Text Mining Using Linear Models of Latent States

Bob Stine
Department of Statistics
The Wharton School, University of Pennsylvania
www-stat.wharton.upenn.edu/~stine

Topics

- Application
  - Statistical named entity recognition
- Feature creation
  - Preprocessing
  - Converting text into numerical data
- Exploiting the features
  - Estimators, standard errors
  - Auctions and experts

- Collaborators
  - Dean Foster in Statistics
  - Lyle Ungar in CS

Application and Motivation

Text Mining Applications

- Cloze
  - What's the next word?
    "...in the midst of modern life the greatest, __"
  - Data compression
- Word disambiguation
  - Meaning of a word in context
  - Does "Washington" refer to a state, a person, a city or perhaps a baseball team? Or politics?
- Speech tagging
  - ~~Identifying parts of speech~~
  - Distinguishing among proper nouns
- Grading papers, classification, ...

Named Entity Recognition

- Annotate plain text in a way that identifies the words that refer to a
  - person (Obama)
  - place (France)
  - organization (Google)

  or something else.
- Wiki example

  Jim bought 300 shares of Acme Corp in 2006.

  person company year
- Customized systems build on grammatical heuristics and statistical models.
  - Time consuming to build
  - Specific to training domain

Second Example

- You get some text, a sequence of "words"
  - bob went to the 7-11 <.> he was hungry <.> ...
- Task is to tag proper nouns, distinguishing those associated with people, places and organizations.

Washington?
person
place
team
politics

- No other information in the test set
- Training data
  - Marked up sequence that includes the tags that you'd ideally produce
  - bob went to the 7-11 <.> he was hungry <.> ...

    person organization
- Test data is just a sequence of "words"

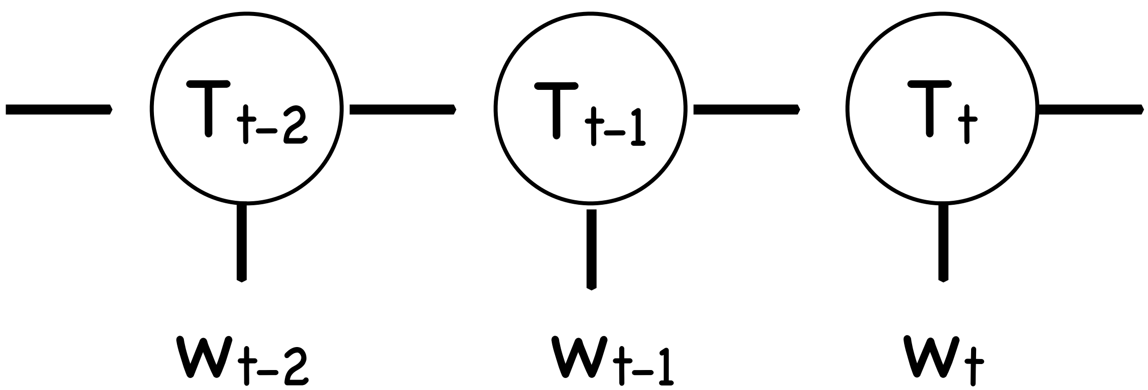
Approaches

- Numerous methods used for NER
  - Gazette
    - lists of proper words/businesses, places
  - Formal grammar, parse trees
    - off the shelf parsing of text into subject/verb
  - Stemming
    - such as noting prior word ends in -ing
  - Capitalization
- Not using any of these…
  - Things like capitalization are not available in some formats, such as text from speech
  - Generalization: gazettes depend on context
  - Languages other than English

Could add these later!

Statistical Models for Text

- Markov chains
  - Hidden Markov models have been successfully used in text mining, particularly speech tagging
- Hidden Markov model (HMM)
  - Transition probabilities for observed words
    $P(w_t|w_{t-1},w_{t-2},\ldots)$ as in P(clear|is,sky,the)
  - Instead specify model for underlying types
    $P(T_t|T_{t-1},T_{t-2},\ldots)$ as in P(adj|is,noun,article)
    with words generated by the state



Concentrate dependence in transitions among relatively few states

State-Based Model

- Appealing heuristic of HMM
  Meaning of text can be described by transitions in a low-dimensional subspace determined by surrounding text
- Estimation of HMM hard and slow
  - Nonlinear
  - Iterative (dynamic programming)
- Objective
  - Linear method for building features that represent underlying state of the text process.
    - Possible? Observable operator algebras for HMMs.
  - Features used by predictive model. Pick favorite.

Connections

- Talks earlier today...

- Probabilistic latent semantic analysis
- Non-negative matrix factorization (NMF)
- Clustering

Building the Features

Summary of Method

- Accumulate correlations between word occurrences in n-grams
  - Preprocessing, all n-grams on Internet
  - Trigrams in example; can use/combine with others
- Perform a canonical correlation analysis (CCA) of these correlations
  - Singular value decomposition (SVD) of corr mat
- Coordinates of words in the space of canonical variables define "attribute dictionary"
- Predictive features are sequences of these coordinates determined by the order of the works in the text to be modeled

Canonical Correlation

- CCA mixes linear regression and principal components analysis
- Regression
  Find linear combination of $X_1,\ldots,X_k$ most correlated with Y
  $\max \operatorname{corr}(Y, \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_k X_k)$
- Canonical correlation
  Find linear combinations of X's and Y's that have maximal correlation
  $\max \operatorname{corr}(\alpha_1 Y_1+\ldots+\alpha_j Y_j, \beta_1 X_1+\ldots+\beta_k X_k)$
- Solution is equivalent to PCA of

$$(\Sigma_{XX})^{-1/2}\ \Sigma_{XY}\ (\Sigma_{YY})^{-1/2}$$

covariance matrices

Coincidence Matrices

	Pre-word $w_1,w_2,w_3,\ldots,w_d$	Word $w_1,w_2,w_3,\ldots,w_d$	Post-word $w_1,w_2,w_3,\ldots,w_d$
w_1,w_2,w_3			
$\vdots$			
w_{t-1},w_t,w_{t+1}	0 1 0 0 0 0 0 0 0	0 0 0 0 1 0 0 0 0	0 0 0 0 0 0 0 1 0
	B_1	B_w	B_2
billions of n-grams $\vdots$			
w_{n-2},w_{n-1},w_n			

