PHYSTAT05 Highlights

Harrison B. Prosper
Florida State University

Single Top Group
5 October 2005

Outline

- Q & A
- Optimal Classification
- Ensemble Methods
- R
- Conclusions

Questions & Answers

1. Is it ever sensible to map data from N variables to M > N variables?
   - Yes! The mapping (of course) does not increase the number of degrees of freedom, however, machine learning algorithms *can* converge faster if variables are added that exploit known structure in data

2. gof tests with systematics: is what we are doing reasonable?
   - Yes, it is *reasonable*!

More Questions & Answers!

3. Are flat priors dangerous?
   - They can be, especially in high dimensions.
   - However, even in low dimensions they can be problematic: a flat prior in cross-section, $\pi(\sigma) = 1$ should *not* be used with an acceptance prior $\pi(\alpha) > 0$ at $\alpha = 0$!

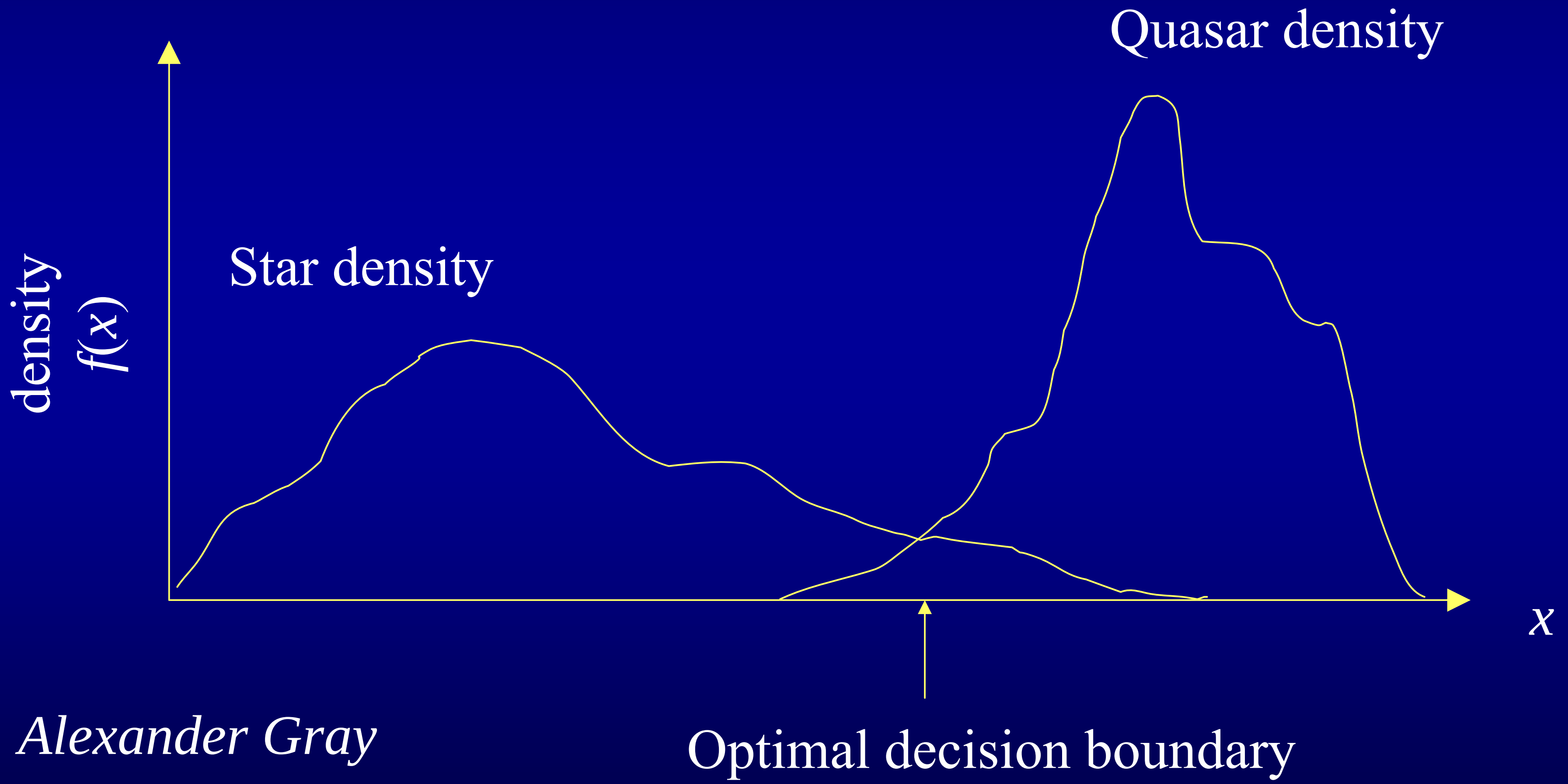
4. Is absolute coverage of upper limits necessary?
   - Only if you are a *statistical* fundamentalist!

