Generation of Synthetic Data to Conform to Constraints Derived from Data Mining Applications

Josh Eno
MS Computer Science

Committee:
Dr. Craig Thompson
Dr. Brahendra Panda
Dr. David Douglas
Agenda

• Introduction
• Background
• PMML Conversion Software
• PMML Model Handlers
• Analysis
• Conclusion
Introduction: Problem

• Synthetic Data is useful in several areas
• Two Basic Approaches
  – Multivariate Distribution
  – System Model
• There is a need for a hybrid model, combining the power and flexibility of both
Introduction: Objective

The objective of this thesis is to demonstrate that the decision tree data mining technique can discover patterns that can be reverse mapped back into synthetic data sets of any size that will faithfully exhibit the same patterns.
Background: PMML

- Need some way to store data mining models
- Used Predictive Model Markup Language (PMML)
  - Industry standard supported by Oracle, SPSS, IBM, Microsoft, etc. through the Data Mining Group
  - XML format with vendor extensions
Background: PMML

- Consists of Data Dictionary and 1 or more Mining Models
- Data Dictionary
  - Contains field name and type information
- Mining Model
  - 11 types of model, including trees, regression, neural net
  - XML Schema varies by model type
Background: SDDL

- Need some way to specify data
- Synthetic Data Definition Language
  - XML language to describe data sets
  - Developed by Joe Hoag and Craig Thompson
  - Contains a database element with multiple pool and table elements
Background: SDDL

• Pool Elements
  – A type of data dictionary with weighted pool choices
  – Each choice can have multiple auxiliary attributes as well as nested sub-pools

• Table Elements
  – Defines a set of fields to generate
  – Variable and field elements
Related Work

• Multivariate Distribution Simulation
  – Simulates data as a statistical distribution
  – Defined by sufficient statistics (mean vector, covariance matrix, etc.)
  – Limitations on the kinds of data that can be simulated
Related Work

• Synthetic Data Generators
  – Commercial and experimental software with a common goal of simulating large data sets with database-like relationships
  – Approaches vary in data definition mechanism and generation framework
  – Each approach lacks either flexibility of SDDL or efficiency of the Parallel Synthetic Data Generator
PMML Conversion Software

- Software demonstrates the ability to create simulated data based on decision tree model
- Parses PMML 3.0 with a decision tree mining model and creates an SDDL file
- Extensible architecture for future mining models
PMML Software Architecture

- Three Tier Architecture
  - Top layer provides an interface, currently CLI
  - Middle layer handles PMML common to all files
  - Bottom layer handles specific models, with varying XML schema
PMML Software: Driver Layer

• Command Line Interface
  – Specify input, output, and properties files
  – Properties file may contain input/output files as well as options for specific models
  – Decision tree model allows the specification of global minimum and maximum field values
  – Other options include random number generator seed value, database name, and number of rows to create
PMML Software: Parser Layer

• Implements a SAX XML Parser
• Receives tags sequentially, rather than as a full XML tree
• Handles tags that are not model specific, which primarily concern the data dictionary
• Also launches model handlers when a mining model tag is encountered
PMML Software: Parser Layer

• Data Dictionary Parsing
  – Creates a field list for use by individual mining models
  – Classifies fields as integer, real, or string
  – For categorical string values, the data dictionary also includes information on valid values
PMML Software: Model Handlers

- Model Handler Interface
  - Each model handler must implement a model handler interface
  - Allows parser layer to pass tags to model handlers
  - Model handlers return a Boolean value to indicate whether the tag was handled
PMML Software: Tree Handler

- Creates SDDG based on a decision tree classification model
- Nodes in the tree have predicates which must be satisfied for a record to fall within the node
- These predicates are used to constrain the generated data
PMML Tree Model Structure

- Mining Schema
  - Classifies fields as active, predicted, or supplementary
    - Active fields determine the path of a record through the tree
    - Predicted field determines the category of a row that reaches a leaf node
    - Supplementary fields are for internal use by the data mining software and are ignored for SDDL purposes
PMML Tree Model Structure

- Node define the actual tree
  - Node score determines the value of predicted field
  - Predicate is used to determine record path through the tree
  - Score distribution tells how many records of each category pass through the node
PMML Tree Model Structure

<Node score="Iris-virginica" recordCount="6" id="5">
  <CompoundPredicate booleanOperator="surrogate">
    <SimplePredicate field="petal length"
      operator="greaterThan" value="4.95"/>
    ...
  </CompoundPredicate>
  <ScoreDistribution value="Iris-setosa" recordCount="0">
    ...
  </ScoreDistribution>
  ...
</Node>