$d = 50{,}000$

d is the size of our dictionary

Using CCA

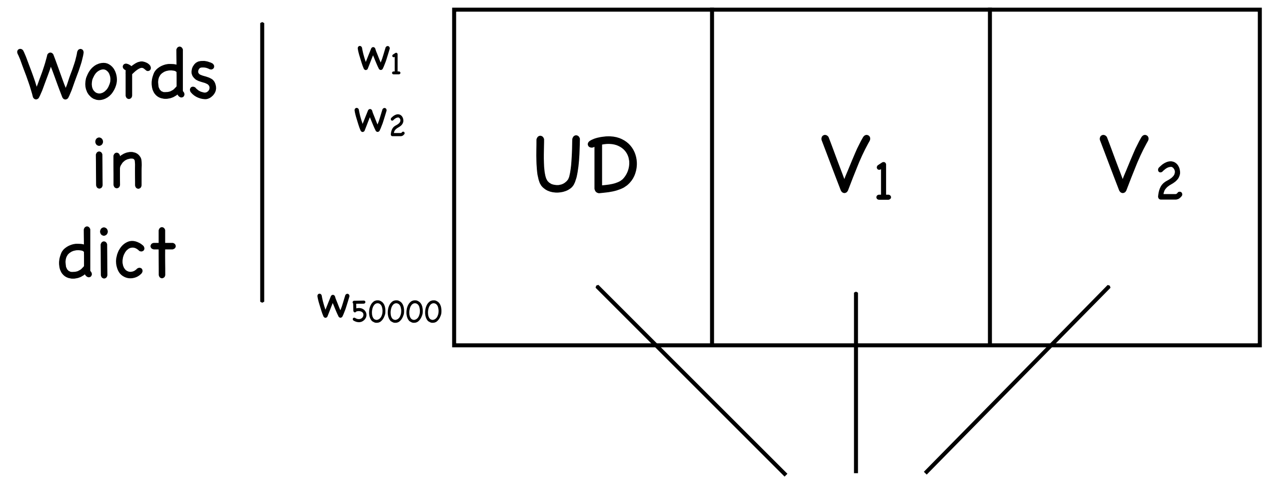
- Which words, or groups of words, co-occur?
- Linear
  Find α_1 in R^d and β_1 in R^{2d} that together
  maximize $\mathrm{corr}(B_w\alpha, [B_1,B_2]\beta)$
  (α_1,β_1) defines first pair of canonical variables
- Subsequent pairs as in principle components
  Find (α_2,β_2) which
  maximize $\mathrm{corr}(B_w\alpha, [B_1,B_2]\beta)$
  while being orthogonal to (α_1,β_1).
- We compute about K=30 to 100 of these canonical coordinates

Canonical Variables

- SVD of correlations $C \approx B_w'[B_1\ B_2]$

$$\underset{}{C} = \underset{(50{,}000 \times 50)}{U}\ \underset{(50 \times 50)}{D}\ \underset{(50 \times 100{,}000)}{V'} = UD[V_1'\ V_2']$$

- Attribute dictionary



K=50 columns in each bundle

Random Projections

- Faster calculation of CCA/SVD
- Computing canonical variables

$$C = B_W'[B_1\ B_2]$$

50,000 x 100,000 is large
- Random projection
  - Low rank approximations
  - Reference Halko, Martinsson, Tropp 2010
  - Two stage approach
    (1) Project into "active" subspace
    (2) Do usual operation

Algorithm for SVD

- Want SVD of correlations (omit scaling)

  $$C = B_W'[B_1\ B_2] = UDV'$$

- Find orthonormal Q with K+m columns for which

  $$\|C - QQ'C\|_2 \text{ is small}$$

- Random projection

  $Q \sim N(0,1)$ works very well!

- Steps
  - Compute coefficients $H = Q'C$
  - SVD of H is U_1DV'
  - Compute $U = QU_1$
- To get rank K, need a few extra columns (m)

