Introduction to Data Mining And Predictive Analytics

Paul Rodriguez, Ph.D.
Predictive Analytics Center of Excellence,
San Diego Supercomputer Center
University of California, San Diego
PACE: Closing the gap between Government, Industry and Academia

PACE is a non-profit, public educational organization
- To promote, educate and innovate in the area of Predictive Analytics
- To leverage predictive analytics to improve the education and well-being of the global population and economy
- To develop and promote a new, multi-level curriculum to broaden participation in the field of predictive analytics
Introduction to

WHAT IS DATA MINING?
Necessity is the Mother of Invention

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

Data explosion

“...explosion of heterogeneous, multi-disciplinary, Earth Science data has rendered traditional means of integration and analysis ineffective, necessitating the application of new analysis methods ... for synthesis, comparison, visualization” (Hoffman et al, Data Mining in Earth System Science 2011)
What does Data Mining Do?

- Explores Your Data
- Finds Patterns
- Performs Predictions
Data Mining perspective

Top-Down: Hypothesis Driven

Surface

Analytical Tools

SQL tools for simple queries and reporting

Shallow

Statistical & OLAP tools for summaries and analysis

Hidden

Data Mining methods for knowledge discovery

Bottom-Up: Data Driven
<table>
<thead>
<tr>
<th>Query Reporting</th>
<th>OLAP</th>
<th>Data Mining</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extraction of data</td>
<td>Descriptive summaries</td>
<td>Discovery of hidden patterns, information</td>
</tr>
<tr>
<td>Information</td>
<td>Analysis</td>
<td>Insight knowledge and prediction</td>
</tr>
<tr>
<td>Who purchased the product in the last 2 quarters?</td>
<td>What is an average income of the buyers per quarter by district?</td>
<td>What is the relationship between customers and their purchases?</td>
</tr>
</tbody>
</table>
What Is Data Mining in practice?

- 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)
Multidisciplinary Field

- Database Technology
- Statistics
- Machine Learning
- Artificial Intelligence
- Visualization
- Other Disciplines
TAXONOMY

• Predictive Methods
  • *Use some variables to predict some unknown or future values of other variables*

• Descriptive Methods
  • *Find human –interpretable patterns that describe the data*

• Supervised (outcomes or labels provided) vs. Unsupervised (just data)
What can we do with Data Mining?

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

- **Science: Chemistry, Physics, Medicine, Energy**
  - 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 Tasks

- Concept/Class description: Characterization and discrimination
  - Generalize, summarize, and contrast data characteristics, e.g., dry vs. wet regions; “normal” vs. fraudulent behavior

- Association (correlation and causality)
  - Multi-dimensional interactions and associations
    age(X, “20-29”) ^ income(X, “60-90K”) \(\rightarrow\) buys(X, “TV”)
    Hospital(area code) ^ procedure(X) -> claim (type) ^ claim(cost)
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, energy usage based on sensor data
  - 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 claims or providers to find distribution patterns of unusual behavior
  • 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

1. Database
2. Selection Transformation
3. Data Preparation
4. Training Data
5. Data Mining
6. Model, Patterns
7. Evaluation, Verification
Steps of a KDD Process (1)

- Learning the application domain:
  - relevant prior knowledge and goals of application
- Creating a target data set: data selection
- Data cleaning and preprocessing: (may take 60% of effort!)
- Data reduction and transformation:
  - Find useful features, dimensionality/variable reduction, representation
- Choosing functions of data mining
  - summarization, classification, regression, association, clustering
Steps of a KDD Process (2)

- Choosing functions of data mining
  - summarization, classification, regression, association, clustering
- Choosing the mining algorithm(s)
- Data mining: search for patterns of interest
- Pattern evaluation and knowledge presentation
  - visualization, transformation, removing redundant patterns, etc.
- Use and integration of discovered knowledge
Learning and Modeling Methods

- Decision Tree Induction (C4.5, J48)
- Regression Tree Induction (CART, MP5)
- Multivariate Adaptive Regression Splines (MARS)
- Clustering (K-means, EM, Cobweb)
- Artificial Neural Networks (Backpropagation, Recurrent)
- Support Vector Machines (SVM)
- 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
Decision Tree Example

