Strategies for Building Predictive Models

Dean Abbott
Abbott Analytics, Inc.
KNIME User Group Meeting
February 14, 2014

Email: dean@abbottanalytics.com
Blog: http://abbottanalytics.blogspot.com
Twitter: @deanabb
Instructor - Dean Abbott

• Education
  • Master of Applied Mathematics, University of Virginia
  • B.S. Computational Mathematics, Rensselaer Polytechnic Institute

• Applied Data Mining for 25+ years in
  • Tax Compliance, Fraud Detection
  • Text Mining and Concept Classification
  • Direct Marketing, CRM, Survey Analysis, Market Basket Analysis
  • Predictive Toxicology, Biological Risk Assessment
• Earlier
  • Signal and Image Processing, Guidance and Control
  • Optical Character Recognition & Postnet Bar Code Readers

• Data Mining Course Instruction
  • Taught dozens of short courses, conference tutorials, lectures, in-house custom courses
Strategies

1. Know what we are doing
2. Have a plan for the project
3. Assess models the way you want to use them
4. Do for the algorithms what they cannot do for themselves
5. Deploy models wisely
Strategy 1: Know What You are Doing

- What is Predictive Analytics?
- How does PA differ from
  - Statistics
  - BI
  - Big Data
What is Predictive Analytics?

- Wikipedia Definitions
  - Predictive analytics is an area of statistical analysis that deals with extracting information from data and uses it to predict future trends and behavior patterns.
  - The core of predictive analytics relies on capturing relationships between explanatory variables and the predicted variables from past occurrences, and exploiting it to predict future outcomes.
What is Predictive Analytics?

• Other Definitions (in the news and blogs)
  • Predictive Analytics is emerging as a game-changer. Instead of looking backward to analyze "what happened?" predictive analytics help executives answer "What's next?" and "What should we do about it?" (Forbes Magazine, April 1, 2010)
  • Predictive analytics is the branch of data mining concerned with the prediction of future probabilities and trends. (searchcrm.com)
  • Predictive Analytics *is* data mining re-badged because too many people were claiming to do data mining and weren't. (Tim Manns paraphrasing Wayne Erickson of TDWI)
What is Predictive Analytics?
Simple Definitions

- *Data driven* analysis for [large] data sets
  - Data-driven to discover input combinations
  - Data-driven to validate models

- *Automated* pattern discovery
  - Key input variables
  - Key input combinations
# Statistics vs. Predictive Analytics

<table>
<thead>
<tr>
<th>View of the &quot;other&quot; field</th>
<th>Statistics</th>
<th>Predictive Analytics</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;data dredging&quot;</td>
<td>&quot;we can do <em>that</em> ... and more!&quot;</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Emphasis</th>
<th>Theory; Optimum Solutions</th>
<th>&quot;Good&quot; Heuristics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Approach</td>
<td>Parametric</td>
<td>Non-parametric</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Key Metrics of Performance</th>
<th>Statistics</th>
<th>Predictive Analytics</th>
</tr>
</thead>
<tbody>
<tr>
<td>R^2, p-values, S.E.</td>
<td>Model</td>
<td>Data</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>What is King?</th>
<th>Statistics</th>
<th>Predictive Analytics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model</td>
<td>Data</td>
<td></td>
</tr>
</tbody>
</table>

# Business Intelligence vs. Predictive Analytics

<table>
<thead>
<tr>
<th>View of the &quot;other&quot; field</th>
<th>Business Intelligence</th>
<th>Predictive Analytics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Emphasis</td>
<td>“we’re the foundation, (they’re so complicated!)”</td>
<td>“they report the past, we predict the future!”</td>
</tr>
<tr>
<td>Approach</td>
<td>What happened?</td>
<td>What do we think will happen?</td>
</tr>
<tr>
<td>Key Metrics of Performance</td>
<td>User-driven Reporting</td>
<td>Algorithms, Searching</td>
</tr>
<tr>
<td>What is King?</td>
<td>KPIs</td>
<td>Lift, ROC</td>
</tr>
<tr>
<td></td>
<td>Data (via Analyst)</td>
<td>Data (via Algorithms)</td>
</tr>
</tbody>
</table>
Rexer Analytics Survey (2013): Predictive Analytics Algorithms

The number of algorithms used varies by the labels people use to describe themselves, with Data Miners (14) and Data Scientists (14) using the most, and Software Developers (9) and Programmers (8) the fewest.
Strategy 2: Have a Plan

- Use CRISP-DM
  - Or similar framework
- Don’t be too strict
  - These are suggested steps, not recipes
What do Predictive Modelers do? The CRISP-DM Process Model

- **CRoss-Industry Standard Process Model for Data Mining**
- Describes Components of Complete Data Mining Cycle from the Project Manager’s Perspective
- Shows Iterative Nature of Data Mining
CRISP-DM: Business Understanding Steps

- Ask Relevant Business Questions
- Determine Data Requirements to Answer Business Question
- Translate Business Question into Appropriate Data Mining Approach
- Determine Project Plan for Data Mining Approach

**Define Business Objectives**

**Assess Situation**

**Determine Data Mining Objectives**

**Produce Project Plan**

**Background**

**Inventory of Resources**

**Data Mining Goals**

**Project Plan**

**Business Objectives**

**Requirements, Assumptions, Constraints**

**Data Mining Success Criteria**

**Initial Assessment of Tools & Techniques**

**Terminology**

**Costs and Benefits**

**Business Success Criteria**

© Abbott Analytics, Inc. 2001-2013
CRISP-DM Step 2: Data Understanding Steps

- Collect initial data
  - Internal data: historical customer behavior, results from previous experiments
  - External data: demographics & census, other studies and government research
  - Extract superset of data (rows and columns) to be used in modeling
  - Identify form of data repository: multiple vs. single table, flat file vs. database, local copy vs. data mart

- Perform Preliminary Analysis
  - Characterize Data (describe, explore, verify)
  - Condition Data
CRISP-DM Step 3: Data Preparation (Conditioning) Steps

Fix Data Problems
- Select Data
- Clean Data

Create Features
- Construct Data
- Integrate Data
- Format Data
- Rationale for Inclusion/Exclusion
- Data Cleaning Report
- Derived Attributes
- Generated Records
- Merged Data
- Reformatted Data
CRISP-DM Step 4: Modeling Steps

- **Algorithm Selection**
  - Select Modeling Techniques
  - Modeling Techniques
  - Modeling Assumptions

- **Sampling**
  - Generate Test Design
  - Test Design

- **Algorithms**
  - Build Model
  - Parameter Settings
  - Models
  - Model Description

- **Model Ranking**
  - Assess Model
  - Model Assessment
  - Revised Parameter Settings
CRISP-DM vs. SEMMA