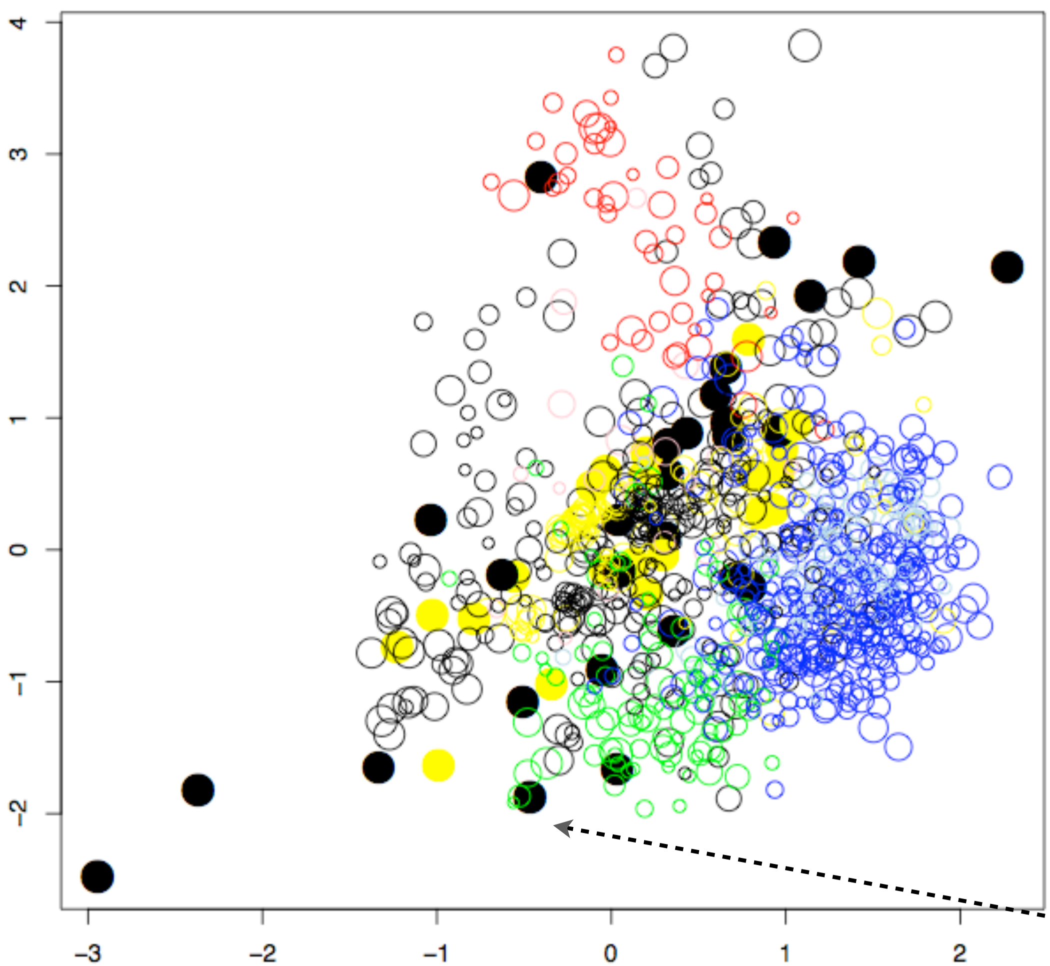
Plots of Attribute Dict

- Isolate the coordinates in the attribute dictionary assigned to "interesting words"
  - Words were not picked out in advance or known while building the attribute dictionary
- Several views
  - Grouped/colored by parts of speech
  - Names
    - Common US names, casual and formal
    - Bob and Robert
  - Numbers
- Plots show projections of the coordinates in the attribute dictionary…