Optimal Classification

- Popular Methods:
  - **Naïve Bayes**: fast but quadratic only
  - **Decision Tree**: fast but inaccurate
  - **Support Vector Machine**: accurate but slow
  - **Boosting**: accurate but requires thousands of classifiers
  - **Neural Net**: reasonable compromise but awkward/human-intensive to train

Alexander Gray et al.

Optimal Decision Theory



Alexander Gray

Optimal Decision Theory – II

$$P(C_1 \mid x) = \frac{f(x \mid C_1)\, P(C_1)}{f(x \mid C_1)\, P(C_1) + f(x \mid C_2)\, P(C_2)}$$

Bayes' Rule

C_1	Signal class
$f(x\|C_1)$	Signal (N-dim) density
C_2	Background class
$f(x\|C_2)$	Background (N-dim) density
$P(C_1) / P(C_2)$	Signal/Background ratio

Approximations to Bayes' Rule

Naïve Bayes $$f(x \mid C_k) \approx \prod_{i=1}^{M} h(x_i \mid C_k)$$

$h(x_i|C_k)$ 1-dim marginal densities

Nonparametric Bayes $$f(x \mid C) \approx \frac{1}{N} \sum_{r}^{N} K_h(\|x - x_r\|)$$

K_h is a kernel, such as

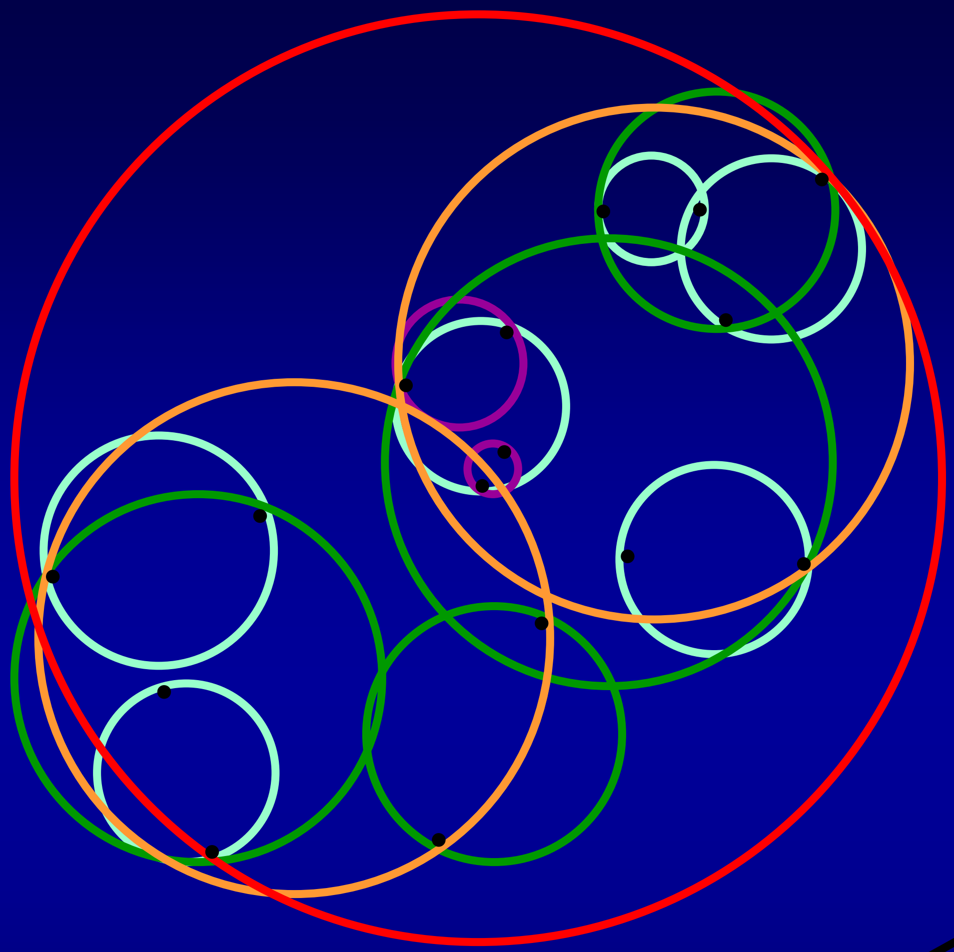
$$K_h(\|x - x_r\|) = \exp\{-(x - x_r)^T (x - x_r) / 2h^2\} / (2\pi h^2)^{N/2}$$