- **CRISP-DM**
  Six Steps

- **SEMMA**
  Five Steps

<table>
<thead>
<tr>
<th>CRISP</th>
<th>SEMMA</th>
<th>Nayak &amp; Qiu</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business understanding</td>
<td>Assumes well-defined question</td>
<td>Goals were defined</td>
</tr>
<tr>
<td>Data understanding</td>
<td>Sample</td>
<td>Develop tools to better utilize</td>
</tr>
<tr>
<td>Data preparation</td>
<td>Explore</td>
<td>problem reports</td>
</tr>
<tr>
<td>Modeling</td>
<td>Modify data</td>
<td>Looked at data in problem reports</td>
</tr>
<tr>
<td>Evaluation</td>
<td>Model</td>
<td>Data pre-processing</td>
</tr>
<tr>
<td>Deployment</td>
<td>Assess</td>
<td>Data cleaning</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data transformation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Data modeling</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Analyzing results</td>
</tr>
</tbody>
</table>

Table 2.3. Comparison of methods

Table from Advanced Data Mining Techniques, Olsen and Delen, Springer, 2008
Strategy 3: Assess Models the Way you Use Them

- Standard Assessment Methods
  - Batch Methods
  - Rank-Ordered Methods

- Why the Method Matters
  - Outliers
  - Sampling and Accuracy
### Classification Accuracy
from Decision Thresholds

- If $P(\text{Target}_B = 1)$ is greater than a pre-defined threshold, the prediction is $\text{Target}_B = 1$.

- If the prediction matches the actual $\text{Target}_B$ value, the decision is “correct”. Otherwise it is wrong.

- With the threshold of 0.05,
  - first 17 records are above the threshold
  - 9 records have “correct” predictions
  - 8 records have “incorrect” predictions

<table>
<thead>
<tr>
<th>$P(\text{Target}_B = 1)$</th>
<th>CONTROLN</th>
<th>LastGift</th>
<th>$\text{TARGET}_B$</th>
<th>$P(\text{Target}_B = 0)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0731</td>
<td>185436</td>
<td>0</td>
<td>0</td>
<td>0.9269</td>
</tr>
<tr>
<td>0.0715</td>
<td>14279</td>
<td>1</td>
<td>1</td>
<td>0.9285</td>
</tr>
<tr>
<td>0.0699</td>
<td>727</td>
<td>2</td>
<td>1</td>
<td>0.9301</td>
</tr>
<tr>
<td>0.0683</td>
<td>24610</td>
<td>3</td>
<td>1</td>
<td>0.9317</td>
</tr>
<tr>
<td>0.0668</td>
<td>22645</td>
<td>4</td>
<td>1</td>
<td>0.9332</td>
</tr>
<tr>
<td>0.0653</td>
<td>82943</td>
<td>5</td>
<td>1</td>
<td>0.9347</td>
</tr>
<tr>
<td>0.0639</td>
<td>108412</td>
<td>6</td>
<td>0</td>
<td>0.9361</td>
</tr>
<tr>
<td>0.0624</td>
<td>190313</td>
<td>7</td>
<td>1</td>
<td>0.9376</td>
</tr>
<tr>
<td>0.0611</td>
<td>48804</td>
<td>8</td>
<td>0</td>
<td>0.9389</td>
</tr>
<tr>
<td>0.0597</td>
<td>123822</td>
<td>9</td>
<td>1</td>
<td>0.9403</td>
</tr>
<tr>
<td>0.0583</td>
<td>94039</td>
<td>10</td>
<td>0</td>
<td>0.9417</td>
</tr>
<tr>
<td>0.0570</td>
<td>47605</td>
<td>11</td>
<td>0</td>
<td>0.9430</td>
</tr>
<tr>
<td>0.0558</td>
<td>25641</td>
<td>12</td>
<td>1</td>
<td>0.9442</td>
</tr>
<tr>
<td>0.0545</td>
<td>47476</td>
<td>13</td>
<td>0</td>
<td>0.9455</td>
</tr>
<tr>
<td>0.0533</td>
<td>6023</td>
<td>14</td>
<td>0</td>
<td>0.9467</td>
</tr>
<tr>
<td>0.0521</td>
<td>47784</td>
<td>15</td>
<td>0</td>
<td>0.9479</td>
</tr>
<tr>
<td>0.0509</td>
<td>148569</td>
<td>16</td>
<td>1</td>
<td>0.9491</td>
</tr>
<tr>
<td>0.0497</td>
<td>171099</td>
<td>17</td>
<td>0</td>
<td>0.9503</td>
</tr>
</tbody>
</table>

**If** ($P(\text{Target}_B = 1) > 0.05$)

**Then** 1

**Else** 0
## Typical Binary Classification Accuracy Metrics

### Confusion Matrix

<table>
<thead>
<tr>
<th>Actual Class</th>
<th>Predicted Class</th>
<th>Total Actual (down)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 (actual value is negative)</td>
<td>0 (predicted value is negative)</td>
<td>( t_n ) (true negative)</td>
</tr>
<tr>
<td>1 (actual value is positive)</td>
<td>1 (predicted value is positive)</td>
<td>( f_n ) (false negative, false dismissal)</td>
</tr>
<tr>
<td>Total Predicted (across)</td>
<td>Total negative predictions ( t_n + f_p )</td>
<td>Total positive predictions ( t_p + f_n )</td>
</tr>
</tbody>
</table>

### Accuracy Metrics

- **PCC** (Probability Correct Classification): 
  \[
PCC = \frac{t_p + t_n}{t_p + t_n + f_p + f_n}
\]

- **Precision**:
  \[
  \text{Precision} = \frac{t_p}{t_p + f_p}
  \]

- **Recall**:
  \[
  \text{Recall} = \frac{t_p}{t_p + f_n}
  \]

- **False Alarm Rate (FA)**:
  \[
  \text{False Alarm Rate (FA)} = \frac{f_p}{t_n + f_p}
  \]

- **False Dismissal Rate (FD)**:
  \[
  \text{False Dismissal Rate (FD)} = 1 - \text{Recall} = \frac{f_n}{t_p + f_n}
  \]

- **Sensitivity**:
  \[
  \text{Sensitivity} = \text{Recall} = \frac{t_p}{t_p + f_n}
  \]

- **Specificity** (True Negative Rate):
  \[
  \text{Specificity} = \text{True Negative Rate} = \frac{t_n}{t_n + f_p}
  \]

- **Type I Error**:
  \[
  \text{Type I Error} = \frac{f_p}{t_p + t_n + f_p + f_n}
  \]

- **Type II Error**:
  \[
  \text{Type II Error} = \frac{f_n}{t_p + t_n + f_p + f_n}
  \]
## Percent Correct Classification (PCC)

\[
PCC = \frac{(t_n + t_p)}{(t_p + t_n + f_p + f_n)}
\]

