Data Mining and Analytics on Gordon

Natasha Balac, Paul Rodriguez, Nicole Wolter
Outline

- Overview of Data Mining
  - Background, Applications, Tasks
  - Knowledge Data Discovery Process
  - Learning and Modeling Methods
- What is currently available at SDSC
- Preliminary case studies
Necessity is the Mother of Invention

Problem

Data explosion

Automated data collection tools and mature database technology lead to tremendous amounts of data stored in databases, data warehouses and other information repositories.

“We are drowning in data, but starving for knowledge!” (John Naisbitt, 1982)
Necessity is the Mother of Invention

Solution

Data Mining

- Extraction or “mining” of interesting knowledge (rules, regularities, patterns, constraints) from data in large databases
- Data-driven discovery and modeling of hidden patterns (we never new existed) in large volumes of data
- Extraction of implicit, previously unknown and unexpected, potentially extremely useful information from data
What Is Data Mining?

- Combination of AI and statistical analysis to discover information that is “hidden” in the data
  - associations (e.g. linking purchase of pizza with beer)
  - sequences (e.g. tying events together: marriage and purchase of furniture)
  - classifications (e.g. recognizing patterns such as the attributes of employees that are most likely to quit)
  - forecasting (e.g. predicting buying habits of customers based on past patterns)
Data Mining is NOT...

- Data Warehousing
- (Deductive) query processing
  - SQL/ Reporting
- Software Agents
- Expert Systems
- Online Analytical Processing (OLAP)
- Statistical Analysis Tool
- Data visualization
- BI – Business Intelligence
Data Mining is...

- Multidisciplinary Field
  - Database technology
  - Artificial Intelligence
    - Machine Learning including Neural Networks
  - Statistics
  - Pattern recognition
  - Knowledge-based systems/acquisition
  - High-performance computing
  - Data visualization
  - Other Disciplines
What can we do with Data Mining?

- Exploratory Data Analysis
- Predictive Modeling: Classification and Regression
- Descriptive Modeling
  - Cluster analysis/segmentation
- Discovering Patterns and Rules
  - Association/Dependency rules
  - Sequential patterns
  - Temporal sequences
- Deviation detection
Data Mining Applications

Science: Chemistry, Physics, Medicine
- Biochemical analysis, remote sensors on a satellite, Telescopes – star galaxy classification, medical image analysis

Bioscience
- Sequence-based analysis, protein structure and function prediction, protein family classification, microarray gene expression

Pharmaceutical companies, Insurance and Health care, Medicine
- Drug development, identify successful medical therapies, claims analysis, fraudulent behavior, medical diagnostic tools, predict office visits

Financial Industry, Banks, Businesses, E-commerce
- Stock and investment analysis, identify loyal customers vs. risky customer, predict customer spending, risk management, sales forecasting
Data Mining Applications

- **Database analysis and decision support**
  - Market analysis and management
    - target marketing, customer relation management, market basket analysis, cross selling, market segmentation (grocery store, Banking and Credit Card scoring, Personalization & Customer Profiling)
  - Risk analysis and management
    - Forecasting, customer retention, improved underwriting, quality control, competitive analysis (Banking and Credit Card scoring)
  - Fraud detection and management
Data Mining Applications

- **Sports and Entertainment**
  - IBM Advanced Scout analyzed NBA game statistics (shots blocked, assists, and fouls) to gain competitive advantage for New York Knicks and Miami Heat

- **Astronomy**
  - JPL and the Palomar Observatory discovered 22 quasars with the help of data mining

- Campaign Management and Database Marketing
Data Mining Tasks

- **Concept/Class description**: Characterization and discrimination
  - Generalize, summarize, and contrast data characteristics, e.g., dry vs. wet regions

- **Association** (correlation and causality)
  - Multi-dimensional interactions and associations
    - \( \text{age}(X, "20-29") \land \text{income}(X, "60-90K") \Rightarrow \text{buys}(X, "TV") \)
Data Mining Tasks

- **Classification and Prediction**
  - Finding models (functions) that describe and distinguish classes or concepts for future prediction
  - Example: classify countries based on climate, or classify cars based on gas mileage, fraud based on claims information
  - Presentation:
    - If-THEN rules, decision-tree, classification rule, neural network
  - Prediction: Predict some unknown or missing numerical values
Cluster analysis

- Class label is unknown: Group data to form new classes
  - Example: cluster houses to find distribution patterns
- Clustering based on the principle: maximizing the intra-class similarity and minimizing the interclass similarity
Data Mining Tasks

- **Outlier analysis**
  - Outlier: a data object that does not comply with the general behavior of the data
  - Mostly considered as noise or exception, but is quite useful in fraud detection, rare events analysis

- **Trend and evolution analysis**
  - Trend and deviation: regression analysis
  - Sequential pattern mining, periodicity analysis
KDD Process

Database

Selection Transformation → Data Preparation → Training Data

Data Mining

Model, Patterns

Evaluation, Verification
Learning and Modeling Methods

- Decision tree Induction (C4.5, J48)
- Regression tree Induction (CART, MP5)
- Multivariate Regression Tree (MARS)
- Clustering (K-means, EM, Cobweb)
- Neural network (Backpropagation, Recurrent)
- Support Vector machines
- Various other models
Decision Tree Induction

