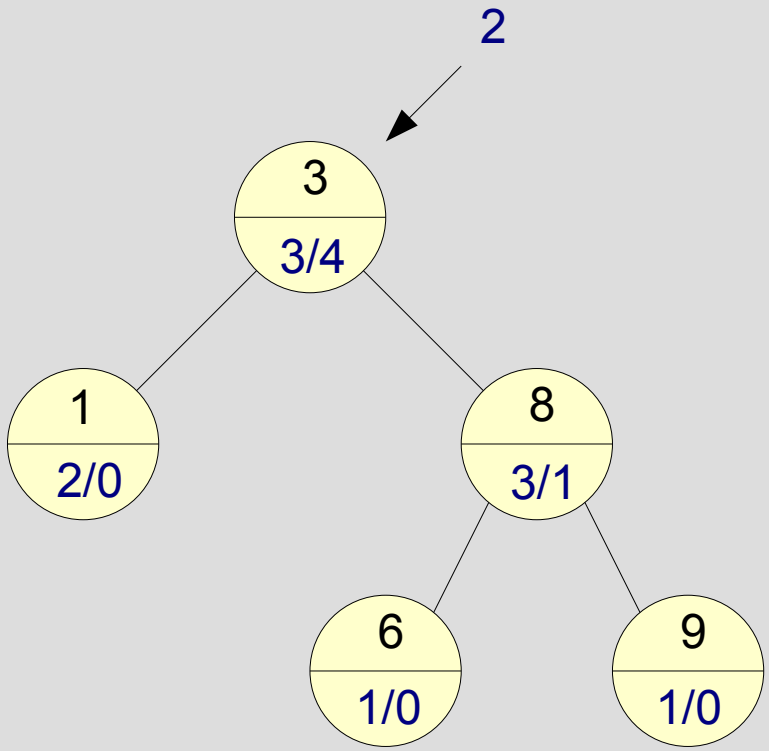
$$inforatio(X, A_j) = \frac{info(X, A_j)}{split(X, A_j)}$$

Umsetzung in VFDTc: Numerische Attribute

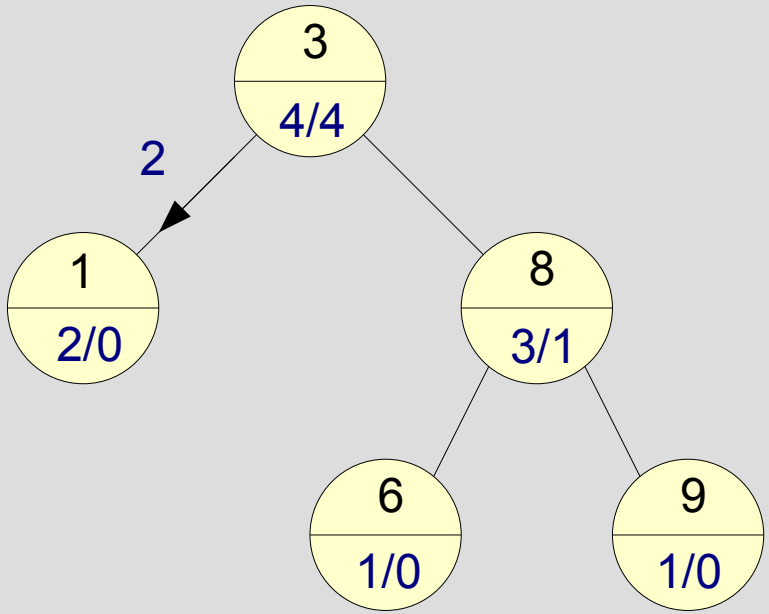
Speicherung numerischer Werte in binären Bäumen



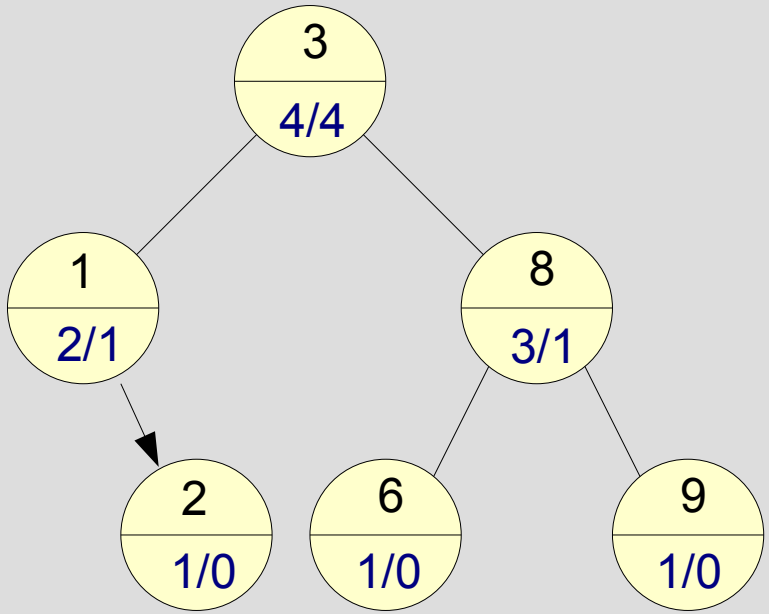
Numerische Attribute: Hinzufügen eines neuen Wertes



