Part 1: Data mining and its take on overfitting

- What is data mining?
- An example: nearest neighbors
- Review: ordinary least squares
- Overfitting: Adjusted R-squared
- A data mining answer to overfitting: cross validation
- An alternative to linear regression: MARS
- MARS: how it works
- MARS: overfitting
- MARS advantages: continuous (actuaries like) handles high dimensions, chooses inflection points
- The R "earth" package
- examples

Part 2: Success with large data sets

- The recommendations problem (aka collaborative filtering)
- Large data sets
- The simplest possible recommender
- Netflix prize
- Using correlation+nearest neighbors as smoothing function
- Engineering concerns (scaling, fault recovery, availability)
- Even better than SQL: Pig!
- What is Map Reduce?
- Final thought on cross-validation
- Q and A

Part 1

Data Mining and its take on overfitting

What is Data Mining?

Data mining, also popularly referred to as knowledge discovery from data (KDD), is the automated or convenient extraction of patterns representing knowledge implicitly stored or captured in large data bases, data warehouses, the Web, other massive information repositories, or data streams.

Han and Kamber

Data mining is the process of extracting previously unknown comprehensible and actionable information from large databases and using it to make crucial business decisions

Zekulin

Data Mining is Decision Trees; Neural Networks; Rule Induction; Nearest Neighbors; Genetic Algorithms.

Mehta

What is Data Mining?

Friedman, J. H. "Data Mining and Statistics: What's the Connection?" (Nov. 1997b).

“This paper addresses the following issues:

What is Data Mining?
What is Statistics?
What is the connection, if any?
How can statisticians contribute if at all?
Should we want to?”

http://www-stat.stanford.edu/~jhf/

What is data mining?

A few of the problem areas

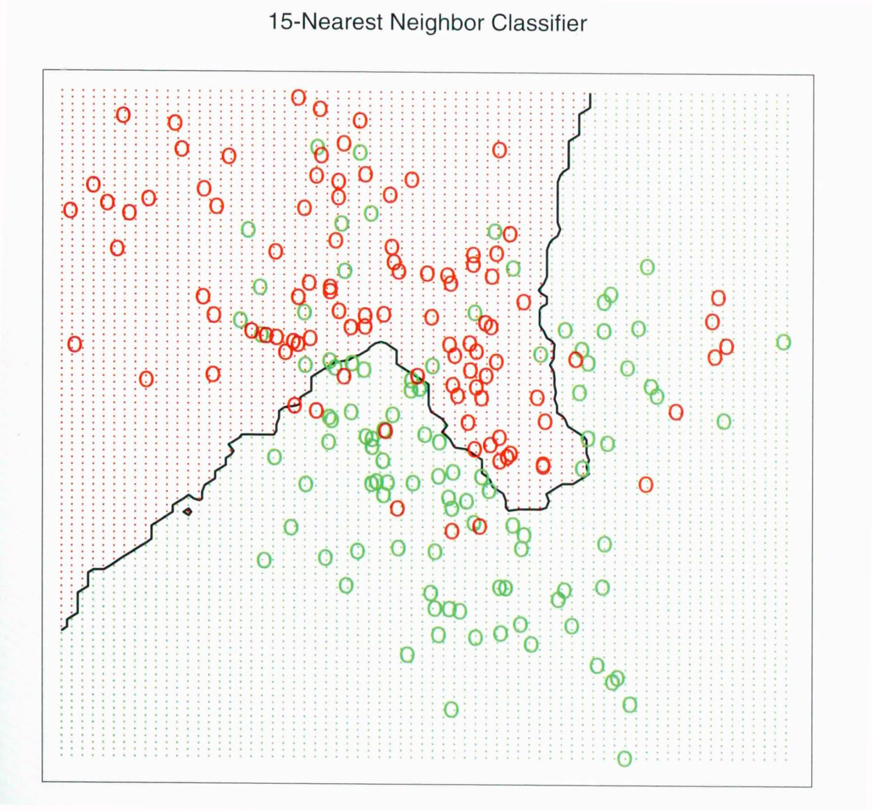
- Supervised learning (aka prediction)
  - Given (vector) pairs (x_i, y_i), describe $y = f(x)$
- Unsupervised learning
  - Given vector elements x_i, partition x_i into meaningful subsets
- Active learning
  - Given the ability to guide x_i through a space and make observations $y_i = f(x_i)$, describe $y = f(x)$.
- Reinforcement learning
  - Active learning, but in an environment which may be changing with time, possibly in response to observations

An example: nearest neighbors

$\widehat{f}(x) = Average(y_i | x_i \in N_k(x))$

where

$N_k$is the largest ball containing k of the x_i.