Fast algorithm for Kernel Density Estimation (KDE)

Alexander Gray

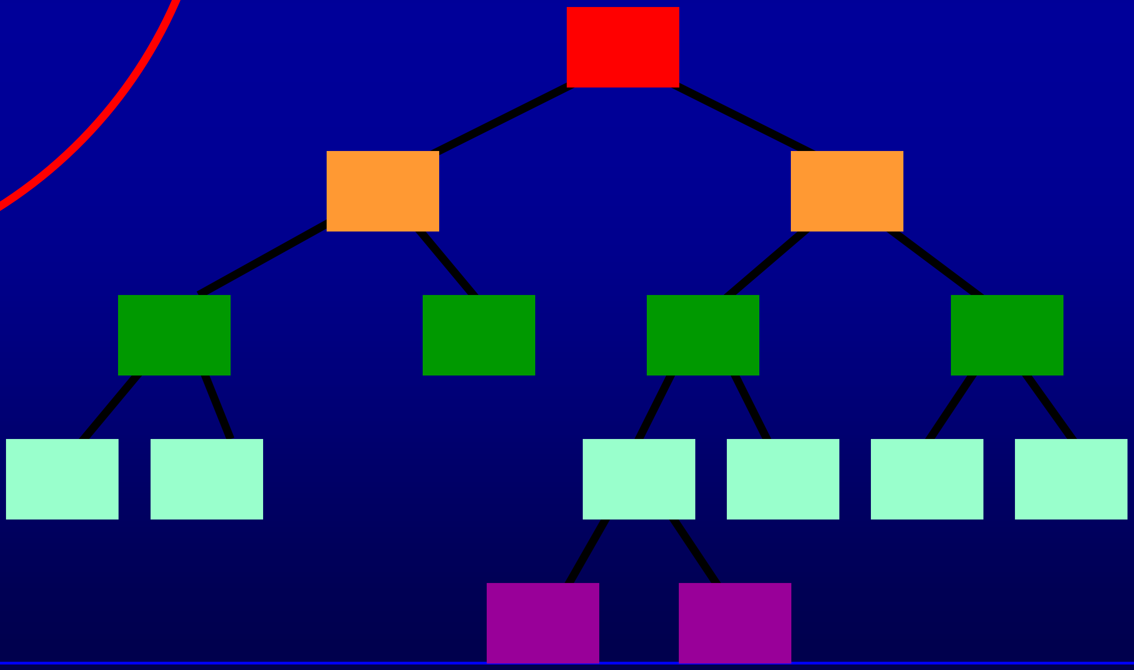
$$f(x) = \frac{1}{N} \sum_{r}^{N} K_h(\|x - x_r\|)$$

- Works in arbitrary dimensions
- The fastest method to date [Gray & Moore 2003]



Ball-trees

(computational geometry)



Alexander Gray

Ensemble Methods

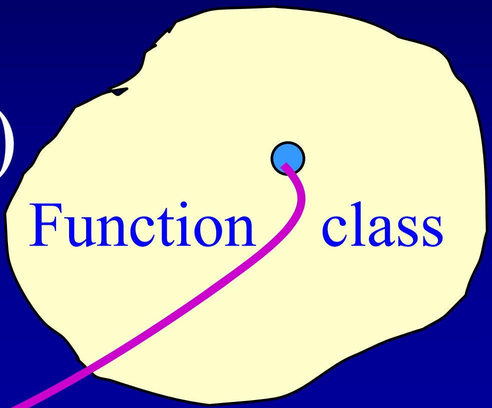
- Popular Methods:
  - **Bagging**: average over trees, each trained using a *random* sub-set drawn from training set
  - **Random Forest**: bagging with *randomized* trees
  - **AdaBoost**: average over trees, each trained with a *different re-weighting* of training set

Jeromme Friedman & Bogdan Popescu

Ensemble Learning

$$F(x) = a_0 + \sum_{m=1}^{M} a_m f_m(x, p_m)$$

$$f(x, p_m) \in \{ f(x, p) \}_{p \in \mathrm{P}}$$



Build, incrementally, an ensemble of *base classifiers* $f(x, p_m)$, choosing each from some function class $\{ f(x, p) \}$ by minimizing some *loss function L*

Jeromme Friedman & Bogdan Popescu

Ensemble Learning – II

$$y = -1 \text{ (background)}$$
$$ = +1 \text{ (signal)}$$
$$L = \text{loss function}$$