### Confusion Matrix

<table>
<thead>
<tr>
<th>Actual Class</th>
<th>Predicted Class</th>
<th>Total Actual (down)</th>
<th>Total Predicted (across)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 (actual value is negative)</td>
<td>0 (predicted value is negative)</td>
<td>$t_n$ (true negative)</td>
<td>Total negative predictions $tn + fn$</td>
</tr>
<tr>
<td></td>
<td>1 (predicted value is positive)</td>
<td>$f_p$ (false positive, false alarm)</td>
<td>Total positive predictions $tp + fp$</td>
</tr>
<tr>
<td>1 (actual value is positive)</td>
<td></td>
<td>$f_n$ (false negative, false dismissal)</td>
<td>Total examples $tp + tn + fp + fn$</td>
</tr>
</tbody>
</table>
Batch vs. Rank-Ordered Approaches to Model Evaluation

- “Batch” approaches
  - Score every record with the same weight
  - Provide a summary for the entire scored data set
    - PCC, Precision, Recall, Type I Errors, Type II Errors, etc.

- Rank-ordered approaches
  - Sort population by model scores, high to low
  - Accumulate a metric as one goes down the ordered file
  - Reports results by group, typically deciles, demi-deciles, percentiles, etc.
  - Examples
    - Lift, Gains, ROC, Profit
## Three Key Rank-Ordered Metrics

<table>
<thead>
<tr>
<th>Measure</th>
<th>What is Measures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gain</td>
<td>% of target records found</td>
</tr>
<tr>
<td>Lift</td>
<td>ratio of target gain to average response rate</td>
</tr>
<tr>
<td>ROC</td>
<td>change classifier probability threshold from 0 to 1; sensitivity vs. 1 - specificity for each threshold</td>
</tr>
</tbody>
</table>
Gains Charts and Lift Curves

- Number Respondants are Rank-ordered (sorted) by predicted values
- X axis is the percentage of records as go down file.
- **Gain** is the pct. of target=1 found at indicated file depth
- **Lift ratio** is how many times more respondants at given customer depth compared to random selection
- Random **gain** has slope equal to proportion of respondants in training data
- Random **lift** is 1.0
ROC Curves

Sensitivity (True Alerts)

1 – Specificity (False Alarms)
Profit / ROI Charts

- **Profit**
  - \[ \text{Profit} = \text{Revenue} - \text{Cost} \]
  - Cumulative Measure
  - Increases as long as ROI > 0 in segments
  - Negative \( \Rightarrow \) campaign loses money

- **ROI Percentage**
  - \[ \text{ROI Percentage} = \frac{\text{Profit}}{\text{Cost}} \]
  - or \( \frac{\text{Revenue}}{\text{Cost}} - 1 \)
  - Cumulative Measure
  - Negative \( \Rightarrow \) campaign loses money
Area Under the (ROC) Curve (AUC)

- AUC = 1.0 is a perfect model
- AUC = 0.5 is a random model
- Gini Index = \(2 \times \text{AUC} - 1\)
  or
  \[
  \text{AUC} = \frac{(\text{Gini Index} + 1)}{2}
  \]

\[G_1 = 2 \times \text{AUC} - 1\]
## Algorithm Objectives

- Linear Regression and Neural networks minimize squared error
- C5 maximizes Information Gain
- CART maximizes Gini index
- Logistic regression maximizes the log of the odds of the probability the record belongs to class “1” (classification accuracy)
- Nearest neighbor minimizes Euclidean distance

## Sample Business Objectives

- Maximize net revenue
- Contact max # customers to achieve response rate of 13%
- Maximize responders subject to a budget of $100,000
- Maximize recovered revenue from customers likely to churn
- Maximize collected revenue by identifying next best case to collect
- Minimize false alarms in 100 transactions most likely to be fraudulent
PAKDD Cup 2007 Results:
Score Metric Changes Winner

- Correlation of AUC rank with top Decile Rank: 0.76

<table>
<thead>
<tr>
<th>Modeling Technique/Implementation</th>
<th>Participant Affiliation Location</th>
<th>Participant Affiliation Type</th>
<th>AUCROC (Trapezoidal Rule) Rank</th>
<th>Top Decile Response Rate Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>TreeNet + Logistic Regression</td>
<td>Salford Systems</td>
<td>Mainland China Practitioner</td>
<td>70.01%</td>
<td>7</td>
</tr>
<tr>
<td>Probit Regression</td>
<td>SAS USA Practitioner</td>
<td>69.99%</td>
<td>13.00%</td>
<td>10</td>
</tr>
<tr>
<td>MLP + n-Tuple Classifier</td>
<td>Brazil Practitioner</td>
<td>69.62%</td>
<td>13.88%</td>
<td>1</td>
</tr>
<tr>
<td>TreeNet</td>
<td>Salford Systems USA Practitioner</td>
<td>69.61%</td>
<td>13.25%</td>
<td>4</td>
</tr>
<tr>
<td>Ridge Regression</td>
<td>Rank Belgium Practitioner</td>
<td>69.28%</td>
<td>12.88%</td>
<td>9</td>
</tr>
<tr>
<td>2-Layer Linear Regression</td>
<td>USA Practitioner</td>
<td>69.14%</td>
<td>12.88%</td>
<td>9</td>
</tr>
<tr>
<td>Log Regr + Decision Stump + AdaBoost + VFI</td>
<td>Mainland China Academia</td>
<td>69.10%</td>
<td>13.25%</td>
<td>4</td>
</tr>
<tr>
<td>Logistic Average of Single Decision Functions</td>
<td>Australia Practitioner</td>
<td>68.85%</td>
<td>12.13%</td>
<td>17</td>
</tr>
<tr>
<td>Logistic Regression</td>
<td>Weka Singapore Academia</td>
<td>68.69%</td>
<td>12.38%</td>
<td>16</td>
</tr>
<tr>
<td>Logistic Regression</td>
<td>Mainland China Practitioner</td>
<td>68.58%</td>
<td>12.88%</td>
<td>9</td>
</tr>
<tr>
<td>Decision Tree + Neural Network + Logistic Regression</td>
<td>Singapore</td>
<td>68.54%</td>
<td>13.00%</td>
<td>7</td>
</tr>
<tr>
<td>Scorecard Linear Additive Model</td>
<td>Xeno USA Practitioner</td>
<td>68.28%</td>
<td>11.75%</td>
<td>20</td>
</tr>
<tr>
<td>Random Forest</td>
<td>Weka USA</td>
<td>68.04%</td>
<td>12.50%</td>
<td>14</td>
</tr>
<tr>
<td>Expanding Regression Tree + RankBoost + Bagging</td>
<td>Weka Mainland China Academia</td>
<td>68.02%</td>
<td>12.50%</td>
<td>14</td>
</tr>
<tr>
<td>Logistic Regression</td>
<td>SAS + Salford India Practitioner</td>
<td>67.58%</td>
<td>12.00%</td>
<td>19</td>
</tr>
<tr>
<td>J48 + BayesNet</td>
<td>Weka Mainland China Academia</td>
<td>67.56%</td>
<td>11.63%</td>
<td>21</td>
</tr>
<tr>
<td>Neural Network + General Additive Model</td>
<td>Tiberius USA Practitioner</td>
<td>67.54%</td>
<td>11.63%</td>
<td>21</td>
</tr>
<tr>
<td>Decision Tree + Neural Network</td>
<td>Mainland China Academia</td>
<td>67.50%</td>
<td>12.88%</td>
<td>2</td>
</tr>
<tr>
<td>Decision Tree + Neural Network + Log. Regression</td>
<td>SAS USA Academia</td>
<td>66.71%</td>
<td>13.50%</td>
<td>2</td>
</tr>
</tbody>
</table>