This is not pen-and-paper statistics!

Review: ordinary least squares

$$f(X)=\beta_0+\sum_{j=1}^{p}\beta_j X_j$$

Optimization: find β that minimizes RSS (aka R^2).

$$RSS(\beta)=\sum_{i=1}^{N}(y_i-f(x_i))^2$$

$$RSS(\beta)=(y-XB)^T(y-XB)$$

$\widehat{\beta}=(X^TX)^{-1}(X^Ty)$ - the “Normal Equations”

[HTF p43-44]

Overfitting: Adjusted R-squared $\overline{R}^2$

$\overline{R}^2 = 1 - \frac{n-1}{n-p}(1-R^2)$

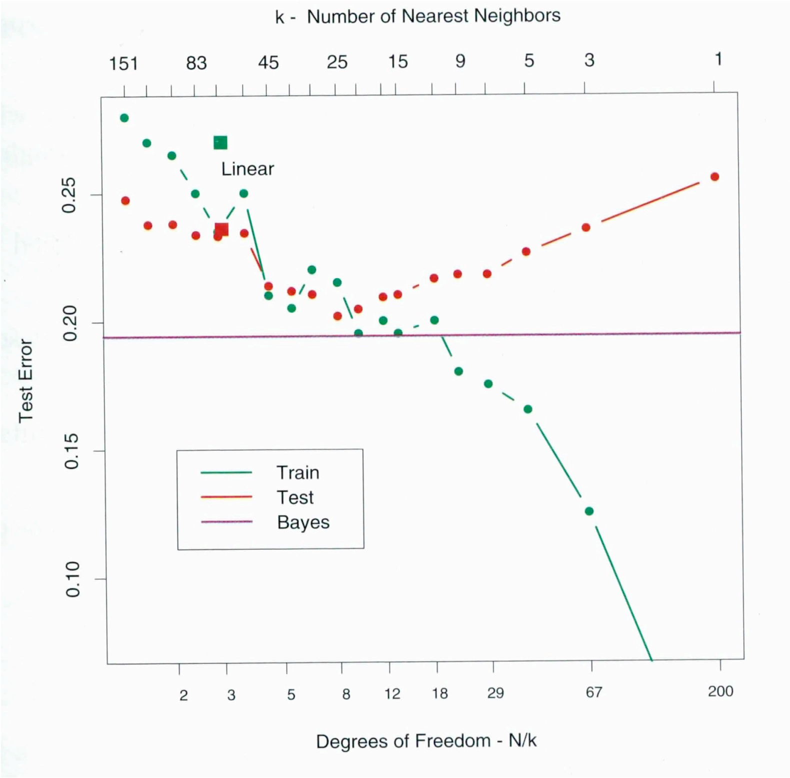
- designed to penalize for the excess number of terms

- always smaller thanR^2

http://en.wikipedia.org/wiki/Ordinary_least_squares

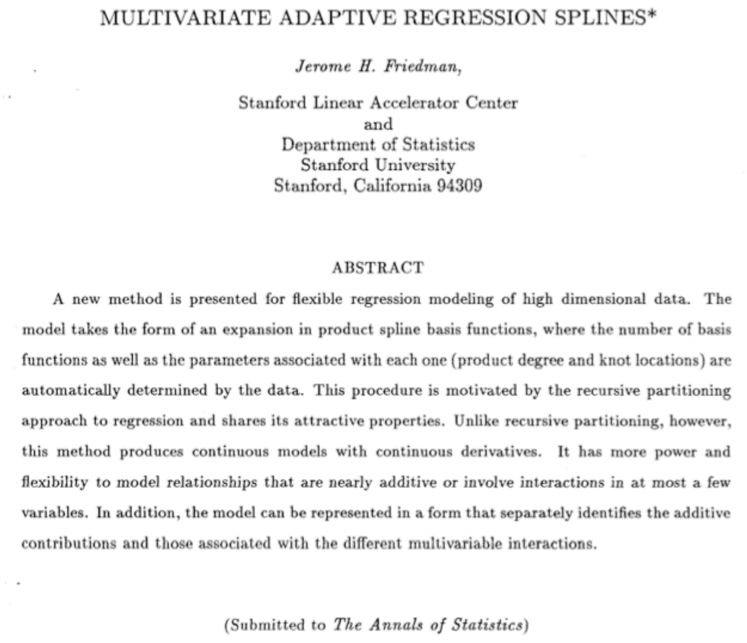
A data mining answer to overfitting: cross validation