$Q_0(x) = 0$

for $m = 1$ to M

{

choose training sample T_m

$$p_m = \arg\min_p \sum_{i \in T_m} L(y_i, Q_{m-1}(x_i) + f(x_i, p))$$

$$Q_m(x) = Q_{m-1}(x) + \nu \cdot f(x, p_m)$$

$$\nu \in [0, 1]$$

}

$$\text{ensemble} = \{ f(x, p_m) \}, \quad m = 1 \ldots M$$

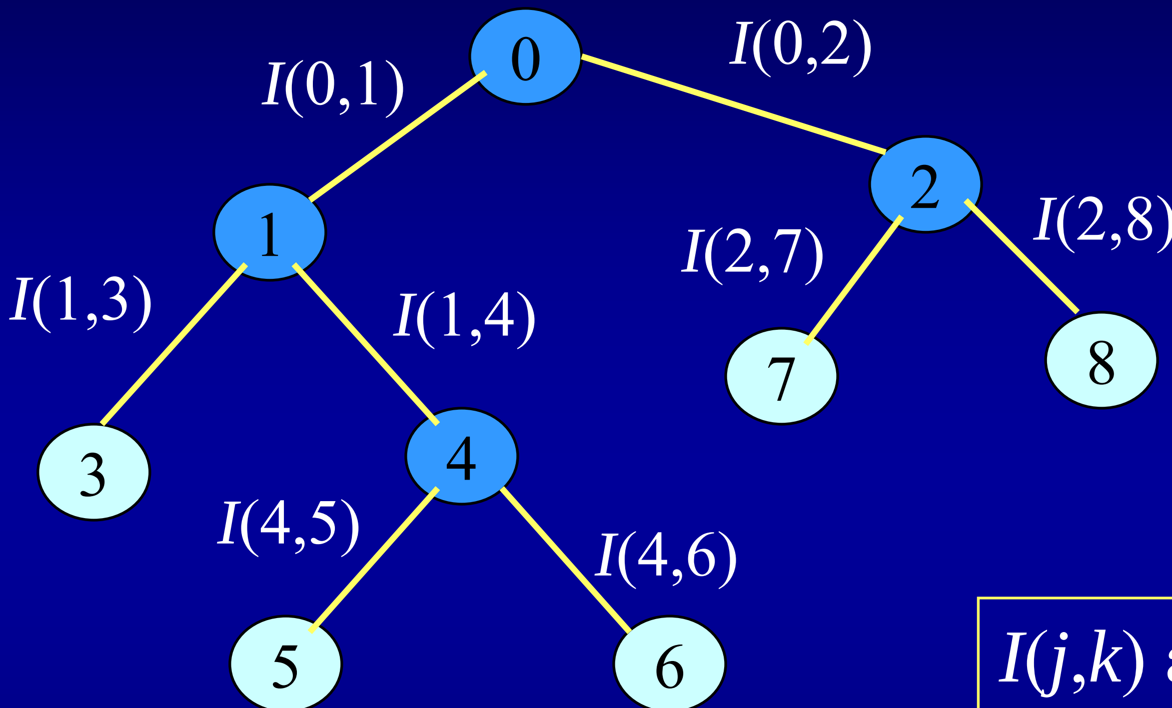
A New Ensemble Method – RuleFit

- Basic Idea (*Friedman & Popescu*)
  - Create an ensemble of trees (a forest!)
  - Create a rule $r(x)$ from each leaf (terminal node) of each tree
  - Final classifier is

$$F(x) = a_0 + \sum_{m=1}^{M} a_m r_m(x)$$

where a_m, $m = 0 \ldots M$ are found by the best fit of $F(x)$ to the target y.

RuleFit – II



$$r_1(x) = I(0,1) \cdot I(1,3)$$
$$r_2(x) = I(0,1) \cdot I(1,4) \cdot I(4,5)$$
$$r_3(x) = I(0,1) \cdot I(1,4) \cdot I(4,6)$$
$$\vdots \qquad\qquad \vdots$$

$I(j,k)$ are cuts, e.g.:

$$I(0,1) = H_T < 200$$
$$I(0,2) = H_T \geq 200$$

RuleFit – III

- Rule Importance
  - Write $$F(x) = a_0 + \sum_{m=1}^{M} a_m \, \sigma_m \left(\frac{r_m(x)}{\sigma_m} \right)$$

    where $\sigma_m = \sqrt{s_m (1 - s_m)}$

  - s_m $= (1/N) \sum_i r_m(x_i)$ is the *support* of the rule
  - I_m $= |a_m| \, \sigma_m$ is the *rule importance*

RuleFit – IV

- Input Variable Importance

$$J(x_j) = \sum_{x_j \in r_m} I_m / n_m$$

where I_m is the importance of the m^th rule containing variable x_j and n_m is the number of variables defining that rule.

