Differentially Private Data Classifiers

Slawek Goryczka
Classifiers

- Build a classification tree using differentially private data (noisy data)
- Interactive classifiers
  - Query the dataset when needed
  - SuLQ-based ID3*, FS differentially private ID3*
- Non-interactive classifiers
  - Use differentially private histogram
  - ID3-based classifier

* Arik Friedman and Assaf Schuster *Data Mining with Differential Privacy*
Interactive classifiers

- Privacy budget is evenly split for each database query
- Predefined maximal classification tree depth (no more than number of attributes)
- To choose a splitting attribute $\sum_i |A_i|*|C|$ queries need to be issued (SuLQ-based ID3)
- Use the exponential mechanism to get a splitting attribute (FS differentially private ID3)
Non-interactive classifiers

- Build a multidimensional histogram of data
  - Partition the domain into cells
  - Get a noisy count for each cell (privacy budget $\epsilon_0$)
  - Partition cells into groups with similar count
  - Get a noisy count for each partition (budget $\epsilon-\epsilon_0$)
- Use the histogram to build a classifier
Diff private histogram (results)

avg(abs(error))

- avg
- median
- stdev

epsilon0
## Classifiers (initial results)

<table>
<thead>
<tr>
<th>training set ratio</th>
<th>SuLQ</th>
<th>FS interactive</th>
<th>Histogram-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>46.81</td>
<td>46.81</td>
<td>99.04</td>
</tr>
<tr>
<td>0.7</td>
<td>47.45</td>
<td>47.45</td>
<td>99.34</td>
</tr>
<tr>
<td>0.9</td>
<td>46.6</td>
<td>46.6</td>
<td>98.84</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>training set ratio</th>
<th>SuLQ</th>
<th>FS interactive</th>
<th>Histogram-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.5</td>
<td>1</td>
<td>7</td>
<td>23</td>
</tr>
<tr>
<td>0.7</td>
<td>7</td>
<td>7</td>
<td>30</td>
</tr>
<tr>
<td>0.9</td>
<td>7</td>
<td>7</td>
<td>22</td>
</tr>
</tbody>
</table>
TODO

- Adapt code to run using Weka
- Prune the classification tree
- Reduce memory footprint to run application with more active dimensions
- Use Gini index and max operator
- Fixing bugs...ehm... introducing new features :)

Thank you!