Parts of Speech

- Projection from attribute dictionary

noun
verb
adj
unk

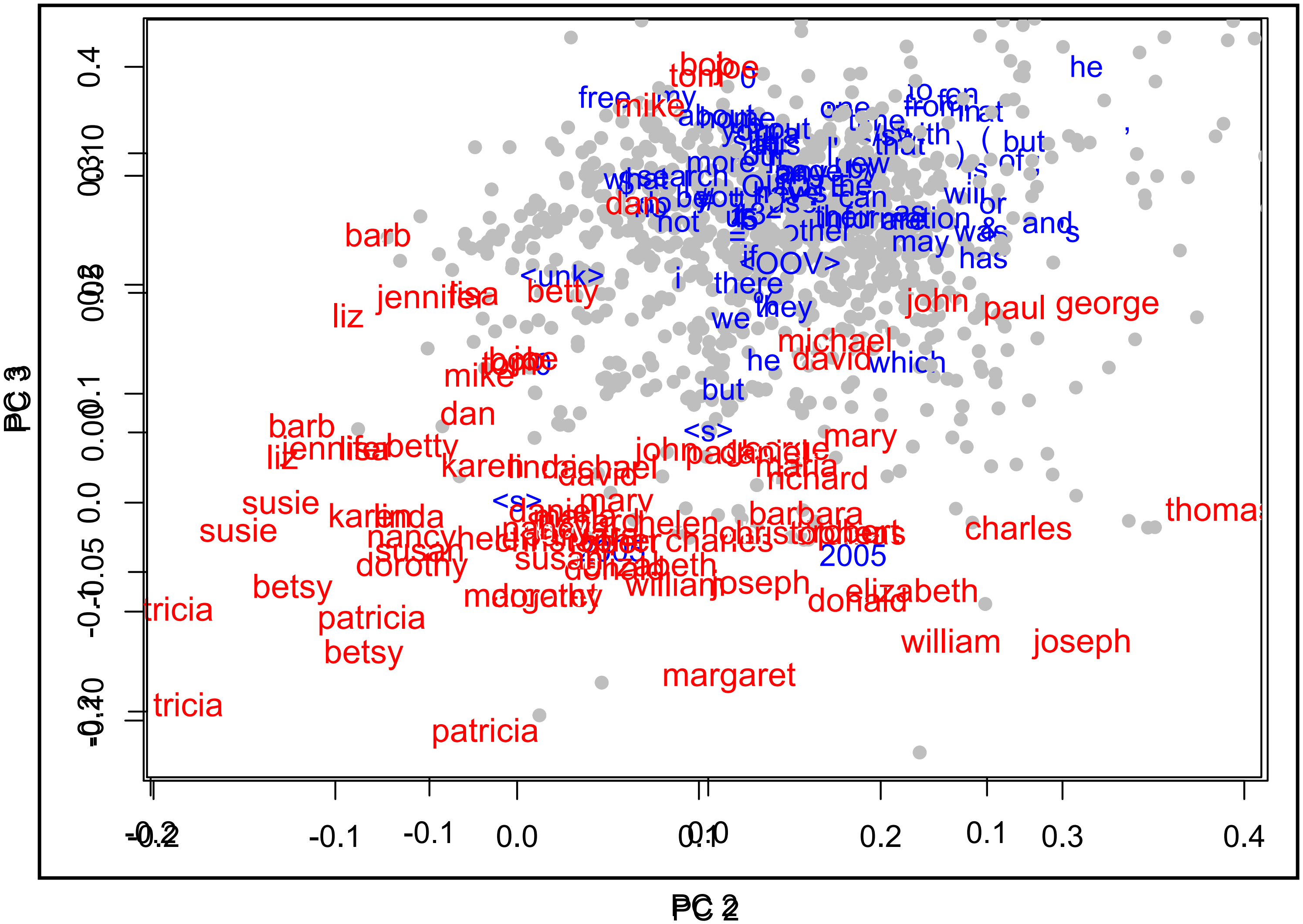


Words from d=10,000 dictionary

Not in dictionary

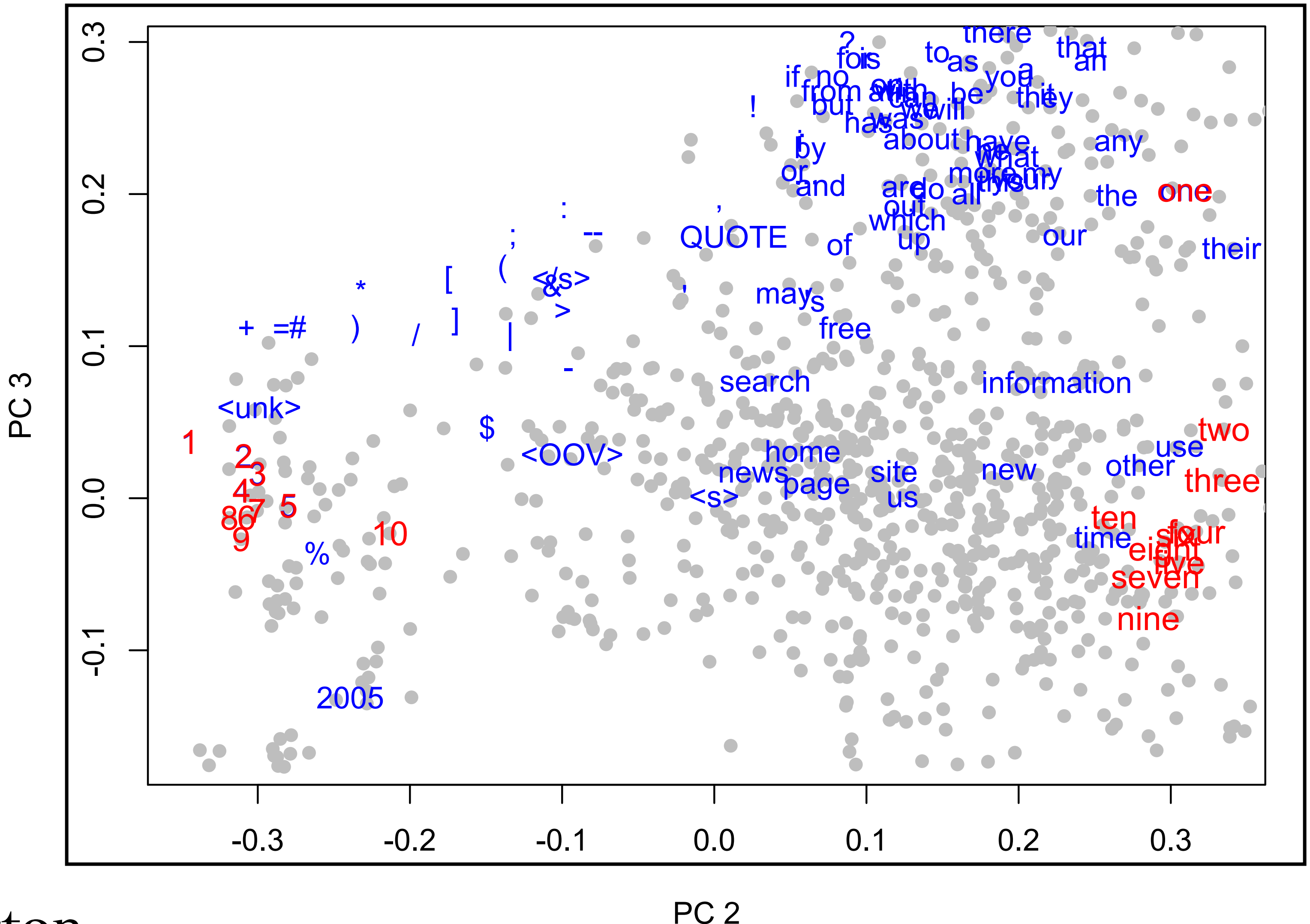
Closer Look at Features

- Focus on a few names



Closer Look at Features

- Numbers as words and digits



Features

- Sequence of words in the document determine the features in the predictive model.
- Further processing, such as exponential smoothing of various lengths

Document	Features from Attr Dictionary		
w_1	$UD[w_1]$	$V_1[w_1]$	$V_2[w_1]$
w_2	$UD[w_2]$	$V_1[w_2]$	$V_2[w_2]$
w_3	$UD[w_3]$	$V_1[w_3]$	$V_2[w_3]$
…		…	
w_n	$UD[w_n]$	$V_1[w_n]$	$V_2[w_n]$

3K features

Predictive Models

Components

- Multiple streaming variable selection
  - Depth-first, guided selection
- Auction framework
  - Blend several strategies
    - raw data, calibration, nonlinearity, interaction
  - Formalize external expert knowledge
- Statistics: Estimates and standard errors
  - Sandwich estimator for robust SE
  - Shrinkage
- Sequential testing
  - Alpha investing avoids need for tuning data
  - Martingale control of expected false discoveries
- Or your favorite method (e.g. R package glmnet)

Based on Regression

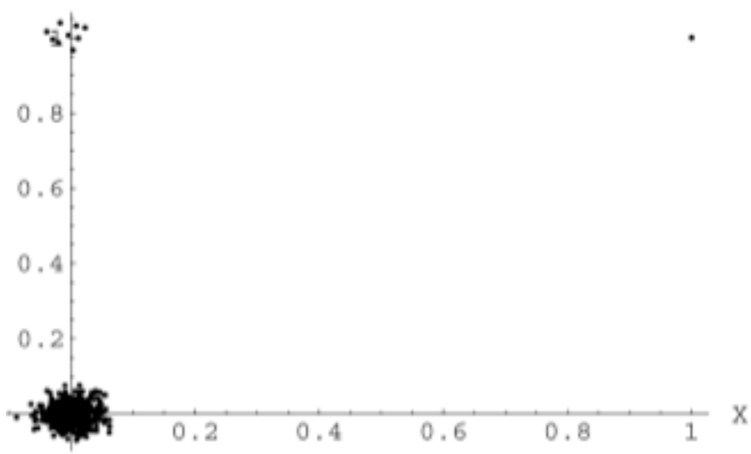
- Familiar, interpretable, good diagnostics
- Regression has worked well
  - Predicting rare events, such as bankruptcy
    Competitive with random forest
  - Function estimation, using wavelets and variations on thresholding
  - Trick is getting the right explanatory variables
- Extend to rich environments
  - Spatial-temporal data
    Retail credit default — MRF, MCMC
  - Linguistics, text mining
    Word disambiguation, cloze — TF-IDF
- Avoid overfitting…

TF-IDF: term frequency-inverse document frequency
frequency in document relative to frequency in corpus

MRF: Markov random fields

Lessons from Prior Work

- "Breadth-first" search
  - Slow, large memory space
  - Fixed set of features in search
  - Severe penalty on largest z-score, sqrt(2 log p)
- If most searched features are interactions, then most selected features are interactions
  - $\mu \gg 0$ and $\beta_1, \beta_2 \neq 0$, then $X_1 {*} X_2 \Rightarrow c + \beta_1 X_1 + \beta_2 X_2$
- Outliers cause problems even with large n