http://lamda.nju.edu.cn/conf/pakdd07/dmc07/results.htm
Model Comparison:
Different Metrics Tell Different Stories

<table>
<thead>
<tr>
<th>Model Number</th>
<th>Model ID</th>
<th>AUC</th>
<th>Train RMS</th>
<th>Test RMS</th>
<th>AUC Rank</th>
<th>Train RMS Rank</th>
<th>Test RMS Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>NeuralNet1032</td>
<td>73.3%</td>
<td>0.459</td>
<td>0.370</td>
<td>9</td>
<td>53</td>
<td>1</td>
</tr>
<tr>
<td>39</td>
<td>NeuralNet303</td>
<td>72.4%</td>
<td>0.477</td>
<td>0.374</td>
<td>42</td>
<td>59</td>
<td>2</td>
</tr>
<tr>
<td>36</td>
<td>NeuralNet284</td>
<td>75.0%</td>
<td>0.458</td>
<td>0.376</td>
<td>2</td>
<td>52</td>
<td>3</td>
</tr>
<tr>
<td>31</td>
<td>NeuralNet244</td>
<td>72.7%</td>
<td>0.454</td>
<td>0.386</td>
<td>33</td>
<td>49</td>
<td>4</td>
</tr>
<tr>
<td>57</td>
<td>CVLinReg2087</td>
<td>70.4%</td>
<td>0.397</td>
<td>0.393</td>
<td>52</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>34</td>
<td>NeuralNet277</td>
<td>72.7%</td>
<td>0.455</td>
<td>0.399</td>
<td>28</td>
<td>50</td>
<td>6</td>
</tr>
<tr>
<td>37</td>
<td>NeuralNet297</td>
<td>72.4%</td>
<td>0.449</td>
<td>0.399</td>
<td>43</td>
<td>38</td>
<td>7</td>
</tr>
<tr>
<td>56</td>
<td>CV_CART2079</td>
<td>68.0%</td>
<td>0.391</td>
<td>0.401</td>
<td>54</td>
<td>4</td>
<td>8</td>
</tr>
<tr>
<td>54</td>
<td>CVNeuralNet2073</td>
<td>67.9%</td>
<td>0.403</td>
<td>0.401</td>
<td>55</td>
<td>6</td>
<td>9</td>
</tr>
<tr>
<td>59</td>
<td>CVNeuralNet2097</td>
<td>66.0%</td>
<td>0.403</td>
<td>0.401</td>
<td>59</td>
<td>7</td>
<td>10</td>
</tr>
<tr>
<td>61</td>
<td>CV_CART2104</td>
<td>70.4%</td>
<td>0.386</td>
<td>0.402</td>
<td>53</td>
<td>3</td>
<td>11</td>
</tr>
<tr>
<td>42</td>
<td>NeuralNet334</td>
<td>72.4%</td>
<td>0.450</td>
<td>0.404</td>
<td>40</td>
<td>44</td>
<td>12</td>
</tr>
<tr>
<td>52</td>
<td>CVLinReg2063</td>
<td>67.5%</td>
<td>0.404</td>
<td>0.404</td>
<td>57</td>
<td>8</td>
<td>13</td>
</tr>
<tr>
<td>41</td>
<td>NeuralNet330</td>
<td>72.4%</td>
<td>0.443</td>
<td>0.406</td>
<td>41</td>
<td>16</td>
<td>14</td>
</tr>
<tr>
<td>38</td>
<td>NeuralNet300</td>
<td>72.4%</td>
<td>0.451</td>
<td>0.408</td>
<td>38</td>
<td>45</td>
<td>15</td>
</tr>
<tr>
<td>55</td>
<td>CV_CHAID2078</td>
<td>64.6%</td>
<td>0.380</td>
<td>0.411</td>
<td>60</td>
<td>2</td>
<td>16</td>
</tr>
<tr>
<td>45</td>
<td>NeuralNet852</td>
<td>74.2%</td>
<td>0.456</td>
<td>0.413</td>
<td>3</td>
<td>51</td>
<td>17</td>
</tr>
<tr>
<td>53</td>
<td>CVLogit2068</td>
<td>67.5%</td>
<td>0.414</td>
<td>0.414</td>
<td>58</td>
<td>10</td>
<td>18</td>
</tr>
<tr>
<td>60</td>
<td>CV_CHAID2102</td>
<td>61.5%</td>
<td>0.380</td>
<td>0.414</td>
<td>61</td>
<td>1</td>
<td>19</td>
</tr>
<tr>
<td>58</td>
<td>CVLogit2092</td>
<td>67.7%</td>
<td>0.413</td>
<td>0.414</td>
<td>56</td>
<td>9</td>
<td>20</td>
</tr>
</tbody>
</table>

- Top RMS model is 9th in AUC, 2nd Test RMS rank is 42nd in AUC
- Correlation between rankings:

<table>
<thead>
<tr>
<th>AUC Rank</th>
<th>Train RMS Rank</th>
<th>Test RMS Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUC Rank</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Train RMS Rank</td>
<td>(0.465)</td>
<td>1</td>
</tr>
<tr>
<td>Test RMS Rank</td>
<td>(0.301)</td>
<td>0.267</td>
</tr>
</tbody>
</table>
KDDCup 98 Data: Top 200 Models Built Using Stepwise Variable Selection

- Error Metrics
  - Root-Mean-Squared (RMS) Error on Test data
  - Area Under the Curve (AUC) at the 70% Depth

Test RMS Error Ranking vs. AUC 70% Rank

\[
y = 0.1018x + 91.165
\]

\[
R^2 = 0.01037
\]