Numerische Attribute: Hinzufügen eines neuen Wertes



Numerische Attribute: Hinzufügen eines neuen Wertes



Umsetzung in VFDTc: Numerische Attribute

- Verwaltung numerischer Attribute in binären Bäumen
- Baumstruktur macht Trennung an beliebigen Cut-off-Werten einfach
- Information Gain des Attributes = Information Gain des besten Cut-off-Wertes

Umsetzung in VFDTc: Naive Bayes in Blättern

Bayes Theorem:

$$P(C_k|x_1, x_2, \ldots, x_n) = \frac{P(C_k)}{P(x_1, x_2, \ldots, x_n)} P(x_1, x_2, \ldots, x_n|C_k)$$

Annahme der Unabhängigkeit der Attribute (daher „naiv“):

$$\frac{P(C_k)}{P(x_1, x_2, \ldots, x_n)} P(x_1, x_2, \ldots, x_n|C_k) = \frac{P(C_k)}{P(x_1, x_2, \ldots, x_n)} \prod_{i=1}^{n} P(x_i|C_k) \propto P(C_k) \prod_{i=1}^{n} P(x_i|C_k)$$

Wahrscheinlichkeiten lassen sich Hilfe der gespeicherten Werte sehr einfach abschätzen

Umsetzung in VFDTc: Naive Bayes in Blättern

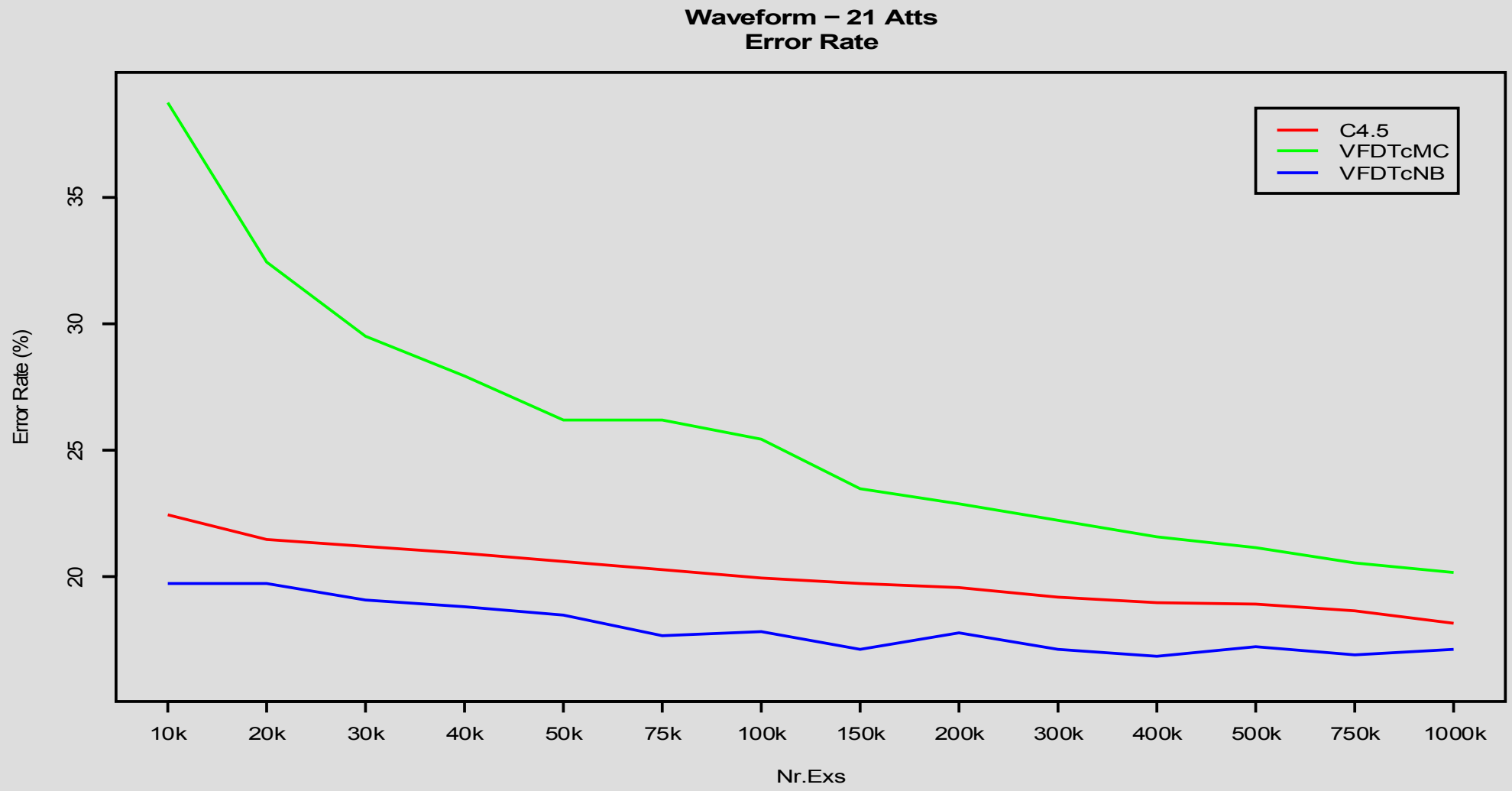
Bei numerischen Werten Unterteilung in maximal 10 gleichgroße Bereiche

