Attribute

-name : String
-index : int

<<Constructor>> #Attribute(name : String, index : int)
#getName() : String
+getIndex() : int
+toString() : String

ContinuousAttribute

-max : float
-min : float

<<Constructor>> ~ContinuousAttribute(name : String, index : int, min : float, max : float)
#getMin() : float
#getMax() : float
+iterator() : Iterator<Float>

<<Interface>>
java.lang.Iterable<Float>

DiscreteAttribute

-values : String[]

<<Constructor>> ~DiscreteAttribute(name : String, index : int, values : String[])
+getValue(index : int) : String
+getNumberOfDistinctValues() : int

<<Interface>>
java.io.Serializable

DiscreteItem

<<Constructor>> +DiscreteItem(attribute : DiscreteAttribute, value : String)
+checkItemCondition(value : Object) : boolean

Data

-data : Object[][]
-numberOfExamples : int
-attributeSet : List<Attribute>

<<Constructor>> +Data(table : String)
+getNumberOfExamples() : int
+getNumberOfAttributes() : int
+getAttributeValue(exampleIndex : int, attributeIndex : int) : Object
+getAttribute(attributeIndex : int) : Attribute
+toString() : String

java.sql.SQLException

database.DatabaseConnectionException

database.NoValueException

mining.Item

EmptySetException

+EmptySetException(mes : String)

java.lang.Exception

ContinuousAttributeIterator

-min : float
-max : float
-j : int = 0
-numValues : int

<<Constructor>> ~ContinuousAttributeIterator(min : float, max : float, numValues : int)
+hasNext() : boolean
+next() : Float
+remove()

<<Interface>>
java.util.Iterator<Float>

Powered By Visual Paradigm Community Edition

data.Attribute

Continuousltem

<<Constructor>> ~ContinuousItem(attribute : ContinuousAttribute, value : Interval)
+checkItemCondition(value : Object) : boolean
+toString() : String
#operation()

Item

-attribute : Attribute
-value : Object

<<Constructor>> #Item(attribute : Attribute, value : Object)
#getAttribute() : Attribute
#getValue() : Object
+checkItemCondition(value : Object) : boolean
+toString() : String

AssociationRule

-antecedent : Item[]
-consequent : Item[]
-support : float
-confidence : float

<<Constructor>> ~AssociationRule(support : float)
#getSupport() : float
+getConfidence() : float
~getAntecedentLength() : int
~getConsequentLength() : int
~addAntecedentItem(item : Item)
~addConsequentItem(item : Item)
~getAntecedentItem(index : int) : Item
~getConsequentItem(index : int) : Item
#setConfidence(confidence : float)
+toString() : String
+compareTo(ar : AssociationRule) : int

<<Interface>>
java.lang.Comparable<AssociationRule>

<<Interface>>
java.io.Serializable

FrequentPattern

-fp : List<Item>
-support : float

<<Constructor>> ~FrequentPattern()
#addItem(item : Item)
~getItem(index : int) : Item
+getSupport() : float
~getPatternLength() : int
~setSupport(support : float)
+toString() : String
+iterator() : Iterator<Item>

Interval

-inf : float
-sup : float

<<Constructor>> ~Interval(inf : float, sup : float)
-setInf(inf : float)
-setSup(sup : float)
#getInf() : float
#getSup() : float
~checkValueInclusion(value : float) : boolean
+toString() : String

AssociationRuleArchive

-archive : HashMap<FrequentPattern, TreeSet<AssociationRule>>

+put(fp : FrequentPattern)
+put(fp : FrequentPattern, rule : AssociationRule)
+getRules(fp : FrequentPattern) : TreeSet<AssociationRule>
+salva(nomefile : String)
+carica(nomefile : String) : AssociationRuleArchive
+getArchive() : HashMap<FrequentPattern, TreeSet<AssociationRule>>
+toString() : String

NoPatternException

<<Constructor>> +NoPatternException(mes : String)

---

java.io.IOException

---

java.io.FileNotFoundException

---

java.lang.ClassNotFoundException

AssociationRuleMiner

+confidentAssociationRuleDiscovery(data : Data, fp : FrequentPattern, minConf : float) : LinkedList<AssociationRule>
-confidentAssociationRuleDiscovery(data : Data, fp : FrequentPattern, minConf : float, iCut : int) : AssociationRule
-computeConfidence(data : Data, AR : AssociationRule) : float

---

OneLevelPatternException

<<Constructor>> +OneLevelPatternException(mes : String)

FrequentPatternMiner

+frequentPatternDiscovery(data : Data, minSup : float) : LinkedList<FrequentPattern>
-refineFrequentPattern(fp : FrequentPattern, item : Item) : FrequentPattern
-expandFrequentPatterns(data : Data, minSup : float, fpQueue : Queue<FrequentPattern>, outputFP : LinkedList<FrequentPattern>)
-computeSupport(data : Data, FP : FrequentPattern) : float

---

data.EmptySetException

<<throws>>

<<throws>>

<<throws>>

<<throws>>

<<throws>>

<<throws>>

<<throws>>

<<throws>>

<<throws>>

<<throws>>

1

1

1

2

1

Pdf

-path : String
-minSup : float
-tableName : String
-image : BufferedImage

+Pdf(path : String, minSup : float, minConf : float, tableName : String, image : BufferedImage)
+SalvaPdf()

java.io.FileNotFoundException

java.io.IOException

com.itextpdf.text.DocumentException

<<throws>>

java.lang.ClassNotFoundException

PrecisionRecallCurve

-archive : HashMap<FrequentPattern, TreeSet<AssociationRule>>
-arch : AssociationRuleArchive

+PrecisionRecallCurve(archive : HashMap<FrequentPattern, TreeSet<AssociationRule>>, AssociationRule arch)
+suppCount() : int[]
+confCount() : int[]

java.sql.SQLException

<<throws>>

mining.AssociationRuleArchive

mining.NoPatternException

MultiServer

-PORT : int = 8080 {readOnly}

<<Constructor>> +MultiServer()
+run()
+main(args : String[])

ServerOneClient

-socket : Socket
-in : ObjectInputStream
-out : ObjectOutputStream
-inBuf : BufferedInputStream
-archive : AssociationRuleArchive

~ServerOneClient(s : Socket)
+run()

java.io.IOException

<<throws>>

1

1

java.lang.Thread

mining.AssociationRuleArchive

ConnectionLog

-outFile : FileOutputStream
-outWriter : PrintStream
-dateFormatter : SimpleDateFormat

+ConnectionLog(logText : String)
+writeLogText(logText : String)
+logClosing()

EmptyQueueException

<<Constructor>> +EmptyQueueException(mes : String)

java.lang.Exception

<<throws>>

Queue

-begin : Record = null
-end : Record = null

+isEmpty() : boolean
+enqueue(e : T)
+first() : T
+dequeue()

T

Record

+elem : T
+next : Record

<<Constructor>> +Record(e : T)

<<throws>>

Cella

~elemento : Object
~successivo : Puntatore = null

<<Constructor>> +Cella(e : Object)

1

Puntatore

+link : Cella

<<Constructor>> +Puntatore(c : Cella)

1

1

1

LinkList

-inizioLista : Puntatore = null

+isEmpty() : boolean
+firstList() : Puntatore
+endList(p : Puntatore) : boolean
+readList(p : Puntatore) : Object
+add(e : Object)
+succ(p : Puntatore) : Puntatore