<Node score="Iris-virginica" recordCount="3" id="6">
  ...
</Node>

<Node score="Iris-versicolor" recordCount="3" id="7">
  ...
</Node>
PMML Tree Model Structure

• Surrogate Compound Predicates
  – SPSS Clementine creates compound predicates with surrogate simple predicates
  – Only the first predicate with available data applies
  – Leaves a question about how to handle lower level predicates that are not applied
Tree Scanning Algorithm

• Performs a depth first search of the tree
• Constraints are propagated down the tree to leaf nodes
• Leaf node scores, record counts, score distributions, and data constraints are stored in a list for table build phase
Tree Scanning Algorithm

- In the event of a predicate conflict, the lower level node takes precedence.
- In this case, node 5 would have a PL minimum of 4.95 and the maximum of 4.75 from node 3 would be loosened to the global maximum.
SDDL Build Algorithm

• Each leaf node is used to build a choice in a pool in the database
• The weight is the record count of the node
• Field minimums and maximums are stored as auxiliary values
• Categorical values are stored as a sub-pool
  – Predicted field weights are set based on score distribution
  – Active field weights are equal unless excluded from the set by a set predicate
SDDL Pool Example

<choice name="3">
   <chol_max>405</chol_max>
   <chol_min>126</chol_min>
   <pool name="sex">
      <choice name="female">
         <weight>1</weight>
      </choice>
      <choice name="male">
         <weight>0</weight>
      </choice>
   </pool>
</choice>

<pool name="R_num">
   <choice name="lt_50">
      <weight>103</weight>
   </choice>
   <choice name="gt_50_1">
      <weight>10</weight>
   </choice>
   <weight>113.0</weight>
</pool>
SDDL Build Algorithm

- **Table Build Phase**
  - Contains one variable field to select node pool choice
  - Numerical fields have minimum and maximum constrained by choice auxiliary fields
    - NodePool[nodeId].chol_min
  - Categorical fields choose a value from a sub-pool
    - NodePool[nodeId].R_num
Analysis

• The PMML file was used to create an SDDL file.
• A large data set was generated based on the SDDL File.
• The generated data was loaded into a relational database for analysis.
• The data was analyzed through a series of SQL queries.
Iris Data Analysis

- Simple data set first used by R. Fisher in 1934 [1], commonly used to evaluate machine learning algorithms
- 150 records, 50 each from 3 species of Iris, measuring length and width of sepal and petal
- One species, Setosa, is linearly separable from the others, but Virginica and Versicolor are not linearly separable
# Iris Data Analysis

<table>
<thead>
<tr>
<th>Node ID</th>
<th>Training Data</th>
<th>Generated Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Total</td>
<td>Setosa</td>
</tr>
<tr>
<td>0</td>
<td>150</td>
<td>50</td>
</tr>
<tr>
<td>1</td>
<td>50</td>
<td>50</td>
</tr>
<tr>
<td>2</td>
<td>100</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>54</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>48</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>3</td>
<td>0</td>
</tr>
<tr>
<td>8</td>
<td>46</td>
<td>0</td>
</tr>
</tbody>
</table>
## Iris Data Analysis

<table>
<thead>
<tr>
<th>Node ID</th>
<th>Total</th>
<th>Setosa</th>
<th>Virginica</th>
<th>Versicolor</th>
<th>Total</th>
<th>Setosa</th>
<th>Virginica</th>
<th>Versicolor</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1.000</td>
<td>0.333</td>
<td>0.333</td>
<td>0.333</td>
<td>1.000</td>
<td>0.333</td>
<td>0.334</td>
<td>0.333</td>
</tr>
<tr>
<td>1</td>
<td>0.333</td>
<td>0.333</td>
<td>0.000</td>
<td>0.000</td>
<td>0.333</td>
<td>0.333</td>
<td>0.000</td>
<td>0.000</td>
</tr>
<tr>
<td>2</td>
<td>0.667</td>
<td>0.000</td>
<td>0.333</td>
<td>0.333</td>
<td>0.667</td>
<td>0.000</td>
<td>0.334</td>
<td>0.333</td>
</tr>
<tr>
<td>3</td>
<td>0.360</td>
<td>0.000</td>
<td>0.033</td>
<td>0.327</td>
<td>0.359</td>
<td>0.000</td>
<td>0.034</td>
<td>0.326</td>
</tr>
<tr>
<td>4</td>
<td>0.320</td>
<td>0.000</td>
<td>0.007</td>
<td>0.313</td>
<td>0.319</td>
<td>0.000</td>
<td>0.007</td>
<td>0.313</td>
</tr>
<tr>
<td>5</td>
<td>0.040</td>
<td>0.000</td>
<td>0.027</td>
<td>0.013</td>
<td>0.040</td>
<td>0.000</td>
<td>0.027</td>
<td>0.013</td>
</tr>
<tr>
<td>6</td>
<td>0.020</td>
<td>0.000</td>
<td>0.020</td>
<td>0.000</td>
<td>0.020</td>
<td>0.000</td>
<td>0.203</td>
<td>0.000</td>
</tr>
<tr>
<td>7</td>
<td>0.020</td>
<td>0.000</td>
<td>0.007</td>
<td>0.013</td>
<td>0.020</td>
<td>0.000</td>
<td>0.007</td>
<td>0.013</td>
</tr>
<tr>
<td>8</td>
<td>0.307</td>
<td>0.000</td>
<td>0.300</td>
<td>0.007</td>
<td>0.307</td>
<td>0.000</td>
<td>0.300</td>
<td>0.007</td>
</tr>
</tbody>
</table>
Heart Data Analysis