Models built using Unica Corp. Affinium Model
How Sampling Effects Accuracy Measures

• For example, 95% non-responders (N), 5% responders (R)

• What’s the Problem? (The justification for resampling)
  • “Sample is biased toward responders”
  • “Models will learn non-responders better”
  • “Most algorithms will generate models that say ‘call everything a non-responder’ and get 93% correct classification!” (I used to say this too)

• Most common solution:
  • Stratify the sample to get 50%/50% (some will argue that one only needs 20-30% responders)
What the Predictions Looks Like

Decision Tree

Neural Network

Quantiles

Summary Statistics

Mean
Std Dev
Std Err Mean
Upper 95% Mean
Lower 95% Mean
N

Mean
Std Dev
Std Err Mean
Upper 95% Mean
Lower 95% Mean
N

0.0715519
0.0747384
0.0010182
0.0735479
0.0695558
5388

0.0778526
0.0812571
0.001107
0.0800228
0.0756824
5388
Confusion Matrices For the Decision Tree: Before and After

**Decision Tree: Threshold at 0.5**

<table>
<thead>
<tr>
<th>Response_ STR</th>
<th>N</th>
<th>R</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>5,002</td>
<td>0</td>
<td>5,002</td>
</tr>
<tr>
<td>R</td>
<td>386</td>
<td>0</td>
<td>386</td>
</tr>
<tr>
<td>Total</td>
<td>5,388</td>
<td>0</td>
<td>5,388</td>
</tr>
</tbody>
</table>

**Decision Tree: Threshold at 0.071**

<table>
<thead>
<tr>
<th>Response_ STR</th>
<th>N</th>
<th>R</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>2,798</td>
<td>2,204</td>
<td>5,002</td>
</tr>
<tr>
<td>R</td>
<td>45</td>
<td>341</td>
<td>386</td>
</tr>
<tr>
<td>Total</td>
<td>2,843</td>
<td>2,545</td>
<td>5,388</td>
</tr>
</tbody>
</table>
To KNIME

KNIME Sampling
The Winner is...
Best Accuracy

http://www.netflixprize.com/leaderboard

<table>
<thead>
<tr>
<th>Rank</th>
<th>Team Name</th>
<th>Best Test Score</th>
<th>% Improvement</th>
<th>Best Submit Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>BellKor's Pragmatic Chaos</td>
<td>0.8567</td>
<td>10.06</td>
<td>2009-07-26 18:18:28</td>
</tr>
<tr>
<td>2</td>
<td>The Ensemble</td>
<td>0.8567</td>
<td>10.06</td>
<td>2009-07-26 18:38:22</td>
</tr>
<tr>
<td>3</td>
<td>Grand Prize Team</td>
<td>0.8582</td>
<td>9.90</td>
<td>2009-07-10 21:24:40</td>
</tr>
<tr>
<td>4</td>
<td>Opera Solutions and Vandalay United</td>
<td>0.8588</td>
<td>9.84</td>
<td>2009-07-10 01:12:31</td>
</tr>
<tr>
<td>5</td>
<td>Vandalay Industries !</td>
<td>0.8591</td>
<td>9.81</td>
<td>2009-07-10 00:32:20</td>
</tr>
<tr>
<td>6</td>
<td>PragmaticTheory</td>
<td>0.8594</td>
<td>9.77</td>
<td>2009-06-24 12:06:56</td>
</tr>
<tr>
<td>7</td>
<td>BellKor in BigChaos</td>
<td>0.8601</td>
<td>9.70</td>
<td>2009-05-13 08:14:09</td>
</tr>
<tr>
<td>8</td>
<td>Dace</td>
<td>0.8612</td>
<td>9.59</td>
<td>2009-07-24 17:18:43</td>
</tr>
<tr>
<td>9</td>
<td>Feeds2</td>
<td>0.8622</td>
<td>9.48</td>
<td>2009-07-12 13:11:51</td>
</tr>
<tr>
<td>10</td>
<td>BigChaos</td>
<td>0.8623</td>
<td>9.47</td>
<td>2009-04-07 12:33:59</td>
</tr>
</tbody>
</table>
Why Model Accuracy is Not Enough: Netflix Prize


A year into the competition, the Korbell team won the first Progress Prize with an 8.43% improvement. They reported more than 2000 hours of work in order to come up with the final combination of 107 algorithms that gave them this prize. And, they gave us the source code. We looked at the two underlying algorithms with the best performance in the ensemble: Matrix Factorization (which the community generally called SVD, Singular Value Decomposition) and Restricted Boltzmann Machines (RBM). SVD by itself provided a 0.8914 RMSE, while RBM alone provided a competitive but slightly worse 0.8990 RMSE. A linear blend of these two reduced the error to 0.88. To put these algorithms to use, we had to work to overcome some limitations, for instance that they were built to handle 100 million ratings, instead of the more than 5 billion that we have, and that they were not built to adapt as members added more ratings. But once we overcame those challenges, we put the two

If you followed the Prize competition, you might be wondering what happened with the final Grand Prize ensemble that won the $1M two years later. This is a truly impressive compilation and culmination of years of work, blending hundreds of predictive models to finally cross the finish line. We evaluated some of the new methods offline but the additional accuracy gains that we measured did not seem to justify the engineering effort needed to bring them into a production environment. Also, our focus on improving Netflix personalization had shifted to the next level by then. In the remainder of this post we will explain how and why it has shifted.
Why Data Science is Not Enough: Netflix Prize


Now it is clear that the Netflix Prize objective, accurate prediction of a movie's rating, is just one of the many components of an effective recommendation system that optimizes our members enjoyment. We also need to take into account factors such as context, title popularity, interest, evidence, novelty, diversity, and freshness. Supporting all the different contexts in which we want to make recommendations requires a range of algorithms that are tuned to the needs of those contexts. In the next part of this post, we will talk in more detail about the ranking problem. We will also dive into the data and models that make all the above possible and discuss our approach to innovating in this space.

There’s more to a solution than accuracy—you have to be able to use it!
Strategy 4
Do for algorithms what they can’t do for themselves

- Get the data right
- Understand how algorithms can be fooled with “correct” data
  - Outliers
  - Missing Values
  - Skew
  - High Cardinality
Clean Data: Outliers

- Are the outliers problems?
  - Some algorithms: “yes”
    - Linear regression, nearest neighbor, nearest mean, principal component analysis
    - In other words, algorithms that need mean values and standard deviations
  - Some algorithms: “no”
    - Decision trees, neural networks

- If outliers are problems for the algorithm
  - Are they key data points?
    - Do not remove these
    - Consider “taming” outliers with transformations (features)
  - Are they anomalies or otherwise uninteresting to the analysis
    - Remove from data so that they don’t bias models
To KNIME

KNIME outlier
Clean Data: Missing Values

• Missing data can appear as
  • blank, NULL, NA, or a code such as 0, 99, 999, or -1.