![Decision Tree Diagram](image)
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
Clustering
Kmeans Results from 45,000 NYTimes articles

7 viable clusters found
Artificial Neural Networks (ANNs)

- Network of many simple units
- Main Components
  - Inputs
  - Hidden layers
  - Outputs
- Adjusting weights of connections
- Backpropagation
Evaluation

- Error on the training data vs. performance on future/unseen data
- Simple solution
  - Split data into training and test set
  - Re-substitution error
    - error rate obtained from the training data
- Three sets
  - training data, validation data, and test data
Training and Testing

• **Test set**
  • set of independent instances that have not been used in formation of classifier in any way

• **Assumption**
  • data contains representative samples of the underlying problem

• **Example: classifiers built using customer data from two different towns A and B**
  • To estimate performance of classifier from town in completely new town, test it on data from B
Error Estimation Methods

- Cross-validation
  - Partition in K disjoint clusters
  - Train k-1, test on remaining
- Leave-one-out Method
- Bootstrap
  - Sampling with replacement
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
Model Recommendations

- Usually no absolute choice and no silver bullets (otherwise we wouldn’t be here)
- Start with simple methods
- Consider trade off as you go more complex
- Find similar application examples (what works in this domain)
- Find paradigmatic examples for models (what works for this model)
- Goals and Expectations!
Available Data Mining Tools

COTs:
- IBM Intelligent Miner
- SAS Enterprise Miner
- Oracle ODM
- Microstrategy
- Microsoft DBMiner
- Pentaho
- Matlab
- Teradata

Open Source:
- WEKA
- KNIME
- Orange
- RapidMiner
- NLTK
- R
- Rattle
On the Importance of Data Prep

- “Garbage in, garbage out”
- A crucial step of the DM process
- Could take 60-80% of the whole data mining effort
Preprocessing Input/Output

• **Inputs:**
  • raw data

• **Outputs:**
  • two data sets: training and test (if available)
  • Training further broken into training and validation
Variables and Features terms

- Variables and their transformations are features
- Instance labels are outcomes or dependent variables
- Set of instances comprise the input data matrix
  - Often represented as a single flat file
- Data Matrix size is often refer by N observations and P variables
  - Large N Small P => usually solvable
  - Small N Large P => not always solvable directly, needs heuristics
Types of Measurements

- Nominal (names)
- Categorical (zip codes)

- Ordinal (H,M,L)
- Real Numbers
  - May or may not have a Natural Zero Point?
    - If not comparisons are OK but not multiplication (e.g. dates)

Qualitative (unordered, non-scalar)

Quantitative (ordered, scalar)
Know variable properties

- Variables as objects of study
  - explore characteristics of each variable:
    - typical values, min, max, range etc.
    - entirely empty or constant variables can be discarded
  - explore variable dependencies

- Sparsity
  - missing, N/A, or 0?

- Monotonicity
  - increasing without bound, e.g. dates, invoice numbers
  - new values not in the training set
Data Integration Issues

- When combining data from multiple sources:
  - integrate metadata (table column properties)
  - the same attribute may have different names in different databases
    e.g. A.patient-id ≡ B.patient-number

  Using edit-distance can help resolve entities

- Detecting and resolving data value conflicts
  e.g. attribute values may have different scales or nominal labels
Noise in Data

• Noise is unknown error source
  • often assumed random Normal distribution
  • often assumed independent between instances

• Approaches to Address Noise
  • Detect suspicious values and remove outliers
  • Smooth by averaging with neighbors
  • Smooth by fitting the data into regression functions
Missing Data

• Important to review statistics of a missing variable
  • Are missing cases random?
  • Are missing cases related to some other variable?
  • Are other variables missing data in same instances?
  • Is there a relation between missing cases and outcome variable?
  • What is frequency of missing cases wrt all instances?
Approaches to Handle Missing Data

- Ignore the tuple: usually done when class label is missing or if there’s plenty of data
- Use a global constant to fill in (‘impute’) the missing value: (or a special “unknown” value for nominals)
- Use the attribute mean to impute the missing value
- Use the attribute mean to impute all samples belonging to the same class
- Use an algorithm to impute missing value
- Use separate models with/without values
Variable Enhancements