- Divide data randomly in half. Model on first half, measure on the second.
- Can compare models of unrelated frameworks so long as they have the same loss function.
- Pseudonyms: "train/test", "random holdback". Common variation called "n-fold cross validation".



An alternative to linear regression: MARS

(1991)



MULTIVARIATE ADAPTIVE REGRESSION SPLINES*

Jerome H. Friedman,

Stanford Linear Accelerator Center
and
Department of Statistics
Stanford University
Stanford, California 94309

ABSTRACT

A new method is presented for flexible regression modeling of high dimensional data. The model takes the form of an expansion in product spline basis functions, where the number of basis functions as well as the parameters associated with each one (product degree and knot locations) are automatically determined by the data. This procedure is motivated by the recursive partitioning approach to regression and shares its attractive properties. Unlike recursive partitioning, however, this method produces continuous models with continuous derivatives. It has more power and flexibility to model relationships that are nearly additive or involve interactions in at most a few variables. In addition, the model can be represented in a form that separately identifies the additive contributions and those associated with the different multivariable interactions.

(Submitted to *The Annals of Statistics*)

MARS: history

- 1983: Founding of Salford Systems

- 1991: Friedman invents MARS. Published in *Annals of Statistics*. Friedman's Stanford faculty web page distributes free code of MARS algorithm.

- 1993: Friedman publishes applications and performance enhancements

(Dates unknown:)

- Friedman sells MARS to Salford Systems

- Salford Systems trademarks MARS

- Friedman removes free MARS code from faculty web page

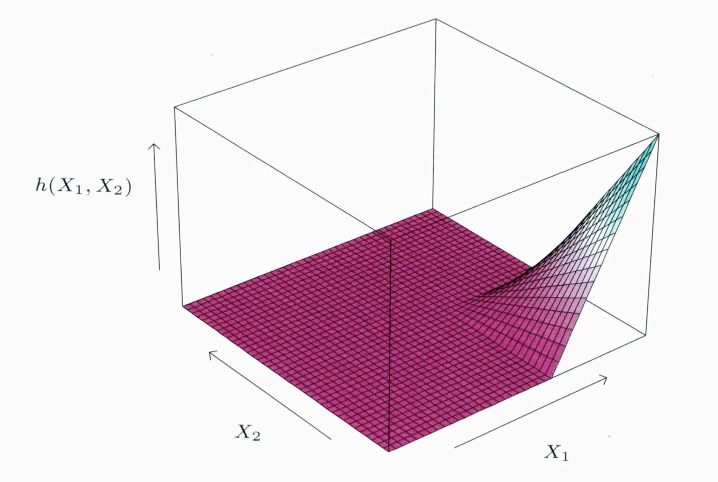
The MARS family of regressors F

1) the constant function is in F

2) for every $i \in Z$ and $c \in R$, the function $f_{i,c}(x) = max(0, x_i - c)$, is in F

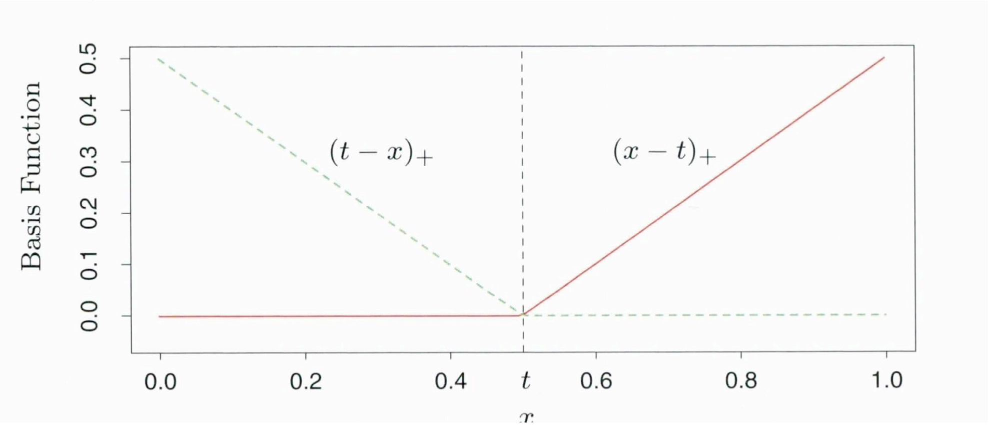
3) if $f_1$and $f_2$are in F, then $f_1 * f_2$ is in F