• Fixing Missing Data:
  • Delete the record (row), or delete the field (column)
  • Replace missing value with mean, median, or distribution
  • Replace with the missing value with an estimate
    • Select value from another field having high correlation with variable containing missing values
    • Build a model with variable containing missing values as output, and other variables without missing values as an input

• Other considerations
  • Create new binary variable (1/0) indicating missing values
  • Know what algorithms and software do by default with missing values
    • Some do listwise deletion, some recode with “0”, some recode with midpoints or means
## Missing Data:
### Imputation with Mean vs. Distribution

<table>
<thead>
<tr>
<th>payments</th>
<th>original data (no missing)</th>
<th>Cumulative with 10% missing</th>
<th>Cumulative with 10% missing, recoded</th>
<th>Cumulative with 30% missing</th>
<th>Cumulative with 30% missing, recoded</th>
</tr>
</thead>
<tbody>
<tr>
<td>500</td>
<td>500</td>
<td>500</td>
<td>500</td>
<td>500</td>
<td>500</td>
</tr>
<tr>
<td>2,019</td>
<td>2,519</td>
<td>2,519</td>
<td>2,519</td>
<td>2,519</td>
<td>2,519</td>
</tr>
<tr>
<td>7,529</td>
<td>10,047</td>
<td>10,047</td>
<td>10,047</td>
<td>10,047</td>
<td>10,047</td>
</tr>
<tr>
<td>7,954</td>
<td>18,001</td>
<td>18,001</td>
<td>18,001</td>
<td>18,001</td>
<td>18,001</td>
</tr>
<tr>
<td>8,438</td>
<td>26,439</td>
<td>26,439</td>
<td>26,439</td>
<td>26,439</td>
<td>80,737</td>
</tr>
<tr>
<td>9,471</td>
<td>44,827</td>
<td>44,827</td>
<td>170,000</td>
<td>80,737</td>
<td>170,000</td>
</tr>
<tr>
<td>9,912</td>
<td>54,739</td>
<td>54,739</td>
<td>54,739</td>
<td>54,739</td>
<td>54,739</td>
</tr>
<tr>
<td>10,373</td>
<td>65,112</td>
<td>65,112</td>
<td>65,112</td>
<td>65,112</td>
<td>65,112</td>
</tr>
<tr>
<td>10,930</td>
<td>76,042</td>
<td>76,042</td>
<td>76,042</td>
<td>76,042</td>
<td>76,042</td>
</tr>
<tr>
<td>11,392</td>
<td>87,434</td>
<td>87,434</td>
<td>87,434</td>
<td>87,434</td>
<td>87,434</td>
</tr>
<tr>
<td>11,855</td>
<td>99,289</td>
<td>99,289</td>
<td>99,289</td>
<td>99,289</td>
<td>80,737</td>
</tr>
<tr>
<td>12,357</td>
<td>111,646</td>
<td>111,646</td>
<td>111,646</td>
<td>111,646</td>
<td>160,000</td>
</tr>
<tr>
<td>12,862</td>
<td>124,509</td>
<td>124,509</td>
<td>124,509</td>
<td>124,509</td>
<td>124,509</td>
</tr>
<tr>
<td>13,340</td>
<td>137,848</td>
<td>137,848</td>
<td>137,848</td>
<td>137,848</td>
<td>22,222</td>
</tr>
<tr>
<td>13,866</td>
<td>151,704</td>
<td>151,704</td>
<td>151,704</td>
<td>151,704</td>
<td>151,704</td>
</tr>
<tr>
<td>14,362</td>
<td>165,956</td>
<td>165,956</td>
<td>165,956</td>
<td>165,956</td>
<td>37,000</td>
</tr>
<tr>
<td>14,813</td>
<td>180,769</td>
<td>180,769</td>
<td>180,769</td>
<td>180,769</td>
<td>180,769</td>
</tr>
<tr>
<td>15,351</td>
<td>196,120</td>
<td>196,120</td>
<td>196,120</td>
<td>196,120</td>
<td>196,120</td>
</tr>
<tr>
<td>15,817</td>
<td>211,937</td>
<td>211,937</td>
<td>211,937</td>
<td>211,937</td>
<td>211,937</td>
</tr>
</tbody>
</table>

Mean: 90,040  Std. Dev.: 67,415  Median: 81,738  Min: 500  Max: 211,937

<table>
<thead>
<tr>
<th></th>
<th>MAINTAIN</th>
<th>MAINTAIN</th>
<th>MAINTAIN</th>
<th>MAINTAIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>about mean</td>
<td>about mean</td>
<td>about mean</td>
<td>about mean</td>
</tr>
<tr>
<td>StDev</td>
<td>about mean</td>
<td>about mean</td>
<td>about mean</td>
<td>about mean</td>
</tr>
</tbody>
</table>

© Abbott Analytics, Inc. 2001-2013
Clean Data: Missing Data

- How much can missing data effect models?

- Example at upper right has 5300+ records, 17 missing values encoded as “0”

- After fixing model with mean imputation, $R^2$ rises from 0.597 to 0.657

- Why? Missing was recoded with “0” in this example, which was a particularly bad imputation for this data
# Transforms: Changing Distribution of Data

## Positive Skew
- Tail of distribution to right
- Correction: log transform
- Example: MAX_DON_AMT

## Negative Skew
- Tail of distribution to left
- Correction: Power $\geq 2$, Exp
- Example: HOMEVAL50

<table>
<thead>
<tr>
<th>Field</th>
<th>Selected Transform</th>
<th>Current Distribution</th>
<th>Inverse</th>
<th>LogN</th>
<th>Log10</th>
<th>Exponential</th>
<th>Square Root</th>
</tr>
</thead>
<tbody>
<tr>
<td>AGE</td>
<td>Current Distribution</td>
<td>61.506 (14.419)</td>
<td>0.017 (0.005)</td>
<td>4.089 (0.253)</td>
<td>1.776 (0.110)</td>
<td>3256315221215830...</td>
<td>7.786 (0.944)</td>
</tr>
<tr>
<td>HOMEVAL50</td>
<td>Current Distribution</td>
<td>75.193 (29.452)</td>
<td>0.023 (0.059)</td>
<td>4.191 (0.667)</td>
<td>1.916 (0.205)</td>
<td>3023021000000011...</td>
<td>0.311 (2.238)</td>
</tr>
<tr>
<td>NUMGIFT_LIFE</td>
<td>Current Distribution</td>
<td>8.402 (8.513)</td>
<td>0.263 (0.289)</td>
<td>1.829 (0.975)</td>
<td>0.794 (0.423)</td>
<td>8340222604463460...</td>
<td>2.799 (1.274)</td>
</tr>
<tr>
<td>REC_DON_A...</td>
<td>Current Distribution</td>
<td>17.897 (12.461)</td>
<td>0.077 (0.054)</td>
<td>2.733 (0.556)</td>
<td>1.187 (0.241)</td>
<td>3805593244323034...</td>
<td>4.062 (1.180)</td>
</tr>
<tr>
<td>AVG_DON</td>
<td>Current Distribution</td>
<td>13.744 (8.630)</td>
<td>0.095 (0.053)</td>
<td>2.484 (0.515)</td>
<td>1.079 (0.224)</td>
<td>9655746729556163...</td>
<td>3.560 (0.964)</td>
</tr>
</tbody>
</table>