- Make analysis easy for the tool
  - if you know how to deduce a feature, do it yourself and don’t make the tool find it out
  - to save time and reduce noise
  - include relevant domain knowledge
- Getting enough data
  - Do the observed values cover the whole range of data?
Data Transformation: Normalizations

- Mean center
  \[ x_{\text{new}} = x - \text{mean}(x) \]

- z-score
  \[ z - \text{score} = \frac{x - \text{mean}(x)}{\text{std}(x)} \]

- Scale to [0…1]
  \[ x_{\text{new}} = \frac{x - \text{min}(x)}{\text{max}(x) - \text{min}(x)} \]

- log scaling
  \[ x_{\text{new}} = \log(x) \]
Variable Transformation Summary

- Smoothing: remove noise from data
- Aggregation: summarization
- Introduce/relabel/categorize variable values
- Normalization: scaled to fall within a small, specified range
- Attribute/feature construction
Pause
Data Exploration and Unsupervised Learning with Clustering

Paul F Rodriguez, PhD
San Diego Supercomputer Center
Predictive Analytic Center of Excellence
Clustering Idea

- Given a set of data can we find a natural grouping?
Why Clustering

- A good grouping implies some structure
- In other words, given a good grouping, we can then:
  - Interpret and label clusters
  - Identify important features
  - Characterize new points by the closest cluster (or nearest neighbors)
  - Use the cluster assignments as a compression or summary of the data
Clustering Objective

- Objective: find subsets that are similar within cluster and dissimilar between clusters
- Similarity defined by distance measures
  - Euclidean distance
  - Manhattan distance
  - Mahalanobis
    (Euclidean w/dimensions rescaled by variance)
Kmeans Clustering

• A simple, effective, and standard method
  Start with K initial cluster centers
  Loop:
    Assign each data point to nearest cluster center
    Calculate mean of cluster for new center
  Stop when assignments don’t change

• Issues:
  How to choose K?
  How to choose initial centers?
  Will it always stop?
**Kmeans Example**

- For K=1, using Euclidean distance, where will the cluster center be?

![Diagram showing a scatter plot with two axes, X1 and X2, and a distribution of points]
Kmeans Example

- For $K=1$, the overall mean minimizes Sum Squared Error (SSE), aka Euclidean distance
**Kmeans Example**

As K increases individual points get a cluster.
Choosing K for Kmeans

- Not much improvement after K=2 (“elbow”)
Kmeans Example – more points

How many clusters should there be?
Choosing K for Kmeans

- Smooth decrease at $K \geq 2$, harder to choose
- In general, smoother decrease $\Rightarrow$ less structure
Kmeans Guidelines

• Choosing K:
  • “Elbow” in total-within-cluster SSE as K=1…N
  • Cross-validation: hold out points, compare fit as K=1…N

• Choosing initial starting points:
  • take K random data points, do several Kmeans, take best fit

• Stopping:
  • may converge to sub-optimal clusters
  • may get stuck or have slow convergence (point assignments bounce around), 10 iterations is often good
Kmeans Example uniform

K=1

K=2

K=3

K=4
Choosing $K$ - uniform

- Smooth decrease across $K$ => less structure
**Kmeans Clustering Issues**

- **Deviations:**
  - Dimensions with large numbers may dominate distance metrics
  - Kmeans doesn’t model group variances explicitly (so groups with unequal variances may get blurred)

- **Outliers:**
  - Outliers can pull cluster mean, K-mediods uses median instead of mean
Soft Clustering Methods

• Fuzzy Clustering
  • Use weighted assignments to all clusters
  • Weights depend on relative distance
  • Find min weighted SSE

• Expectation-Maximization:
  • Mixture of multivariate Gaussian distributions
  • Mixture weights are ‘missing’
  • Find most likely means & variances, for the expectations of the data given the weights
Kmeans – unequal cluster variance
**Choosing K – unequal distributions**