- if a piecewise linear additive model is desired, we can exclude 3)



MARS: funny notation

$(x-t)_+ = max(0,x-t)$
$(t-x)_+ = max(0,t-x)$



These two functions of x are called a “reflected pair”.

For categorical variables can use indicator (0/1) functions.

MARS: forward phase

- Choose a sequence of basis functions h_m, starting with the constant$h_0 = 1$
- repeat from i = 1 to n
    - Search for a reflected pair on variable j, inflection point t, along with one of the previous bases h_l with $l \in [0, 2(i-1)]$. Each j, t, l choice corresponds to adding these two functions to the model:

      $(x-t)_+ h_l$ and $(t-x)_+ h_l$
    - Choose j, t, l based on the maximum decrease of the loss function (sum of squares error)
    - Perform an OLS regression to determine the new weights β_l for the h_l.

MARS: backward phase

- Arrange to have looped enough times n so that the model is now overfit
- For each size n down to 1, find the model term which contributes least to the reduction of mean squared training error, yielding a best model for every size
- Use cross validation to determine which size n is the “best”, or to save computer cycles, minimize an $\overline{R}^2$like penalty:

$$GCV(\lambda) = \frac{\sum_{i=1}^{N} (y_i - f_\lambda(x_i))^2}{(1 - M(\lambda)/N)^2}$$

$M(\lambda)$is the “effective degrees of freedom”. Count 1 for every β and 2 for every t. (The count of β and t can be different in the presence of quadratic interactions.) [Friedman 1991] p19-20 for more info.

MARS: Solution to overfitting

- add only one dimension at a time
- add interactions only with previously added dimensions
- overfit on purpose while growing the model (forward)
- prune with cross validation (backward)

MARS computational cost tricks

(But isn't that too slow?)

- Presort by $x_i$for every i.

- Use a recurrence of squares to evaluate $(y_i - f_{i,c}(x))^2$for every value of c along one axis in the data x_i.

