Data Mining with Regression

Teaching an old dog some new tricks

Bob Stine
Department of Statistics
The Wharton School of the Univ of Pennsylvania
March 31, 2006

Overview

- Familiar regression model, but...
- Adapt to the context of data mining
  - Scope: Borrow from machine learning
  - Search: Heuristic sequential strategies
  - Selection: Alpha-investing rules
  - Estimation: Adaptive "empirical Bayes"
  - Structure: Calibration
- Does it work?
  - Numerical comparisons to other methods using reference data sets

Data Mining Context

- Predictive modeling of wide data
- Modern data sets
  - n ... Thousands to millions of rows
  - m ... Hundreds to thousands of columns.
- No matter how large n becomes, can conceive of models with m > n
  - Derived features (e.g., interactions)
- Consequence
  - Cannot fit "saturated" model to estimate \( \sigma^2 \)
  - Cannot assume true model in fitted class

Acknowledgments

- Colleagues
  - Dean Foster in Statistics
  - Lyle Ungar in Computer Science
Wide Data

<table>
<thead>
<tr>
<th>Application</th>
<th>Rows</th>
<th>Columns</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit</td>
<td>3,000,000</td>
<td>350</td>
</tr>
<tr>
<td>Faces</td>
<td>10,000</td>
<td>1,400</td>
</tr>
<tr>
<td>Genetics</td>
<td>1,000</td>
<td>10,000</td>
</tr>
<tr>
<td>CiteSeer</td>
<td>500</td>
<td>∞</td>
</tr>
</tbody>
</table>

Lots of Data

- Credit scoring
  - Millions of credit card users
  - Past use of credit, economics, transactions
- Text
  - Documents to be classified into categories
  - Large corpus of marked documents, and even more that have not been marked
- Images
  - Millions of images from video surveillance
  - All those pixel patterns become features

Experience

- Model for bankruptcy
  - Stepwise regression selecting from more than 67,000 predictors
- Successful
  - Better classifications than C4.5
- But
  - Fit dominated by interactions
    - Linear terms hidden
  - Know missed some things, even with 67,000
    - Unable to exploit domain knowledge
  - Not the fastest code to run

Why use regression?

- Familiarity
  - Reduce the chances for pilot error
- Well-defined classical inference
  - IF you know the predictors, inference easy
- Linear approximation good enough
  - Even if the “right answer” is nonlinear
- Good diagnostics
  - Residual analysis helpful, even with millions
- Framework for studying other methods
Key Challenge

- Which features to use in a model?
- Cannot use them all!
  - Too many
  - Over-fitting
- May need transformations
  - Even if did use them all, may not find best
- Model averaging?
  - Too slow
  - Save for later... along with bagging.

Extending Regression

- Scope of feature space
  - Reproducing kernel Hilbert space (from SVMs)
- Search and selection methods
  - Auction
  - Estimation
  - Adaptive shrinkage improves testimator
- Structure of model
  - Calibration

Extending Regression

- Scope of feature space
  - Reproducing kernel Hilbert space

Larger Scope

- Lesson from analysis of bankruptcy
  - Interactions can be very useful
  - But dominate if all predictors treated as monolithic group (m linear, m^2 second order)
- Question
  - How to incorporate useful quadratic interactions, other transformations?
  - Particularly hard to answer in "genetic situations" with every wide data sets for which m >> n.
Reproducing Kernels

- Some history
  - Introduced in Stat by Parzen and Wahba
  - Adopted by machine learning community for use in support vector machines.

- Use in regression
  - Find "interesting" directions in feature space
  - Avoid explicit calculation of the points in the very high dimensional feature space.

Example of RKHS

- Bulls-eye pattern
- Non-linear boundary between cases in the two groups

Example of RKHS

- Linearize boundary
  - Add $X_1^2$ and $X_2^2$ to basis
  - Does not generalize easily (too many)

- Alternative using RKHS
  - Define new feature space $X \rightarrow \varphi(X)$
    - Possibly much higher dimension than m
  - Inner product between points $x_1$ and $x_2$ in new space is $\langle \varphi(x_1), \varphi(x_2) \rangle$
  - Reproducing kernel $K$ evaluates inner product without forming $\varphi(x)$ explicitly
    - $K(x_1, x_2) = \langle \varphi(x_1), \varphi(x_2) \rangle$