- Smooth decrease across K => less structure
EM clustering

- Selects K=2 (using Bayesian Information Criterion)
- Handles unequal variance
Kmeans computations

- **Distance of each point to each cluster center**
  - For $N$ points, $D$ dimensions: each loop requires $N \times D \times K$ operations

- **Update Cluster centers**
  - only track points that change, get change in cluster center

- **On HPC:**
  - Distance calculations can be partitioned data across dimension
Dimensionality Reduction via Principle Components

• Idea: Given $N$ points and $P$ features (aka dimensions), can we represent data with fewer features:
  • Yes, if features are constant
  • Yes, if features are redundant
  • Yes, if features only contribute noise (conversely, want features that contribute to variations of the data)
Dimensionality Reduction via Principle Components

- **PCA:**
  - Find set of vector (aka factors) that describe data in alternative way
  - First component is the vector that maximizes the variance of data projected onto that vector
  - K-th component is orthogonal to all k-1 previous components
PCA on 2012 Olympic Athletes’ Height by Weight scatter plot

Idea:
Can you rotate the axis so that the data lines up on one axis as much as possible?

Or, find the direction in space with maximum variance of the data?
PCA on 2012 Olympic Athletes’ Height by Weight scatter plot

Total Variance Conserved:
\[ \text{Var in Weight} + \text{Var in Height} = \text{Var in PC1} + \text{Var in PC2} \]

In general:
\[ \text{Var in PC1} > \text{Var in PC2} > \text{Var in PC3} \ldots \]

Projection of (145,5) to PCs

Weight- Kg (mean centered)

Height- cm (mean centered)
**SVD (=PCA for non-square matrix)**

- Informally, \( X_{NxP} = U_{NxP} X S_{PxP} X V_{PxP} \)

- Taking \( k < P \) \( => \) \( X_{NxP} \sim U_{NxK} X S_{KxK} X V_{KxP} \)
  - \( U \) corresponds to observations in \( k \)-dimensional factor space
    (eg each row in \( U \) is a point in \( K \) dimensional space, AKA factor loadings for data)
  - \( V' \) corresponds to measurements in \( K \) factor space
  - (eg each row in \( V' \) is a point in \( k \)-dim. factor space, AKA factor loadings for independent variable)
  - \( V \) corresponds to factors in original space
    (eg each row in \( V \) is a point in \( P \) dimensional data space)
**Principle Components**

- Can choose $k$ heuristically as approximation improves, or choose $k$ so that 95% of data variance accounted.
- aka Singular Value Decomposition
  - PCA on square matrices only
  - SVD, more general
- Works for numeric data only
- In contrast, clustering reduces to categorical groups
- In some cases, $k$ PCs $\Leftrightarrow k$ clusters
Summary

• Having no label doesn’t stop you from finding structure in data

• Unsupervised methods are somewhat related
Earth System Sciences Example

  (e.g. Quest_Out_Data.csv)

• Summary: “…identify patterns and relationships… anomalous data … promote resource assessment and exploration”

• Data: 15,000 rows x 42 measures (and metadata)
Data location and spreadsheet
N=15,000 x P=42 correlation plot
Standard Deviation of Columns

![Graph showing standard deviation of columns]
Kmeans SSE as K=1..30

K=1…30; Authors used 20
$K=20$, heat map of centroids
Try Using Matrix Factorization (SVD), use 3 factors, then take Kmeans on those

K=1…30, Data is N=15000 x 3 factors
Plots of all data points on 2 factors at a time (the U matrix from svd())
(you could also plot V matrix)
PC1xPC2 with Cluster Coded
PC1xPC3 with Cluster Coded
**Kmeans vs SiroSOM w/Kmeans**

Fraser: used SOM (self organizing map, a non-linear, interactive matrix factorization technique), then Kmeans

Kmeans, k=20 on raw data gives 58% overlap with article

3 SVD factors, then Kmeans k=20 gives 29% overlap

10 SVD factors, then Kmeans K=20 gives 38% overlap
Plotting in Physical Space

- Fraser (l) cluster assignment vs Kmeans alone (r)
End

• If time, hierarchical clustering next
Incremental & Hierarchical Clustering

• Start with 1 cluster (all instances) and do splits
  OR
  Start with N clusters (1 per instance) and do merges