- Method for approximating discrete-valued functions
  - robust to noisy/missing data
  - can learn non-linear relationships
  - inductive bias towards shorter trees
Decision Tree Induction

- Applications:
  - medical diagnosis – ex. heart disease
  - analysis of complex chemical compounds
  - classifying equipment malfunction
  - risk of loan applicants
  - Boston housing project – price prediction
  - fraud detection
DT for Medical Diagnosis and Prognosis
Heart Disease

Minimum systolic blood pressure over a 24-hour period following admission to the hospital

<= 91

Class 2:
Early death

> 91

Age of Patient

<= 62.5

Class 1:
Survivors

Was there sinus tachycardia?

NO

Class 1:
Survivors

YES

Class 2:
Early death

<= 61.5

Class 1:
Survivors

> 62.5

Class 2:
Early death

Beriman et al, 1984
Regression Tree Induction

Why Regression tree?

- Ability to:
  - Predict continuous variable
  - Model conditional effects
  - Model uncertainty
Regression Trees

- Continuous goal variables
- Induction by means of an efficient recursive partitioning algorithm
- Uses linear regression to select internal nodes

Quinlan, 1992
Clustering

- Basic idea: Group similar things together
- Unsupervised Learning – Useful when no other info is available
- K-means
  - Partitioning instances into $k$ disjoint clusters
  - Measure of similarity
Artificial Neural Networks (ANNs)

- Network of many simple units
- Main Components
  - Inputs
  - Hidden layers
  - Outputs
- Adjusting weights of connections
- Backpropagation
Support Vector Machines (SVM)

- Blend of linear modeling and instance based learning
- SVM select a small number of critical boundary instances called support vectors from each class and build a linear discriminant function that separates them as widely as possible
- They transcend the limitations of linear boundaries by making it practical to include extra nonlinear terms in the calculations
  - making it possible to form quadratic, cubic, higher-order decision boundaries
Support Vector Machines

- Algorithms for learning linear classifiers
- Resilient to overfitting because they learn a particular linear decision boundary
  - The maximum margin hyperplane
- They are fast in the nonlinear case
  - Employ a clever mathematical trick to avoid the creation of “pseudo-attributes”
  - Nonlinear space is created implicitly
Support vectors

- The instances closest to the maximum margin hyperplane are called support vectors.
- Important observation: the support vectors define the maximum margin hyperplane.
  - All other instances can be deleted without changing the position and orientation of the hyperplane.
SVM: finding the maximum margin hyperplane
Data Mining Challenges

- Computationally expensive to investigate all possibilities
- Dealing with noise/missing information and errors in data
- Mining methodology and user interaction
  - Mining different kinds of knowledge in databases
  - Incorporation of background knowledge
  - Handling noise and incomplete data
  - Pattern evaluation: the interestingness problem
  - Expression and visualization of data mining results
Data Mining Heuristics and Guide

- Choosing appropriate attributes/input representation
- Finding the minimal attribute space
- Finding adequate evaluation function(s)
- Extracting meaningful information
- Not overfitting
Data mining applications on Dash

DM Suites
- SciberQuest
- WEKA
- RAPID MINER

Computational Packages with DM tools
- MathWorks
- Octave

Libraries for building tools
- FASTlib
- A library of Fundamental Algorithmic and Statistical Tools

Others as Requested

The Phoenix System for MapReduce Programming
Matlab and Data Analysis

- Mathematical and matrix operations
- Interactive or scripts
- Comes with tools (scripts) for data analysis,
  - e.g. clustering, neural networks, SVMs, ...
- Uses MKL (Intels math kernel library) for Matrix calculations with threads
Matlab in HPC setting

- Distributed toolbox provides distribute/gather functions

- In a nutshell:
  - Create job object: `createMatlabPoolJob(scheduler information)`
  - Create tasks for that job: `createTask(job,@myfunction,#tasks,{parameters..})`
  - In your code: `spmd`

    ```matlab
    D=codistributed(X); or D=codistributed.build(X);
    < statements>
    end;
    ```
Matlab in vSMP setting

- vSMP submission indicates threads
  
  In submission script,
  
  set environment variables for MKL

  In matlab code:

  ```matlab
  setenv('MKL_NUM_THREADS', getenv(number_of_procs));
  ```

- No programming changes necessary, but programming considerations exist
Matlab: matrix multiplication

In vSMP: $Y = X'_{N \times N} \times X_{N \times N}$

<table>
<thead>
<tr>
<th>Matrix size</th>
<th>Gb</th>
</tr>
</thead>
<tbody>
<tr>
<td>10K</td>
<td>2</td>
</tr>
<tr>
<td>20K</td>
<td>6.5</td>
</tr>
<tr>
<td>30K</td>
<td>14</td>
</tr>
<tr>
<td>40K</td>
<td>25</td>
</tr>
<tr>
<td>50K</td>
<td>40</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>8 threads</th>
<th>32 threads</th>
</tr>
</thead>
<tbody>
<tr>
<td>N=10K</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>20K</td>
<td>6.5</td>
<td>5</td>
</tr>
<tr>
<td>30K</td>
<td>14</td>
<td>13</td>
</tr>
<tr>
<td>40K</td>
<td>25</td>
<td>24</td>
</tr>
<tr>
<td>50K</td>
<td>40</td>
<td>40</td>
</tr>
</tbody>
</table>