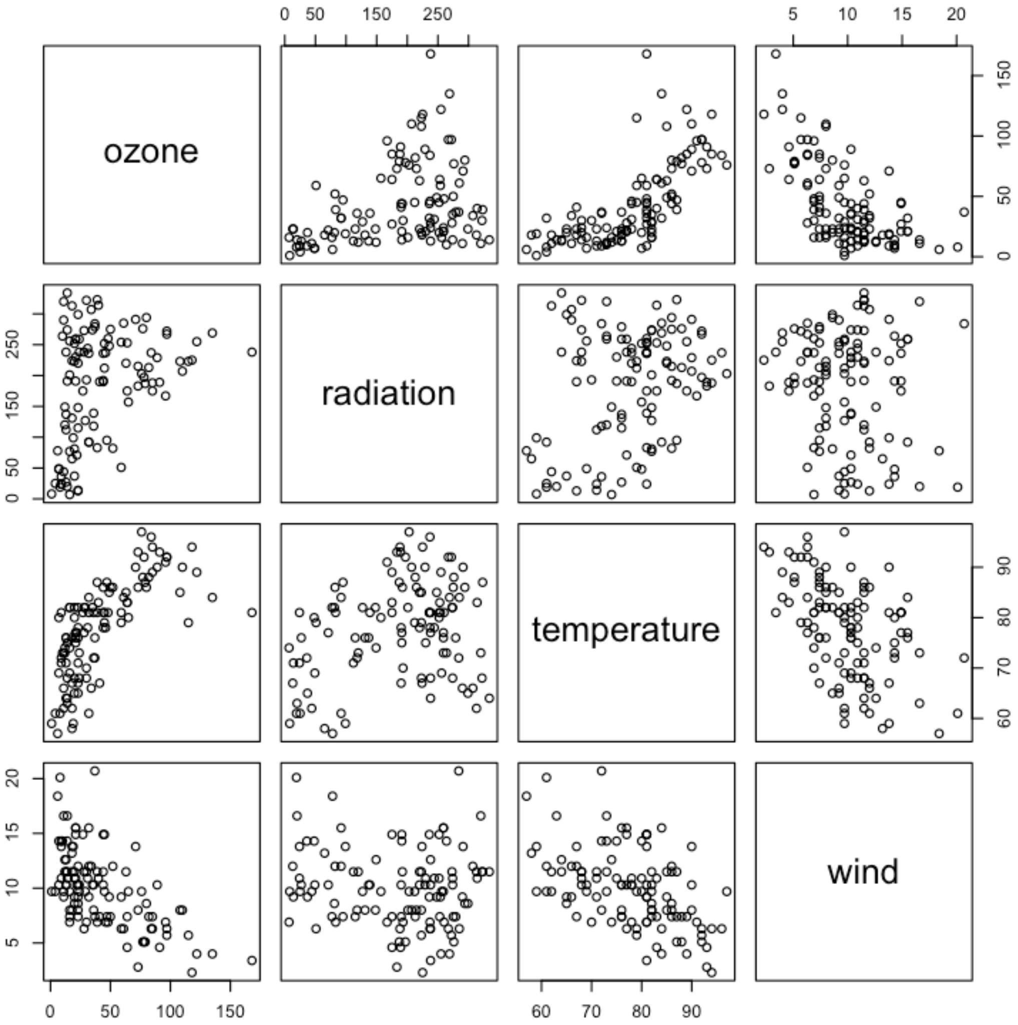
- Use a fancy data structure (a "priority queue") to only evaluate least squares on "important" choices of (i,c).

- Result is similar cost to stepwise linear regression.

MARS advantages

- produces a continuous model (actuaries like)
- piecewise linear model subspace handles phenomena such as “dose response”
- automatically chooses inflection points
- can handle categorical inputs along with continuous inputs
- handles 100-1000 dimensions (until you run out of computer cycles)

MARS: an example



MARS: an example (“earth” R package)

```
> a <- earth(ozone ~ ., data = ozone)
> summary(a, digits = 2, style = "max")
Call: earth(formula=ozone~., data=ozone)

ozone =
 12
 +  0.29 * max(0,   radiation -             175)
 -  0.48 * max(0,   radiation -             238)
 +   3.5 * max(0, temperature -              76)
 -   6.3 * max(0, temperature -              90)
 +        10 * max(0,             8.6 -            wind)

Selected 6 of 13 terms, and 3 of 3 predictors
Importance: temperature, wind, radiation
Number of terms at each degree of interaction: 1 5 (additive model)
GCV 329        RSS 29615        GRSq 0.7        RSq 0.76
```