• Can be greedy & expensive in its search
  some algorithms might merge & split
  algorithms need to store and recalculate distances

• Need distance between groups
  in contrast to K-means
Incremental & Hierarchical Clustering

- Result is a hierarchy of clusters
  - displayed as a ‘dendrogram’ tree

- Useful for tree-like interpretations
  - syntax (e.g. word co-occurences)
  - concepts (e.g. classification of animals)
  - topics (e.g. sorting Enron emails)
  - spatial data (e.g. city distances)
  - genetic expression (e.g. possible biological networks)
  - exploratory analysis
Incremental & Hierarchical Clustering

- Clusters are merged/split according to distance or utility measure
  - Euclidean distance (squared differences)
  - conditional probabilities (for nominal features)

- Options to choose which clusters to ‘Link’
  - single linkage, mean, average (w.r.t. points in clusters)
    (may lead to different trees, depending on spreads)
  - Ward method (smallest increase within cluster variance)
  - change in probability of features for given clusters
**Linkage options**

• e.g. single linkage (closest to any cluster instance)

• e.g. mean (closest to mean of all cluster instances)
Linkage options (cont’)

• e.g. average (mean of pairwise distances)

• e.g. Ward’s method (find new cluster with min. variance)
Hierarchical Clustering Demo

• 3888 Interactions among 685 proteins
  From Hu et.al. TAP dataset http://www.compsysbio.org/bacteriome/dataset/)

b0009b00140.92
b0009b22310.87
b0014b01691.0
b0014b05950.76
b0014b26141.0
b0014b33390.95
b0014b36360.9
b0015b00140.99
Hierarchical Clustering Demo

- Essential R commands:

  ```r
  > d = read.table("hu_tap_ppi.txt"),
  > str(d)  # show d structure
  'data.frame': 3888 obs. of 3 variables:
  $ V1: Factor w/ 685 levels "b0009","b0014",...:
  1 1 2 2 2 2 2 3 3 3 3 ...
  $ V2: Factor w/ 536 levels "b0011","b0014",...:
  2 248 28 66 297 396 ...
  $ V3: num 0.92 0.87 1 0.76 1 0.95 0.9 0.99 0.99 0.93 ...
  > fs = c(d[,1],d[,2]);  # combine factor levels
  > str(fs)
  int [1:7776] 1 1 2 2 2 2 3 3 3 ...
  ```

  Note: strings read as “factors”
Hierarchical Clustering Demo

Essential R commands:

```r
C = matrix(0, P, P);  # Connection matrix (aka Adjacency matrix)
IJ = cbind(d[,1], d[,2])  # Factor level is saved as Nx2 list of i-th,j-th protein
for (i in 1:N) {C[IJ[i,1], IJ[i,2]] = 1;}  # Populate C with 1 for connections

install.packages('igraph')
library('igraph')
gc = graph.adjacency(C, mode = "directed")
plot.graph(gc, vertex.size = 3, edge.arrow.size = 0, vertex.label = NA)

or just plot( ....
```
Hierarchical Clustering Demo

- hclust with “single” distance: chaining

```
d2use = dist(C, method = "binary")
fit <- hclust(d2use, method = "single")
plot(fit)
```
Hierarchical Clustering Demo

- hclust with “Ward” distance: spherical clusters
Hierarchical Clustering Demo

- Where height change looks big, cut off tree

groups <- cutree(fit, k=7)
rect.hclust(fit, k=7, border="red")

Cluster Dendrogram
Hierarchical Clustering Demo

- **Kmeans vs Hierarchical:**
  
  Lots of overlap despite that Kmeans not have ‘binary’ distance option

<table>
<thead>
<tr>
<th>Kmeans cluster assignment</th>
<th>Hierarchical Group Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>26</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>72</td>
</tr>
<tr>
<td>...</td>
<td></td>
</tr>
</tbody>
</table>

```r
groups <- cutree(fit, k=7)
Kresult = kmeans(d2use, 7, 10)
table(Kresult$cluster, groups)
```