Cells contain: Mean (Standard Deviation)
Why Skew Matters (In Regression Modeling)

• Obscures information in plot
  • Spaced in scatterplot taken up by empty space in upper (or lower) end of skewed values

• Regression models fit worse with skewed data
  • In example at right, by simply applying the log transform, performance is improved from $R^2=0.566$ to 0.597
To KNIME

KNIME outlier
Transforms: Scaling Data

- Before normalization, income scale “dwarfs” age
- z-score
  - \( x^* = \frac{ x - \text{mean} }{ \text{std} } \)
  - Income and age on same scale
- Scale to range \([0,1]\)
  - \( x^* = \frac{ x - x_{\text{min}} }{ x_{\text{max}} - x_{\text{min}} } \)
- Both allow one to see both variables on same scale
- Can apply this to subsamples of data (regional data, for example)
### Grouping and Exploding Categorical Data

#### Group: State to Region

<table>
<thead>
<tr>
<th>State</th>
<th>Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>AK</td>
<td>Northwest</td>
</tr>
<tr>
<td>AL</td>
<td>Southeast</td>
</tr>
<tr>
<td>AR</td>
<td>Southeast</td>
</tr>
<tr>
<td>CA</td>
<td>Southwest</td>
</tr>
<tr>
<td>CO</td>
<td>Mountain</td>
</tr>
<tr>
<td>CT</td>
<td>Northeast</td>
</tr>
<tr>
<td>DC</td>
<td>Mid-Atlantic</td>
</tr>
<tr>
<td>DE</td>
<td>Mid-Atlantic</td>
</tr>
<tr>
<td>FL</td>
<td>Southeast</td>
</tr>
<tr>
<td>GA</td>
<td>Southeast</td>
</tr>
<tr>
<td>HI</td>
<td>Southwest</td>
</tr>
<tr>
<td>IA</td>
<td>Midwest</td>
</tr>
<tr>
<td>ID</td>
<td>Northwest</td>
</tr>
<tr>
<td>IL</td>
<td>Midwest</td>
</tr>
<tr>
<td>IN</td>
<td>Midwest</td>
</tr>
</tbody>
</table>

#### Explode: Region “dummy” variables

<table>
<thead>
<tr>
<th>Region</th>
<th>Mid-Atlantic</th>
<th>Midwest</th>
<th>Mountain</th>
<th>Northeast</th>
<th>Northwest</th>
<th>Southeast</th>
</tr>
</thead>
<tbody>
<tr>
<td>Northwest</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>Southeast</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Mountain</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>North</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Mid-Atlantic</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Southeast</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Southeast</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Southwest</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Southwest</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Midwest</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Northwest</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Midwest</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

- Categorical data having values with small populations (10s of cases) is very problematic in modeling. They should be binned up (grouped) as much as is possible!
Effect of High Cardinality

- **Cardinality**: number of levels in a variable  
  - We care about cardinality in categorical variables  
  - # levels -> frequency counts

- **Why do we care?**  
  - Decision trees are biased toward accepting splits with variables having high cardinality  
  - Numeric algorithm implementations that automatically create dummy variables for categoricals may create *lots* of new 1/0 dummies  
    - Higher # inputs in models  
    - Lots of low information content variables
Effect of High Cardinality

<table>
<thead>
<tr>
<th>Variable</th>
<th># populated</th>
<th># missing</th>
<th>% missing</th>
<th># levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFA_2$</td>
<td>95412</td>
<td>0</td>
<td>0</td>
<td>14</td>
</tr>
<tr>
<td>RFA_3$</td>
<td>93462</td>
<td>1950</td>
<td>2.04</td>
<td>70</td>
</tr>
</tbody>
</table>
KNIME Decision Trees and High Cardinality
To KNIME

KNIME cardinality
Strategy 5: Deploy Models Wisely

• Get the data right
• Understand how algorithms can be fooled with “correct” data
  • Outliers
  • Missing Values
  • Skew
  • High Cardinality
What is Deployment?

• Using the model for its intended purpose
  • Reporting
  • Transactional scoring
  • Batch scoring

• Most models relate to decisions to be made within the organization
## Different Approaches to Deployment

<table>
<thead>
<tr>
<th>Data Prep</th>
<th>Model</th>
<th>Type of Application</th>
</tr>
</thead>
<tbody>
<tr>
<td>In Database</td>
<td>In Database</td>
<td>real-time scoring</td>
</tr>
<tr>
<td></td>
<td>In PA Software</td>
<td>weekly/monthly scoring</td>
</tr>
<tr>
<td></td>
<td>Standalone</td>
<td>real-time scoring</td>
</tr>
<tr>
<td></td>
<td>In Cloud</td>
<td>large, real-time scoring</td>
</tr>
<tr>
<td>in PA Software</td>
<td>In Database</td>
<td>large, real-time scoring</td>
</tr>
<tr>
<td></td>
<td>In PA Software</td>
<td>ad hoc scoring</td>
</tr>
<tr>
<td></td>
<td>Standalone</td>
<td>complex prep; occasional scoring</td>
</tr>
<tr>
<td></td>
<td>In Cloud</td>
<td>big data; complex prep; occasional scoring</td>
</tr>
<tr>
<td>In Cloud</td>
<td>In Database</td>
<td>big data; complex prep; occasional scoring</td>
</tr>
<tr>
<td></td>
<td>In PA Software</td>
<td>unlikely</td>
</tr>
<tr>
<td></td>
<td>Standalone</td>
<td>big data; complex prep; occasional scoring</td>
</tr>
<tr>
<td></td>
<td>In Cloud</td>
<td>large, real-time scoring</td>
</tr>
</tbody>
</table>

© Abbott Analytics, Inc. 2001-2013
Form of Models for Deployment: In-PA Software Deployment

- Run models through original software in ad hoc or automated process

- Benefits:
  - Data prep done in software still there
    - But still may have to trim down processing for efficiency
  - no further work to be done to deploy