Example of RKHS

- Industry inventing kernel functions
- Gaussian kernel (aka, radial basis)
  - $K(x_1, x_2) = c \exp(-\|x_1 - x_2\|^2)$
- Generate several new features
  - Compute Gram matrix in feature space $\varphi$ indirectly using kernel $K$
    - $G = [K(x_i, x_j)]_{n \times n}$
  - Find leading singular vectors of $G$, as in a principal component analysis
  - These become directions in the model
**Example of RKHS**

- For the bulls-eye, leading two singular vectors convert circle to hyperplane

![Original vs Gaussian kernel](image)

**Extending Regression**

- Scope of feature space
  - Expand with components from RKHS
- Search and selection methods
  - Experts recommend features to auction

**Auction-Based Search**

- Lesson from analysis of bankruptcy
  - Interactions help, but all interactions?
  - Must we consider every interaction, or just those among predictors in the model?
- Further motivation
  - Substantive experts reveal missing features.
  - In some applications, the scope of the search depends on the state of the model
    - Examples: citations in CiteSeer, genetics
  - Streaming features

**Feature Auction**

- “Expert”
  - Strategy that recommends a candidate feature to add to the model
- Examples
  - PCA of original data
  - RKHS using various kernels
  - Interactions
  - Parasitic experts
  - Substantive transformations
- Experts bid for opportunity to recommend a feature (or bundle)
Feature Auction
-
- Expert is rewarded if correct
- Experts have “wealth”
- If recommended feature is accepted in the model, expert earns $\omega$ additional wealth
- If recommended feature is refused, expert loses bid

As auction proceeds, it...
-
- Rewards experts that offer useful features, allowing these to recommend more X’s
- Eliminates experts whose features are not accepted.

Alpha-Investing

- Wealth = Type I error
- Each expert begins auction with nominal level to spend, say $W_0 = 0.05$
- At step $j$ of the auction,
  - Expert bids $0 \leq \alpha_j \leq W_{j-1}$ to recommend $X$
  - Assume this is the largest bid
  - Model assigns p-value $p$ to $X$
  - If $p < \alpha$: add $X$; set $W_j = W_{j-1} + (\omega-p)$
  - If $p > \alpha$: don’t add $X$; set $W_j = W_{j-1} - \alpha_j$

Discussion of Alpha-Investing

- Similar to alpha-spending rules that are used in clinical trials
- But allows good experts to continue suggesting features
- Infinitely many tests
- Can imitate various tests of multiple null hypotheses
  - Bonferroni
  - Step-down testing
Discussion of Alpha-Investing

- Can test an infinite sequence of hypotheses
- Step-down testing allows only finite collection: must begin with ordered p-values
- Alpha investing is sequential
- If expert has “good science”, then bids heavily on the hypotheses assumed to be most useful

\[ \alpha_j \propto \frac{W_0}{j^2} \]

Over-fitting?

If expert receives \( \alpha \) back in the feature auction, then what’s to stop model from over-fitting?

Excess Discovery Count

- Number of correct rejections in excess of, say, 95% of total rejections
- Terminology
  \( S_\theta(m) = \# \) correct rejections in \( m \) tests
  \( R(m) = \# \) rejections in \( m \) tests
- Excess discovery count
  \[ EDC_{\alpha,\gamma}(m) = \alpha + E\theta(S_\theta(m) - \gamma R(m)) \]
- Procedure
  “controls EDC” \( \iff \) \( EDC_{\alpha,\gamma}(m) \geq 0 \)

Excess Discovery Count

\[ EDC_{\alpha,\gamma}(m) = \alpha + E\theta(S_\theta(m) - \gamma R(m)) \]
Alpha-Investing Controls EDC

Theorem: An alpha-investing rule with initial wealth $W_0 \leq \alpha$ and payoff $\omega \leq (1-\gamma)$ controls EDC.