Alternative: Verteilung schätzen – dazu später mehr

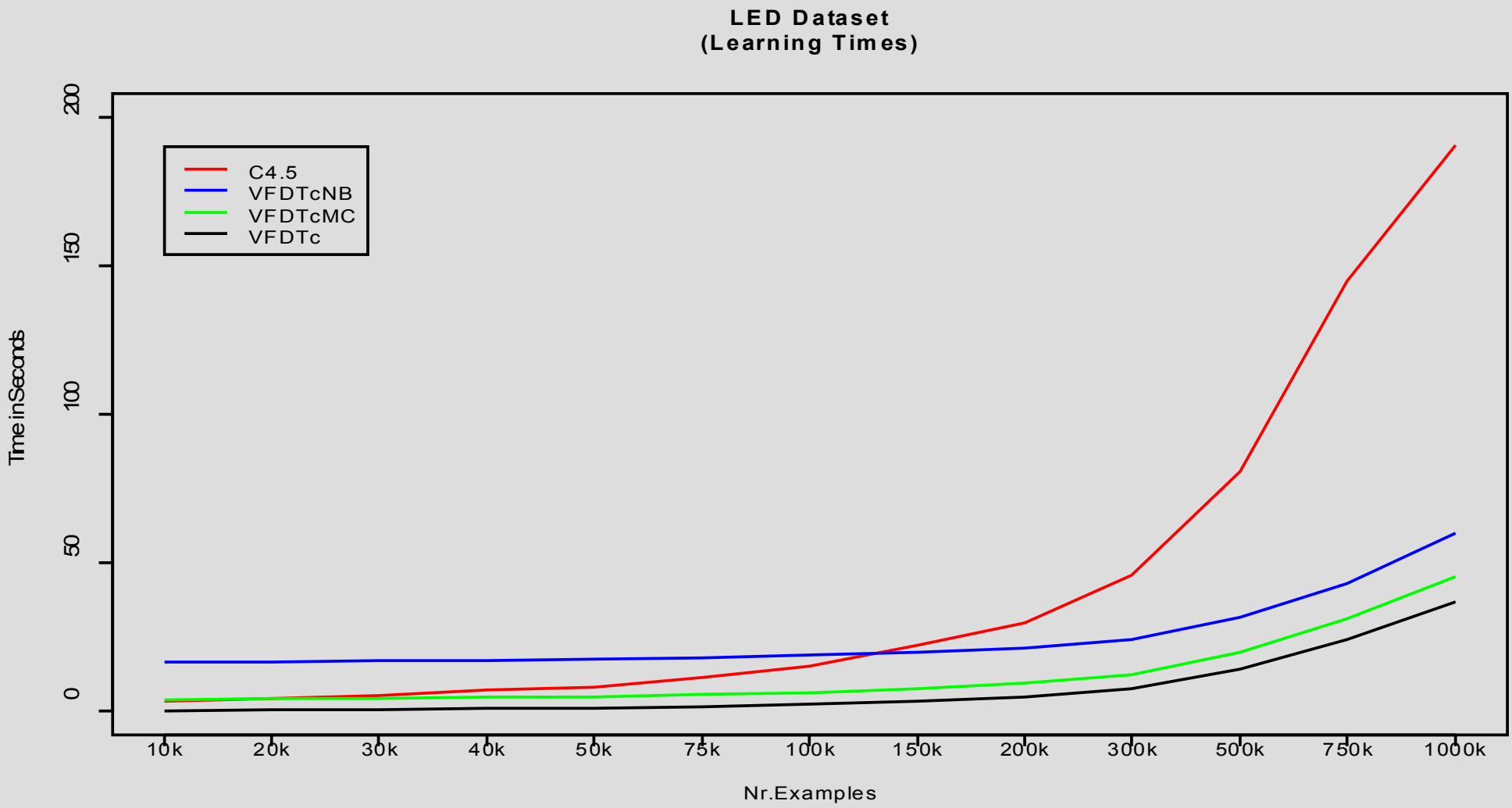
Vorteile von Naive Bayes

- Kann sehr gut mit inkrementellen Beispielsets umgehen
- Liefert schon bei kleiner Zahl von Beispielen gute Ergebnisse
- Kann mit heterogenen Daten und fehlenden Werten umgehen

Naive Bayes vs. Majority Class



Naive Bayes vs. Majority Class



Naive Bayes vs. Majority Class

VFDTcNB

- braucht sehr viel weniger Vorlaufzeit als VFDTcMC
- erzielt sogar bei relativ wenig Beispielen bessere Ergebnisse als C4.5
- benötigt etwas mehr Zeit wegen der Anwendung des Naive-Bayes-Algorithmus

Verbesserungsvorschläge

Verbesserungen bei UFFT-Algorithmus

- Verwenden statistischer Heuristik zur Wahl des Cut-off-Werts
- Speichern der letzten n Beispiele in „Kurzzeitgedächtnis“ zum Initialiseren neuer Blätter
- Bei Gleichstand zweier Attribute Vergleich mit Naive Bayes als Entscheider

Statistische Heuristik zur Bestimmung eines Cut-off-Punktes

- Es wird angenommen, daß die Klassen gleichverteilt sind.
- Zwischen zwei Klassen werden die Schnittpunkte der Häufigkeitsfunktionen, multipliziert mit der Wahrscheinlichkeit der jeweiligen Klasse, berechnet und derjenige gewählt, der näher am Mittelpunkt beider Verteilungen liegt
- Dadurch ist für jedes Attribut nur noch die Speicherung der Varianz, Anzahl und des Mittelpunkts der Beispiele nötig
- Ermöglichkeit dem Naive-Bayes-Algorithmus eine Berechnung der Wahrscheinlichkeit ohne Diskretisierung

- Nachteil: es können nur zwei Klassen verglichen werden