• Heart Disease indicators from the Cleveland Clinic [2]
• 13 active fields, with a mix of categorical and numeric values
• 303 records, classified by greater than or less than 50% narrowed arteries
# Heart Data Analysis

## Heart Node Record Counts

<table>
<thead>
<tr>
<th>Node ID</th>
<th>Total</th>
<th>&lt;50</th>
<th>&gt;50</th>
<th>Total</th>
<th>&lt;50</th>
<th>&gt;50</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>303</td>
<td>165</td>
<td>138</td>
<td>303000</td>
<td>165239</td>
<td>137761</td>
</tr>
<tr>
<td>1</td>
<td>167</td>
<td>130</td>
<td>37</td>
<td>167176</td>
<td>130212</td>
<td>36964</td>
</tr>
<tr>
<td>2</td>
<td>118</td>
<td>105</td>
<td>13</td>
<td>117958</td>
<td>104994</td>
<td>12964</td>
</tr>
<tr>
<td>3</td>
<td>113</td>
<td>103</td>
<td>10</td>
<td>112997</td>
<td>102996</td>
<td>10001</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>2</td>
<td>3</td>
<td>4961</td>
<td>1998</td>
<td>2963</td>
</tr>
<tr>
<td>5</td>
<td>49</td>
<td>25</td>
<td>24</td>
<td>49218</td>
<td>25218</td>
<td>24000</td>
</tr>
<tr>
<td>6</td>
<td>20</td>
<td>3</td>
<td>17</td>
<td>20134</td>
<td>3037</td>
<td>17097</td>
</tr>
<tr>
<td>7</td>
<td>29</td>
<td>22</td>
<td>7</td>
<td>29084</td>
<td>22181</td>
<td>6903</td>
</tr>
<tr>
<td>8</td>
<td>11</td>
<td>6</td>
<td>5</td>
<td>10966</td>
<td>6040</td>
<td>4926</td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td>4</td>
<td>1</td>
<td>5006</td>
<td>4028</td>
<td>978</td>
</tr>
<tr>
<td>10</td>
<td>6</td>
<td>2</td>
<td>4</td>
<td>5960</td>
<td>2012</td>
<td>3948</td>
</tr>
<tr>
<td>11</td>
<td>18</td>
<td>16</td>
<td>2</td>
<td>18118</td>
<td>16141</td>
<td>1977</td>
</tr>
<tr>
<td>12</td>
<td>136</td>
<td>35</td>
<td>101</td>
<td>135824</td>
<td>35027</td>
<td>100797</td>
</tr>
<tr>
<td>13</td>
<td>90</td>
<td>10</td>
<td>80</td>
<td>90014</td>
<td>10083</td>
<td>79931</td>
</tr>
<tr>
<td>14</td>
<td>22</td>
<td>8</td>
<td>14</td>
<td>22105</td>
<td>8152</td>
<td>13953</td>
</tr>
<tr>
<td>15</td>
<td>12</td>
<td>7</td>
<td>5</td>
<td>12078</td>
<td>7111</td>
<td>4967</td>
</tr>
<tr>
<td>16</td>
<td>4</td>
<td>4</td>
<td>0</td>
<td>4124</td>
<td>4124</td>
<td>0</td>
</tr>
<tr>
<td>17</td>
<td>8</td>
<td>3</td>
<td>5</td>
<td>7954</td>
<td>2987</td>
<td>4967</td>
</tr>
<tr>
<td>18</td>
<td>10</td>
<td>1</td>
<td>9</td>
<td>10027</td>
<td>1041</td>
<td>8986</td>
</tr>
<tr>
<td>19</td>
<td>68</td>
<td>2</td>
<td>66</td>
<td>67909</td>
<td>1931</td>
<td>65978</td>
</tr>
<tr>
<td>20</td>
<td>46</td>
<td>25</td>
<td>21</td>
<td>45810</td>
<td>24944</td>
<td>20866</td>
</tr>
<tr>
<td>21</td>
<td>29</td>
<td>21</td>
<td>8</td>
<td>28846</td>
<td>20924</td>
<td>7922</td>
</tr>
<tr>
<td>22</td>
<td>17</td>
<td>4</td>
<td>13</td>
<td>16964</td>
<td>4020</td>
<td>12944</td>
</tr>
<tr>
<td>23</td>
<td>12</td>
<td>1</td>
<td>11</td>
<td>11948</td>
<td>1004</td>
<td>10944</td>
</tr>
<tr>
<td>24</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td>5016</td>
<td>3016</td>
<td>2000</td>
</tr>
</tbody>
</table>
# Heart Data Analysis