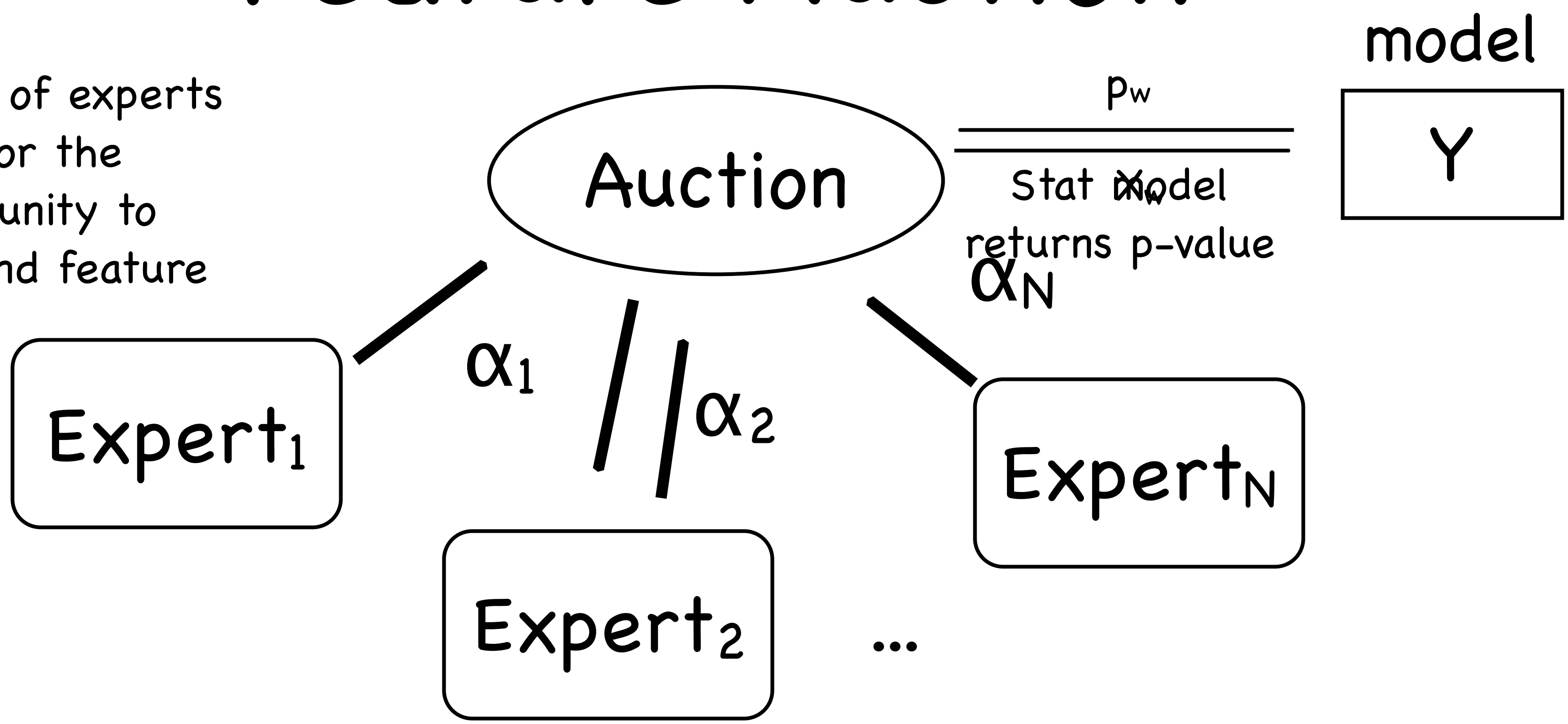


Real p-value ≈ 1/1000,
but
usual t-statistic ≈ 10

Wharton
Department of Statistics

Feature Auction

Collection of experts
bid for the
opportunity to
recommend feature



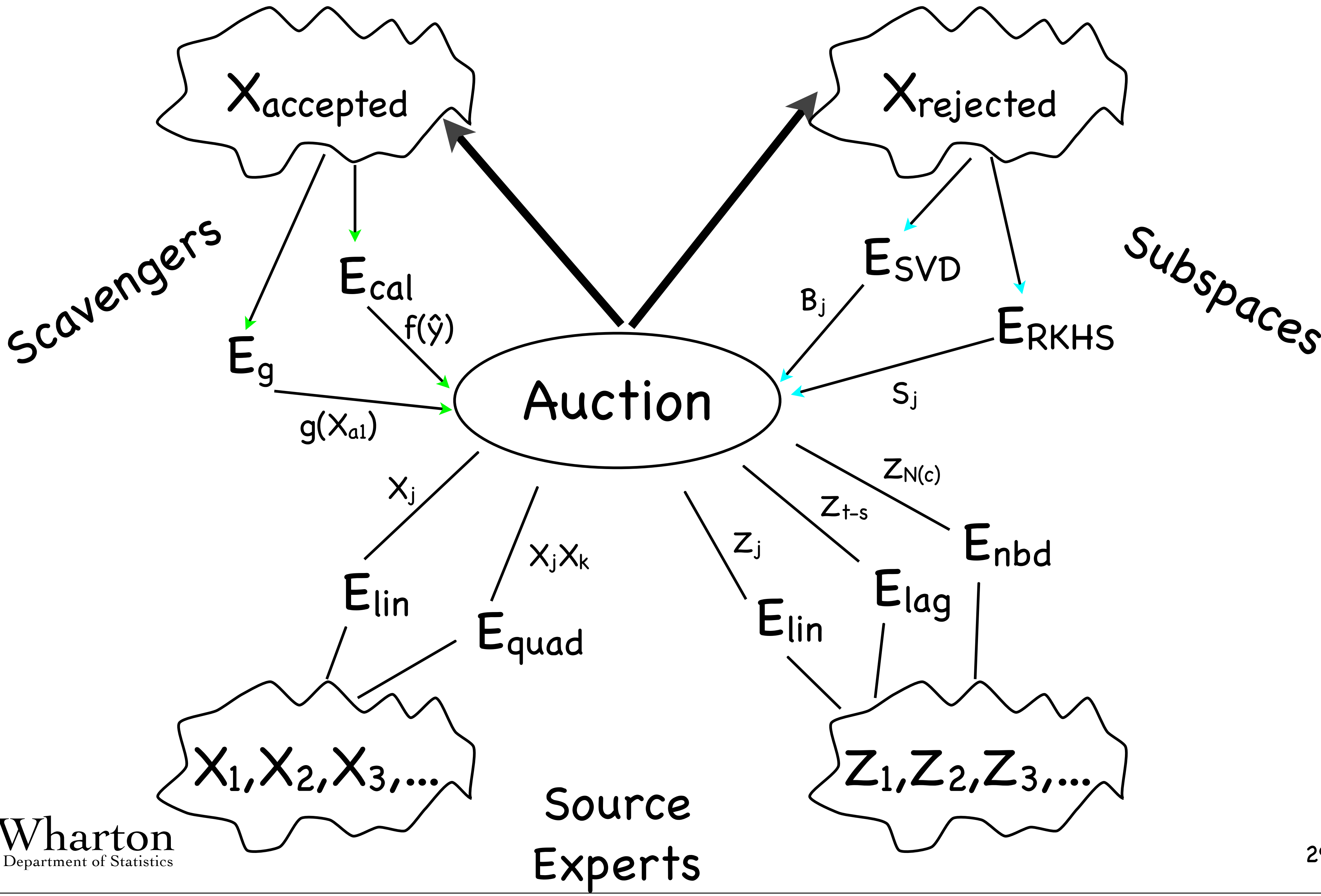
Auction collects
winning bid α_2

Expert supplies
recommended feature X_w

Expert receives payoff ω
if $p_w \leq \alpha_2$

Experts learn if the bid was accepted,
not the effect size or p_w.

Experts


$X_{accepted}$
$X_{rejected}$
Scavengers
Subspaces
E_{cal}
E_g
$f(\hat{y})$
$g(X_{a1})$
E_{SVD}
E_{RKHS}
B_j
S_j
Auction
X_j
X_jX_k
Z_j
Z_{t-s}
$Z_{N(c)}$
E_{lin}
E_{quad}
E_{lin}
E_{lag}
E_{nbd}
$X_1, X_2, X_3, \ldots$
$Z_1, Z_2, Z_3, \ldots$
Source Experts

Experts

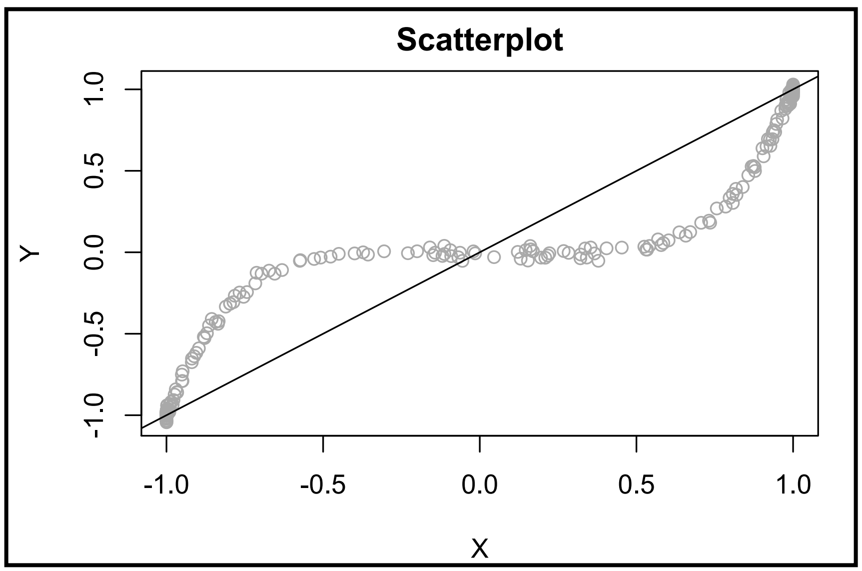