- Urpsrünglicher Vorschlag des 2-Means-Clustering nicht möglich
- Lösung: Round Robin – für jeweils 2 Klassen wird ein Baum erstellt

Naive Bayes als Entscheider in einem Knoten

- Bei Feststellung eines Gleichstands wird Statistik geführt, wie Naive Bayes die nachfolgenden Beispiele klassifiziert
- Erreicht Naive Bayes ein besseres Information Gain Ratio, wird dieses als Entscheider verwendet

- Vorteil: Naive Bayes kann mehrere (evtl. korrelierte) Attribute kombinieren
- Nachteil: Verwendet alle Attribute, also auch evtl. unwichtige
- Lösung: Naive Bayes-Klassifizierer nur über die Attribute mit dem höchsten Information Gain Ratio

Fazit

VFDTc

- ist eine effiziente Implementierung des VFDT-Algorithmus
- erweitert VFDT um funktionelle Knoten, um die Klassifizierung bei wenigen Beispielen zu verbessern
- bietet trotz allem noch Möglichkeiten zur Optimierung (siehe UFFT)

Quellen

- João Gama, Ricardo Rocha, Pedro Medas, Accurate Decision Trees for mining high-speed Data Streams, Proceedings of the 9th ACM SigKDD International Conference in Knowledge Discovery and Data Mining, ACM Press, 2003
- P. Domingos and G. Hulten, Mining high-speed data streams, Proceedings of the 6th ACM SIGKDD International Conference on Knowledge discovery and data mining 71-80, 2000.
- João Gama, Pedro Medas, Pedro Rodrigues, Learning decision trees from dynamic data streams, Proceedings of the 2005 ACM symposium on Applied computing, March 13-17, 2005, Santa Fe, New Mexico
- Wei-Yin Loh and Yu-Shan Shih, Split Selection Methods for Classification Trees, Statistica Sinica 7(1997), 815-840