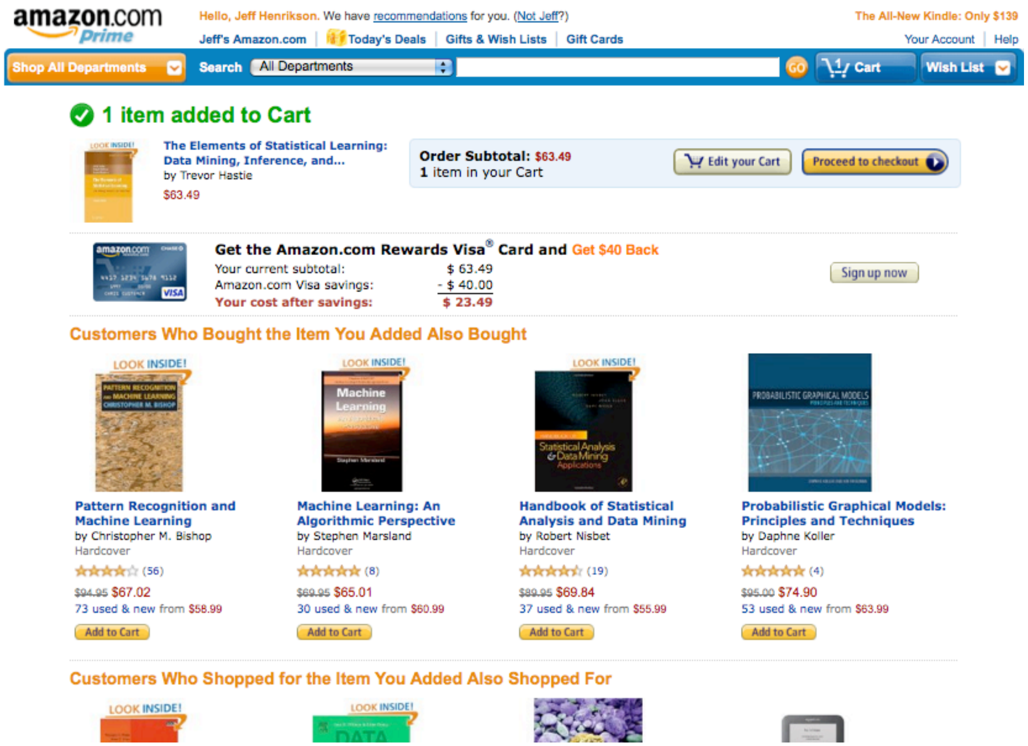
MARS: gotchas

- No one “true” model. Variables and inflection points may change over time (possibly back and forth).
- No built-in relationship for versioning, e.g. “in the new version don’t change more than 2 variables”, though Salford tools may have configurations like this.
- Salford Systems tools are expensive
- earth is GPL3, but mda.mars doesn’t fully support categorical variables

Part 2

Success with large data sets

The recommendations problem



Hello, Jeff Henrikson. We have recommendations for you. (Not Jeff?)

The All-New Kindle: Only $139

✓ 1 item added to Cart

The Elements of Statistical Learning: Data Mining, Inference, and...
by Trevor Hastie
$63.49

Order Subtotal: $63.49
1 item in your Cart

Get the Amazon.com Rewards Visa® Card and Get $40 Back

Your current subtotal:	$ 63.49
Amazon.com Visa savings:	- $ 40.00
Your cost after savings:	**$ 23.49**

Customers Who Bought the Item You Added Also Bought

Pattern Recognition and Machine Learning
by Christopher M. Bishop
Hardcover
★★★★☆ (56)
~~$94.95~~ $67.02
73 used & new from $58.99

Machine Learning: An Algorithmic Perspective
by Stephen Marsland
Hardcover
★★★★★ (8)
~~$69.95~~ $65.01
30 used & new from $60.99

Handbook of Statistical Analysis and Data Mining
by Robert Nisbet
Hardcover
★★★★½ (19)
~~$89.95~~ $69.84
37 used & new from $55.99

Probabilistic Graphical Models: Principles and Techniques
by Daphne Koller
Hardcover
★★★★★ (4)
~~$95.00~~ $74.90
53 used & new from $63.99

Customers Who Shopped for the Item You Added Also Shopped For

The recommendations problem

(aka collaborative filtering)

Suppose you have tuples (person, item) representing purchases (or any interest). Determine other items the person may be willing to purchase.

The simplest possible recommender

conditional probability!

$P(item\ x \mid item\ y) = P(item\ x \bigcap\ item\ y) / P(item\ y)$

estimate with by counting people:

= count of people buying item x and item y / count of people buying y

Getting started

- 1 year of data is tens of times larger than RAM
- Typically requires D/M passes over the data
- Hand coding typically 10x faster than SQL on comparable hardware
- Hash join D/M times is typically enough
- Hash join is easy to write

Netflix prize

- Netflix posts data set to run public contest (2007-2009)
- Prize was $1M cash
- Early leaders present at ACM KDD conference for $50k prize

www.netflixprize.com

Using correlation+nearest neighbors as smoothing function

(from Y. Kohen at early leaders KDD talk)

- Problem with formulas like

$P(item\ x \mid item\ y) = P(item\ x \bigcap item\ y) / P(item\ y)$

Is not enough people in the denominator.

- Correlation function is computable just like conditional probability
- Associate small items y with its k nearest neighbors N_k(y) (by correlation)
- Then run conditional probability, but condition on N_k(y) instead of y
- IIRC, ~50% of the way from simplest possible solution to winning solution
- Most other winning tricks were much harder

Engineering concerns (scaling, fault recovery, availability)

- Will your data grow bigger?
- Can other services make you fail?
- How often do you need to update? Display? Rebuild model?