- Drawbacks
  - Usually slower
    - have to pull data out and push it back to database
    - Software not usually optimized for speed; optimized for usability
  - Requires a software expert to maintain and troubleshoot
    - Analyst usually involved
    - Errors not always handled gracefully
Form of Models for Deployment: External Call to PA Software

- Run models through original software in ad hoc or automated process, but as a call from the OS

- Benefits:
  - Data prep done in software still there
  - But still may have to trim down processing for efficiency
  - no further work to be done to deploy

- Drawbacks
  - Usually slower
  - have to pull data out and push it back to database
  - Software not usually optimized for speed; optimized for usability
  - Requires a software expert to maintain and troubleshoot
  - Analyst usually involved
  - Errors not always handled gracefully
Form of Models for Deployment: Translation to Another Language

- Translate models into SQL, C (++, #, etc.), Java, PMML
  - If in C/Java, can create standalone application just for the model scoring

- Benefits
  - Get models out of software environment where they can be run and maintained by others
  - Often run more efficiently in database or other environment
  - Many tools provide export capabilities into other languages

- Drawbacks
  - Translation of dataprep not usually included in tool export, requires significant time and QC/QA to ensure consistency with the tool
  - Bug fixes take longer
Form of Models for Deployment: PMML

- Translate models into PMML
  - Different than SQL, C, Java, etc.

- Benefits
  - PMML supports (natively) entire predictive modeling process
  - Language is simple
  - Database support
  - Online support for scalable scoring (Zementis)

- Drawbacks
  - Translation of dataprep not usually included in predictive modeling software tools, requires coding
  - Models are verbose
  - Open source scoring options are limited
Typical Model Deployment Processing Flow

Data to Score → Import/Select Data to Score → Select Fields Needed → Clean Data (missing, recodes, …)

Scored Data → Decile** Scored Data → Score* Data → Re-create Derived Variables

The key: reproduce all data pre-processing done to build the models

* Score
** Decile
Resources

Dean Abbott
Abbott Analytics, Inc.
Predictive Analytics World, Berlin (#pawcon)
November 6, 2013

Email: dean@abbottanalytics.com
Blog: http://abbottanalytics.blogspot.com
Twitter: @deanabb
Predictive Analytics Overview

From Amazon.com

- Hardcover: 320 pages
- Publisher: Wiley; 1 edition (February 18, 2013)
- Language: English
- ISBN-10: 1118356853

Great introduction to Predictive Analytics
Accessible Books on Statistics

• The Cartoon Guide to Statistics by Larry Gonick, Woollcott Smith
  (Contributor), Woollcott Smith
  Paperback - 240 pages (February 25, 1994)
  HarperCollins (paper); ISBN: 0062731025

How to Lie With Statistics
  by Darrell Huff, Irving Geis (Illustrator)
  Paperback Reissue edition (November 1993)
From Amazon.com

- Data Preparation for Data Mining
- by Dorian Pyle
- Paperback - 540 pages Bk&Cd Rom edition (March 15, 1999)
- Morgan Kaufmann Publishers;
- ISBN: 1558605290 ;

Excellent resource for the part of data mining that takes the most time. Best book on the market for data preparation.
Data Mining Methods

From Amazon.com

- Handbook of Statistical Analysis and Data Mining Applications by Robert Nisbet, John Elder, Gary Miner
- Hardcover: 900 pages
- Publisher: Academic Press (April 23, 2009)
- Language: English
- ISBN-10: 0123747651

New data mining book written for practitioners, with case studies and specifics of how problems were worked in Enterprise Miner, Clementine, STATISTICA, or another tool.
Learn the art and science of predictive analytics — techniques that get results

Predictive analytics is what translates big data into meaningful, usable business information. Written by a leading expert in the field, this guide examines the science of the underlying algorithms as well as the principles and best practices that govern the art of predictive analytics. It clearly explains the theory behind predictive analytics, teaches the methods, principles, and techniques for conducting predictive analytics projects, and offers tips and tricks that are essential for successful predictive modeling. Hands-on examples and case studies are included.
Go beyond mere insight and build models than you can deploy in the day to day running of your business
Save time and effort while getting more value from your data than ever before
Loaded with detailed step-by-step examples that show you exactly how it’s done by the best in the business
Data Mining Algorithms

From Amazon.com

- Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations
- By Eibe Frank, Ian H. Witten
- Paperback - 416 pages (October 13, 1999)
- Morgan Kaufmann Publishers;
- ISBN: 1558605525;

Best book I’ve found in between highly technical and introductory books. Good coverage of topics, especially trees and rules, but no neural networks.
From Amazon.com

- Neural Networks for Pattern Recognition by Christopher M. Bishop
- Paperback (November 1995)
- Oxford Univ Press;
- ISBN: 0198538642

Excellent book for neural network algorithms, including some lesser known varieties.

Described as “Best of the best” by Warren Sarle (Neural Network FAQ)
Ripley is a statistician who has embraced data mining. This book is not just about neural networks, but covers all the major data mining algorithms in a very technical and complete manner.

Sarle calls this the best advanced book on Neural Networks.
Accessible Technical Description of Algorithms

Machine Learning, Neural and Statistical Classification
D. Michie, D.J. Spiegelhalter, C.C. Taylor (eds)
Available free online (PDF)
http://www.amsta.leeds.ac.uk/~charles/statlog/

From Amazon.com
Classification and Regression Trees by Leo Breiman
Paperback (December 1983)
CRC Press;
ISBN: 0412048418
The definitive textbook on decision trees from the inventors of the CART algorithm.
Popular Data Mining Books
Biographies

With contributions from:

Dean Abbott, @deanabb
Charu Aggarwal
Michael Berthold
Chris Clifton
John Elder IV
David J. Hand
Cheryl G. Howard
J. Dustin Hux
Hillol Kargupta
Colleen McLaughlin McCue
G. J. McLachlan
Gregory Piatetsky-Shapiro, @kdnuggets
Shusaku Tsumoto
Graham J. Williams
Mohammed J. Zaki

Publication Date: July 21, 2012
ISBN-10: 3642280463
Descriptions of Algorithms

- Neural Network FAQ

- Statistical data mining tutorials by Andrew Moore, Carnegie Mellon
  - http://www-2.cs.cmu.edu/~awm/tutorials/

- A list of papers and abstracts from The University of Bonn
  Data Clustering and Visualization is a category of particular interest. Hasn’t been updated since 2003, but still a good selection of papers.
  - http://www-dbv.informatik.uni-bonn.de

- A Statistical Learning/ Pattern Recognition Glossary by Thomas Minka.
  Very comprehensive list of data mining terms and glossary-like descriptions
  - http://www.stat.cmu.edu/~minka/statlearn/glossary/