- Strategy for creating sequence of possible explanatory variables.
  - Embody domain knowledge, science of application.
- Source experts
  - A collection of measurements (CCA features)
  - Subspace basis (PCA, RKHS)
  - Multiple smooths of context variables
  - Interactions between within/between groups
- Scavengers
  - Interactions
    - among features accepted/rejected by model
  - Transformations
    - segmenting, as in scatterplot smoothing
    - polynomial transformations
- Calibration

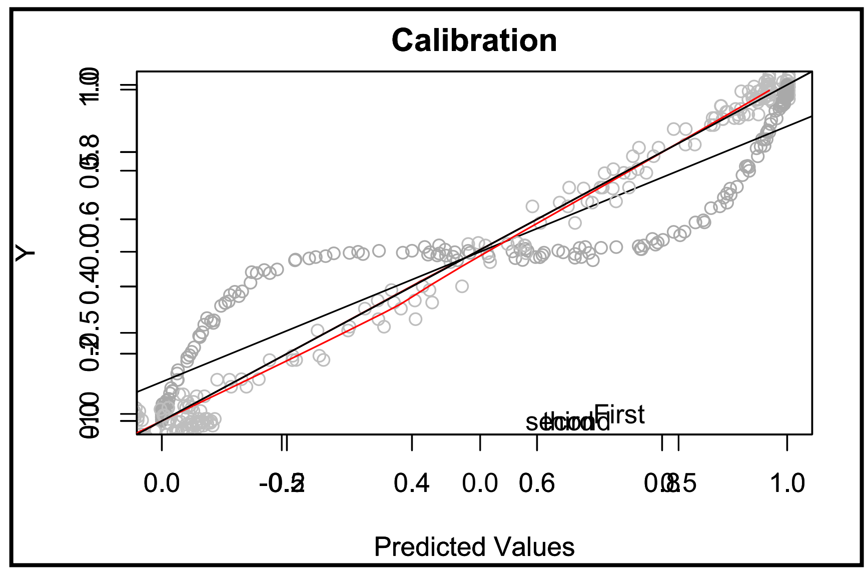
Calibration

- Simple way to capture global nonlinearity
  - aka, nonparametric single-index model
- Predictor is calibrated if

$$E(\hat{Y}) = Y$$

- Simple way to calibrate a model is to regression Y on $\hat{Y}^2$ and $\hat{Y}^3$ until linear.