Your friendly neighborhood programmer may be able to help

Even better than SQL: Pig!

Advantages:

- rich data types (cells can contain bags, lists, tuples, dictionaries)
- more natural query descriptions
- close to perfect data size and processing speed scaling on map reduce clusters

SQL

- structured data means normalization
- normalization means writing meaningless joins
- normalilzation means wasting precious IOPS (IOs per second)
- completely based on sets, so sequence is abhorred

What is "Map reduce"?

- n similar machines wired together in a network
- data is decentralized
- every query breaks into 2 parts
- “map” part is 100% parallelizable
- “reduce” part is 80% parallelizable

Pig: an example

```
quote; 3485; 9482; 2011-03-07; 123-45-6789; {(2004, honda, civic),(1980, ford,
F150)}; {(m, 55), (f, 54), (m, 17)}; 1000
quote; 3956; 9482; 2011-03-09; 123-74-1234; {(2004, honda, civic),(1980, ford,
F150)}; {(m, 55), (f, 54), (m, 17)}; 1100
quote; 3975; 9482; 2011-03-11; 123-45-6789; {(2004, honda, civic),(1980, ford,
F150)}; {(m, 55), (f, 54), (m, 17)}; 1000
close; 1209; 9482; 2011-03-11;;;;;;3975
```

```
quotes = load 'quotes2.txt' using PigStorage(';') as (kind,id:int,agent:int,t,ssn,
vehicles:bag{Tvehicle:tuple(year:int,make:chararray,model:chararray)},
drivers:bag{Tdriver:tuple(sex:chararray,age:int)}, premium:double, idprev:int);
```

Other data mining software/standards:

KNIME

Weka/Pentaho

PMML/DMG

Final thought: Cross Validation for Loss Ratios?

“We use a train/test methodology to build and evaluate models. This means that the modeling dataset is randomly divided into two samples, called the training and test samples. A number of models are fit on the training sample, and these models are used to "score" the test sample. The test sample therefore contains both the actual loss ratio (or any other target variable) as well as the predicted loss ratio, despite the fact that it was not used to fit the model. The policies in the test sample are then sorted by the score, and then broken into (for example) ten equal-sized pices, called deciles. Loss ratio, frequency, and capped loss ratio are computed for each decile. These numbers constitute "lift curves". A model with a low loss ratio for the "best" decile and a very high loss ratio for the "worst" decile is said to have "large lift". We believe that the lift curves are as meaningful for measuring the business value of models as such traditional statistical measures as mean absolute deviation or R^2.

Peter Wu, FCAS, James Guszcza, ACAS.
http://www.casact.org/pubs/forum/03wforum/03wf113.pdf
http://www.datashaping.com/deloitte.ppt

My question: Why lift chart and not plain testing error?

More about the Lift Chart

An economically optimal model application:

- A model is built for a direct mail campaign to predict probability of purchase.

- Sending mail costs a constant payment

- Choose the number of households to receive mail that maximizes expected profit

Solution: use cross-validation to separate true from false positives at every possible sending threshold x. Graph cost x against benefit y:

$x(t)=\frac{TP(t)+FP(t)}{P+N}$, $y(t)=TP(t)$

Vuk, Curk: ROC Curve, Lift Chart and Calibration Plot, Metodološki zvezki, Vol. 3, No. 1. (2006), pp. 89-108. (citeseerx has pdf)

How to learn more

[HTF] - Elements of Statistical Learning, Hastie, Tibshirani, and Friedman.
(several graphs in these slides are from here)
- Association of Computing Machinery special interest group in Knowledge Discovery and Data Mining

[Friedman 1991] - Friedman, J. H. (1991). "Multivariate Adaptive Regression Splines". Annals of Statistics.
- Friedman, J. H. (1993) Fast MARS, Stanford University Department of Statistics, Technical Report 110
- Friedman, J. H. (1993) Estimating Functions of Mixed Ordinal and Categorical Variables Using Adaptive Splines, New Directions in Statistical Data Analysis and Robustness (Morgenthaler, Ronchetti, Stahel, eds.), Birkhauser

Special thanks

Q and A