<table>
<thead>
<tr>
<th>Node ID</th>
<th>Training Data</th>
<th>Generated Data</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>&lt;50</td>
</tr>
<tr>
<td>0</td>
<td>0.545</td>
<td>0.455</td>
</tr>
<tr>
<td>1</td>
<td>0.429</td>
<td>0.122</td>
</tr>
<tr>
<td>2</td>
<td>0.340</td>
<td>0.033</td>
</tr>
<tr>
<td>3</td>
<td>0.017</td>
<td>0.010</td>
</tr>
<tr>
<td>4</td>
<td>0.083</td>
<td>0.079</td>
</tr>
<tr>
<td>5</td>
<td>0.010</td>
<td>0.056</td>
</tr>
<tr>
<td>6</td>
<td>0.073</td>
<td>0.023</td>
</tr>
<tr>
<td>7</td>
<td>0.020</td>
<td>0.017</td>
</tr>
<tr>
<td>8</td>
<td>0.013</td>
<td>0.003</td>
</tr>
<tr>
<td>9</td>
<td>0.007</td>
<td>0.013</td>
</tr>
<tr>
<td>10</td>
<td>0.053</td>
<td>0.007</td>
</tr>
<tr>
<td>11</td>
<td>0.116</td>
<td>0.333</td>
</tr>
<tr>
<td>12</td>
<td>0.033</td>
<td>0.264</td>
</tr>
<tr>
<td>13</td>
<td>0.026</td>
<td>0.046</td>
</tr>
<tr>
<td>14</td>
<td>0.013</td>
<td>0.000</td>
</tr>
<tr>
<td>15</td>
<td>0.010</td>
<td>0.017</td>
</tr>
<tr>
<td>16</td>
<td>0.003</td>
<td>0.030</td>
</tr>
<tr>
<td>17</td>
<td>0.007</td>
<td>0.218</td>
</tr>
<tr>
<td>18</td>
<td>0.083</td>
<td>0.069</td>
</tr>
<tr>
<td>19</td>
<td>0.069</td>
<td>0.100</td>
</tr>
<tr>
<td>20</td>
<td>0.013</td>
<td>0.043</td>
</tr>
<tr>
<td>21</td>
<td>0.003</td>
<td>0.036</td>
</tr>
<tr>
<td>22</td>
<td>0.010</td>
<td>0.007</td>
</tr>
</tbody>
</table>
Data Analysis: Summary

- 150,000 rows in Iris set, 303,000 rows in Heart Disease set
- Of 68 probability measures, only one had a probability difference greater than .001
- Data sets indistinguishable based on decision tree model
Conclusion: Summary

• Goal: Simulate data based on decision tree mining model
• Created software to convert a decision tree stored as PMML to an SDDL data specification
• Generated data sets 1000 times as large and tested for similarity
Conclusion: Contributions

• Demonstrated viability of using a mining model as a data description
• Established architecture to expand to other mining models
• Creates a way to rapidly simulate data that does not conform to standard distributions
Conclusion: Future Work

• Implementation Work
  – Interactive interface
  – Additional mining models
  – Different PMML versions and software

• Theoretical Work
  – Simulating multiple models simultaneously
  – Theoretical limits of this approach