For sequence of “honest” tests of the sequence $H_0(1), \ldots, H_0(m), \ldots$ and any stopping time $M$

$$\inf_{M} \inf_{\theta} E_{\theta} EDC_{\alpha,\gamma}(M) \geq 0$$

Comparison to FDR

Notation

- $R(m) = \#$ rejected $= S_{\theta}(m) + V_{\theta}(m)$
- $V_{\theta}(m) = \#$ false rejections (Type I errors)

False discovery rate controls ratio of false positives to rejections

$$E_{\theta} \left( \frac{V_{\theta}(m)}{R(m)} \mid R(m) > 0 \right) P(R(m) > 0) \leq FDR$$

Control of EDC implies that

$$\frac{E_{\theta} V_{\theta}(m)}{E_{\theta} R(m)} \leq (1-\gamma) + \frac{\alpha}{E_{\theta} R(m)}$$

Extending Regression

- Scope of feature space
  - Expand with components from RKHS
- Search and selection methods
  - Experts recommend features to auction
- Estimation
  - Adaptive shrinkage improves testimator

Testimators

Estimate mean $\mu$ in multivariate normal $Y \sim N(\mu, \sigma^2 I)$

- Hard thresholding (D&J, 1994, wavelet)
- Possesses certain minimax optimality
- Bounds risk relative to an oracle that knows which variables to include in the model
- Basically same as using a Bonferroni rule applied to p-values
Adaptive Estimator

- "Polyshrink" adaptively shrinks estimator when fitting in higher dimensions
- About the same as a testimator when fitting one estimator
- In higher dimensions, shrinkage varies with the level of signal found
- Possesses type of optimality in the sense of a robust prior.
- Resembles empirical Bayes estimators (e.g., Silverman & Johnstone)

Value in Modeling

- Evaluate one predictor at a time
  - No real gain over testimator
- Evaluate several predictors at once
  - Shrinkage has some teeth
- Several predictors at once?
  - Generally do one at a time, eschew "Principle of Marginality"
  - Bundles originate in RKHS: take top k components from feature space

Extending Regression

- Scope of feature space
  - Expand with components from RKHS
- Search and selection methods
  - Experts recommend features to auction
- Estimation
  - Adaptive shrinkage improves testimator
- Structure of model
  - Estimate empirical link function

Calibration

- Model is calibrated if predictions are correct on average
  \[ E(Y|\hat{Y}) = \hat{Y} \]
- Link function in generalized linear model has similar role
  \[ E(y) = g(x'\beta) \]
- Rather than assume a known link, estimate the link as part of the modeling
Empirical Calibration

Before

After

Extending Regression

Scope of feature space
- Expand with components from RKHS
- Search and selection methods
- Experts recommend features to auction

Estimation
- Adaptive shrinkage improves testimator

Structure of model
- Estimate empirical link function

Challenges

Problems
- Control proliferation of interactions
- Incorporate expert guidance
- Explore richer spaces of predictors
- Run faster

Computing
- Streaming selection is much faster than batch
- Have run 1,000,000+ features in applications

Comparisons

NIPS data sets
- Competition among 100+ algorithms
- Goal to predict cases in a hold back sample
- Success based on area under ROC

Data sets
- Variety of contexts
- More wide than tall
Results: 2003 NIPS

Unlike BR: Very high signal rates...

<table>
<thead>
<tr>
<th>Dataset</th>
<th>n</th>
<th>m</th>
<th>AUC</th>
<th>NIPS*</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arcene</td>
<td>200</td>
<td>10,000</td>
<td>0.93</td>
<td>0.96</td>
</tr>
<tr>
<td>Dexter</td>
<td>600</td>
<td>20,000</td>
<td>0.99+</td>
<td>0.992</td>
</tr>
<tr>
<td>Dorothea</td>
<td>1150</td>
<td>100,000</td>
<td>?</td>
<td></td>
</tr>
<tr>
<td>Gisette</td>
<td>7000</td>
<td>5,000</td>
<td>0.995</td>
<td>0.999</td>
</tr>
<tr>
<td>Madelon</td>
<td>2600</td>
<td>500</td>
<td>0.94</td>
<td>0.95</td>
</tr>
</tbody>
</table>

Results: Face Detection

- 10,000 images,
- 5,000 with faces and 5,000 without
- Type I error at 50% Type II

<table>
<thead>
<tr>
<th>Method</th>
<th>Type I</th>
</tr>
</thead>
<tbody>
<tr>
<td>AdaBoost</td>
<td>0.07</td>
</tr>
<tr>
<td>FFS</td>
<td>0.07</td>
</tr>
<tr>
<td>AsymBoost</td>
<td>0.07</td>
</tr>
<tr>
<td>Streaming</td>
<td>0.05</td>
</tr>
</tbody>
</table>

What Next?

- More examples
- Working on faster version of software
- Data formats are a big issue
- Implement subspace shrinkage
  - Current implementation uses hard thresholding
- Improve expert strategy
  - Goal of machine learning is turn-key system
  - Prefer ability to build in expertise