36
Matlab: matrix inversion

In vSMP: $Y = \text{inv}(X_{NxN} + I \times 0.0001)$

<table>
<thead>
<tr>
<th>Matrix size</th>
<th>32 threads</th>
<th>16 threads</th>
</tr>
</thead>
<tbody>
<tr>
<td>N=10K</td>
<td>time(s)</td>
<td>time(s)</td>
</tr>
<tr>
<td>Gb=2</td>
<td>2</td>
<td>6.5</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>40</td>
</tr>
<tr>
<td>20K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>30K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>40K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>50K</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Matlab and Data Mining Case Study

- **Kmeans clustering**
  - Assign each point to one of a few clusters so that total distance to center is minimized
  - Options: distance function, number of clusters, initial cluster centers, number of iterations, stopping criteria
Matlab original Kmeans Script

1. Difference by col = X(:,1) - Cluster_Means(1,1)

2. square difference

3. sum as you loop across cols to get Distances to cluster center

Works better for large N small P
Matlab Kmeans Script altered

1. Difference_by_row = X(1,:) - Cluster_Means(1,:)

   \[ X_{NxP} \]
   \[ \text{Cluster_Means}_{MxP} \]

   each row is a point in \( R^p \)

2. \( \text{dot}(\text{difference_by_row}) \)

3. loop across rows to get Distances

Works better for large P and \( \text{dot}(\ ) \) will use threads
Matlab Kmeans Benchmarks

- Kmeans on 10,000,000 entries from NYTimes articles
  (http://archive.ics.uci.edu/ml/datasets/Bag+of+Words)
  - Running as full data matrix ~ 45K articles x102K words,
    - Each cell holds word count (double float)
    - about 37Gb in Matlab, total memory for script about 61Gb
  - Kmeans (original) runtime ~ 50 hours
  - Kmeans (altered) runtime ~ 10 hours, 8 threads
Matlab Kmeans Results

7 viable clusters found
MapReduce Framework

- A library for distributed computing
  - Started by Google, gaining popularity
  - Various implementations: Hadoop (distributed), Phoenix (threaded)

User outputs keys & values

MR provides parallelization, concurrency, and intermediate data functions (by key&value)

User defined functions
MapReduce Paradigmatic Example: string counting

- Scheduler: manage threads, initiate data split and call Map
- Map: count strings, output key=string & value=count
- Scheduler: re-partitions keys & values
- Reduce: sum up counts

User defined functions

MR provides parallelization, concurrency, and intermediate data functions (by key&value)
MapReduce Kmeans clustering

- C-code for Kmeans (sample code with MapReduce)
- Use 10,000,000 entries from NYTimes articles
  - Running as full data matrix (int) ~ 45K docs x 102K word tokens, about 20 Gb total in vSMP
  - Running time about 20 min, 32 threads

(but not a completely fair comparison to Matlab kmeans)
MapReduce Kmeans clustering

- Use ~70,000,000 entries from NYTimes articles
  - full data matrix (int) ~ 300K docs x 102K word tokens, about 120 Gb total in vSMP memory
  - Running time about 120 min, 32 threads
Case Study: Matrix Factorization
(work by R. Anil & C. Elkan, UCSD, CSE for KDD 2011 competition)

- Given large N sparse data matrix

\[ \mathbf{X}_{N \times P} \]

sparse missing data

customer ratings

items rated

Find vectors \( \mathbf{U}, \mathbf{V} \) such that \( \mathbf{X} \sim \sum f(\mathbf{U} \cdot \mathbf{V}') + \text{penalty} \)
Case Study: Matrix Factorization

- C code with pthreads
- For different $f(U \cdot V')$ functions

<table>
<thead>
<tr>
<th>Function</th>
<th>Time (s) 1 iteration</th>
<th>Memory</th>
<th>Dash node</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sigmoidal</td>
<td>74</td>
<td>29Gb</td>
<td>non-vSMP</td>
</tr>
<tr>
<td>Alternating Least Squares</td>
<td>673</td>
<td>15Gb</td>
<td>non-vSMP</td>
</tr>
<tr>
<td>Log-linear</td>
<td>1110</td>
<td>70Gb</td>
<td>vSMP</td>
</tr>
</tbody>
</table>
Data mining: discovering interesting patterns from large amounts of data

Discovery includes data cleaning, data integration, data selection, transformation, data mining, pattern evaluation, and knowledge presentation

Exploratory analyses
Ongoing and Future

- Continue building experience with large memory trade offs for Data Mining Algorithms
- Dash/Gordon will support a variety of tools
- Dash/Gordon will support a variety of ways to execute tools
  - hybrid options between shared and distributed memory