Expert Wealth

- Expert gains wealth if feature accepted
  - Experts have alpha-wealth
  - If recommended feature is accepted in the model, expert earns ω additional wealth
  - If recommended feature is refused, expert loses bid
- As auction proceeds...
  - Reward experts that offer useful features. These then can afford later bids and recommend more X's
  - Eliminate experts whose features are not useful.
- Taxes fund parasites and scavengers
  - Continue control overall FDR
- Critical
  - control multiplicity in a sequence of hypotheses
  - p-values determine useful features

Robust Standard Errors

- p-values depend on many things
  - p-value = f(effect size, std error, prob dist)
  - Error structure likely heteroscedastic
  - Observations frequently dependent
- Dependence
  - Complex spatial dependence in default rates
  - Documents from various news feeds
  - Transfer learning
    When train on observations from selected regions or document sources, what can you infer to others?
- What are the right degrees of freedom?
  - Tukey story

Sandwich Estimator

- Usual OLS estimate of variance
  - Assume your model is true

$$\begin{aligned} var(b) &= (X'X)^{-1}X'E(ee')X(X'X)^{-1} \\ &= \sigma^2(X'X)^{-1}(X'X)\ (X'X)^{-1} \\ &= \sigma^2(X'X)^{-1} \end{aligned}$$

- Sandwich estimators
  - Robust to deviations from assumptions

heteroscedasticity

$$\begin{aligned} var(b) &= (X'X)^{-1}X'E(ee')X(X'X)^{-1} \\ &= (X'X)^{-1}\ X'D^2X\ (X'X)^{-1} \end{aligned}$$

diagonal

dependence

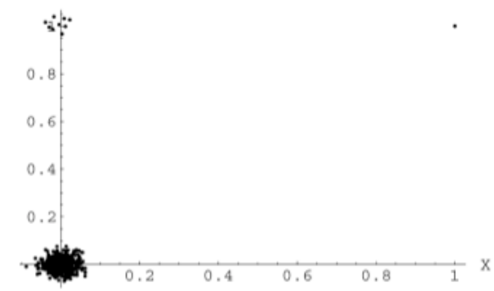
$$\begin{aligned} var(b) &= (X'X)^{-1}X'E(ee')X(X'X)^{-1} \\ &= \sigma^2(X'X)^{-1}\ X'BX\ (X'X)^{-1} \end{aligned}$$

block diagonal

Essentially the "Tukey method"

Flashback...

- Heteroscedastic errors
  - Estimate standard error with outlier
  - Sandwich estimator allowing heteroscedastic error variances gives a t-stat ≈ 1, not 10.

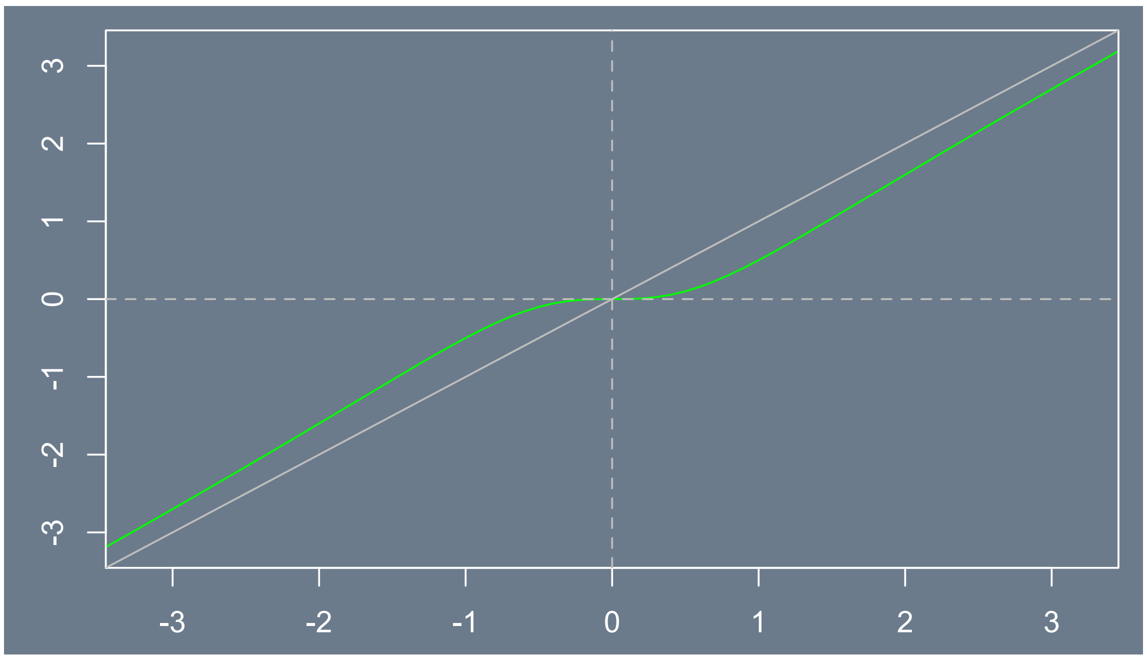


- Dependent errors
  - Even more critical to obtain an accurate SE
  - Netflix example
    Bonferroni (hard thresholding) overfits due to dependence in responses.
  - Credit default modeling
    Everything seems significant unless incorporate dependence into the calculation of the SE

Estimators

- Shrinkage
  - Two estimates of β_j: 0 and b_j
  - Std error determines the amount of shrinkage
    - Larger the t-statistic, the smaller the shinkage
  - Resembles Bayes estimator with Cauchy prior
  - "Smooth" version of hard thresholding

t-stat, shrunken estimate



t-stat, LS estimate

Wharton
Department of Statistics

Alpha Investing

- Context
  - Test possibly infinite sequence of m hypotheses

    $H_1, H_2, H_3, \ldots H_m \ldots$

    obtaining p-values $p_1, p_2, \ldots$
  - Order of tests can depend prior outcomes
- Procedure
  - Start with an initial alpha wealth $W_0 = \alpha$
  - Invest wealth $0 \leq \alpha_j \leq W_j$ in the test of H_j
  - Change in wealth depends on test outcome
  - $\omega \leq \alpha$ denotes the payout earned by rejecting

$$W_j - W_{j-1} = \begin{cases} \omega & \text{if } p_j \leq \alpha_j \\ -\alpha_j & \text{if } p_j > \alpha_j \end{cases}$$

Martingale Control

- Provides uniform control of the expected false discovery rate. At any stopping time during testing, martingale argument shows

$$\sup_{\theta} \frac{E(\#\text{false rejects})}{E(\#\text{rejects})+1} \leq \alpha$$

- Flexibility in choice of how to invest alpha-wealth in test of each hypothesis
  - Invest more when just reject if suspect that significant results cluster.
  - Universal investing strategies
- Avoids computing all p-values in advance

Multiple Testing

- Other methods are special cases
  - Note: alpha-investing does not require the full set of p-values or estimates at the start.
- Bonferroni test of $H_1,\ldots,H_m$
  - Set initial $W_0 = \alpha$ and reward to $\omega = 0.05$.
  - Bid $\alpha_j = \alpha/m$
- Step-down test of Benjamini and Hochberg
  - Set initial $W_0 = \alpha$ and reward to $\omega = 0.05$.
  - Test $H_1,\ldots H_m$ at fixed level α/m
  - If none reject -> finished.
  - If one rejects, earn $\alpha = 0.05$ for next round
  - Test next round conditionally on $p_j > \alpha/m$
    -> continue with remaining hypotheses.

Example...

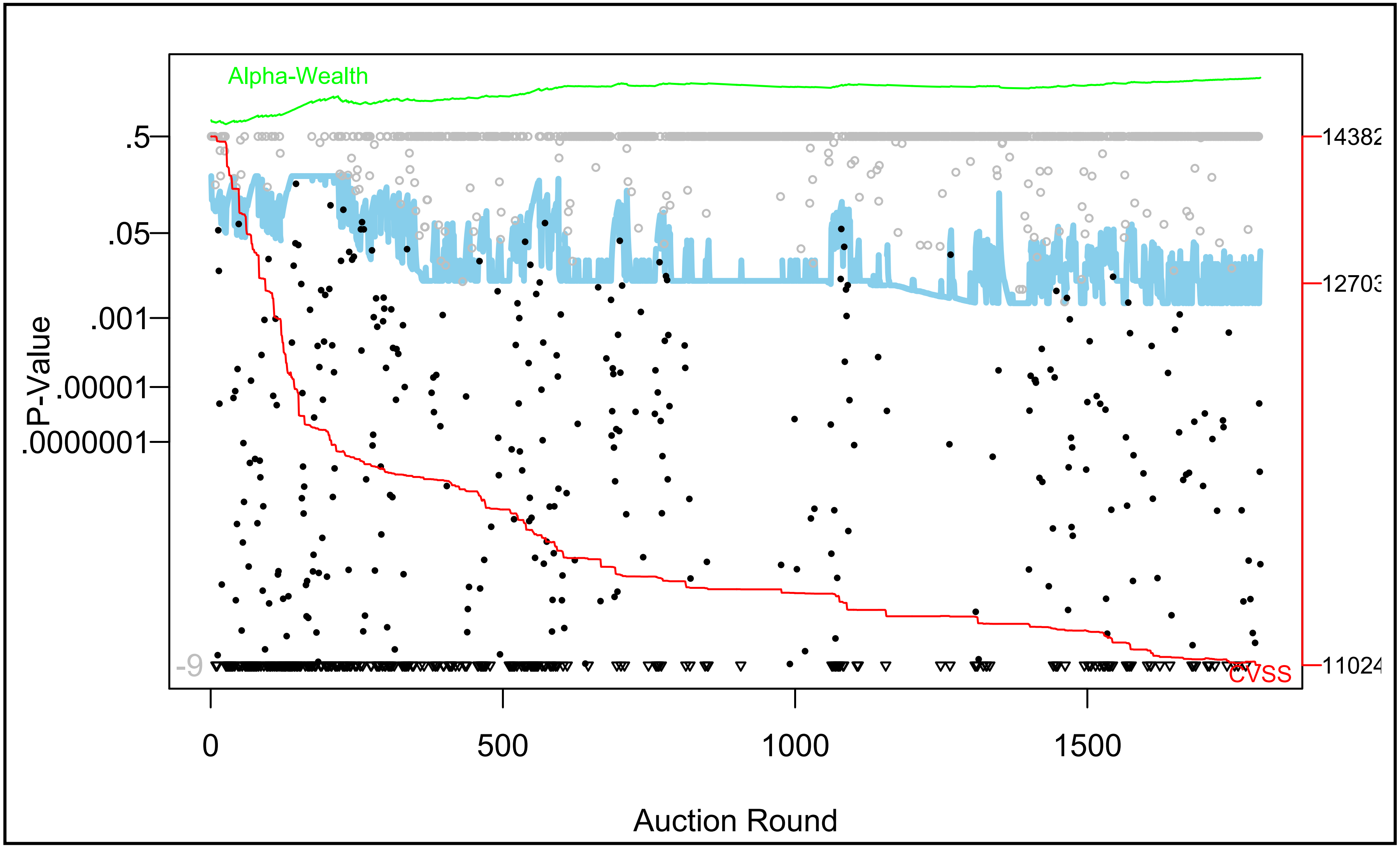
Back to text processing

Named Entity Results

- Model
  - Approximate max entropy classifier
    - Fancy name for multinomial logit
  - Other predictive models can be used
- Data
  - Portion of the ConLL03 data
  - Training and test subsets
- Dictionary
  - d=50,000 words
  - Exponential smooths of content features
  - Interactions
- Precision and recall about 0.85

Auction Run

First 2,000 rounds of auction modeling.



What are the predictors?

- Interactions
  - Combinations of canonical variables
- Principal components of factors
  - Combinations of skipped features
  - RKHS finds some nonlinear combinations
- Calibration adjustments
  - Simple method to estimate single-index model

$$\hat{y} = g(b_0 + b_1 X_1 + \ldots + b_k X_k)$$

  Estimate g cheaply by building a nonlinear regression of y on linear $\hat{y}$.

Closing Comments

Next Steps

- Text
  - Incorporate features from other methods
  - Understanding the CCA
  - Other "neighborhood" features
- Theory
  - Develop martingale that controls expected loss.
  - Adapt theory from the "nearly black" world of modern statistics to "nearly white" world of text
- Computing
  - Multi-threading is necessary to exploit trend toward vast number of cores in CPU
  - More specialized matrix code

Linguistics ≈ Spatial TS

Text	Credit default
Predict word in new documents, different authors	Predict rates in same locations, but changing economic conditions
Latent structure associated with corpus	Latent temporal changes as economy evolves
Neighborhoods: nearby words, sentences	Neighborhoods: nearby locations, time periods
Vast possible corpus	70 quarters, 3000 counties. Possible to drill lower.
Sparse	May be sparse

References

- Feature auction
  - www-stat.wharton.upenn.edu/~stine
- Alpha investing
  - "α-investing: a procedure for sequential control of expected false discoveries", JRSSB. 2008
- Streaming variable selection
  - "VIF regression", JASA. 2011
- Linear structure of HMM
  - "A spectral algorithm for learning hidden Markov models", Hsu, Kakade, Zhang, TTI. 2008
- Random projections
  - "Finding structure with randomness", Halko, Martinsson, and Tropp. 2010

Thanks!