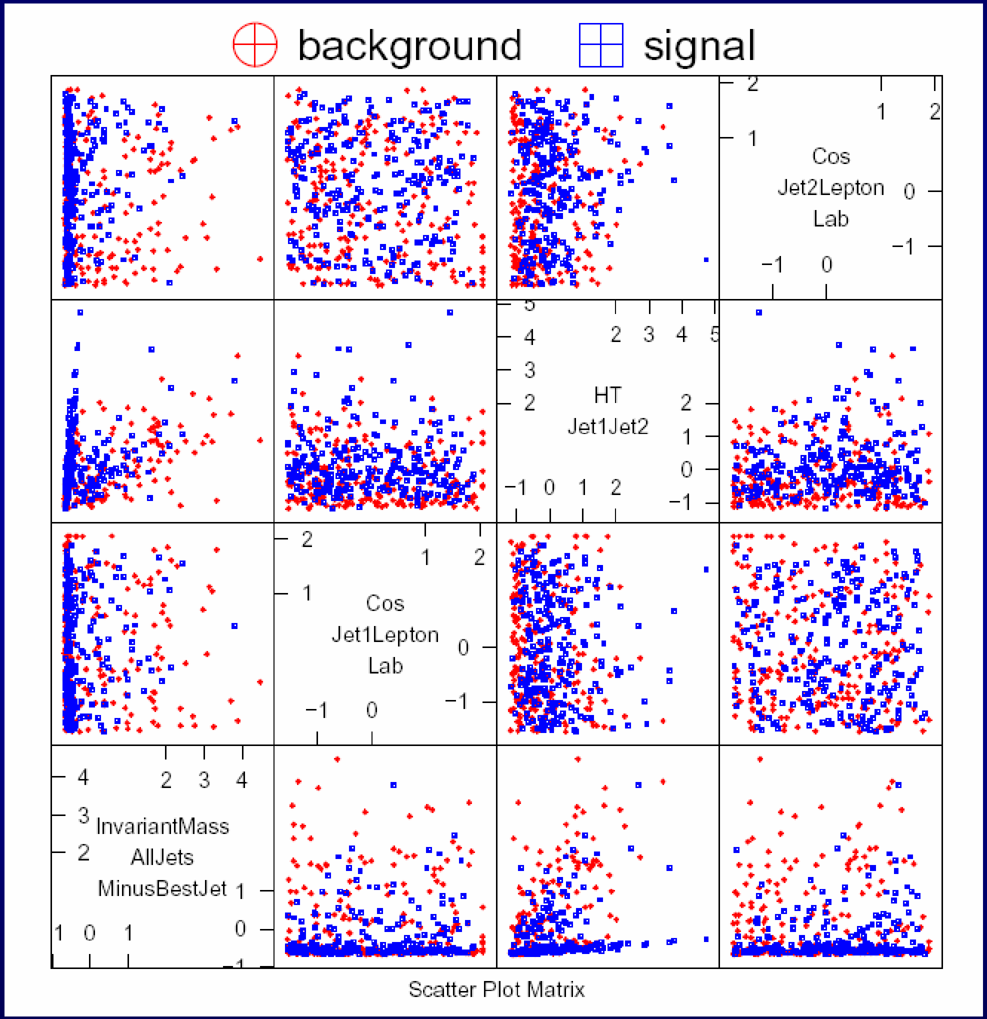
- RuleFit Site
  - http://www-stat.stanford.edu/~jhf/R-RuleFit.html

R

- What is R?
  - A *free* general purpose data analysis language that provides
    - an interpreter
    - excellent graphical tools
    - hundreds of data analysis tools
    - vector-based data manipulation

      e.g., if x is a vector, then $y = \sin(x)$ applies $\sin(\cdot)$ to each element of x.
    - many standard data input formats, including, of course, text!

R – Scatter Plot Matrix (splom-plot)



R – II

- Modules
  - neural networks
  - decision trees
  - fitting
  - bootstrapping
  - clustering
  - spatial models
  - linear models
  - Markov-chain
  - genetic algorithm

    :

- Modules
  - Over 700 modules, each comprising many functions
- Why R?
  - It is the standard language used by professional statisticians. Consequently, new statistical methods, such as RuleFit, are typically written in R

R – Example

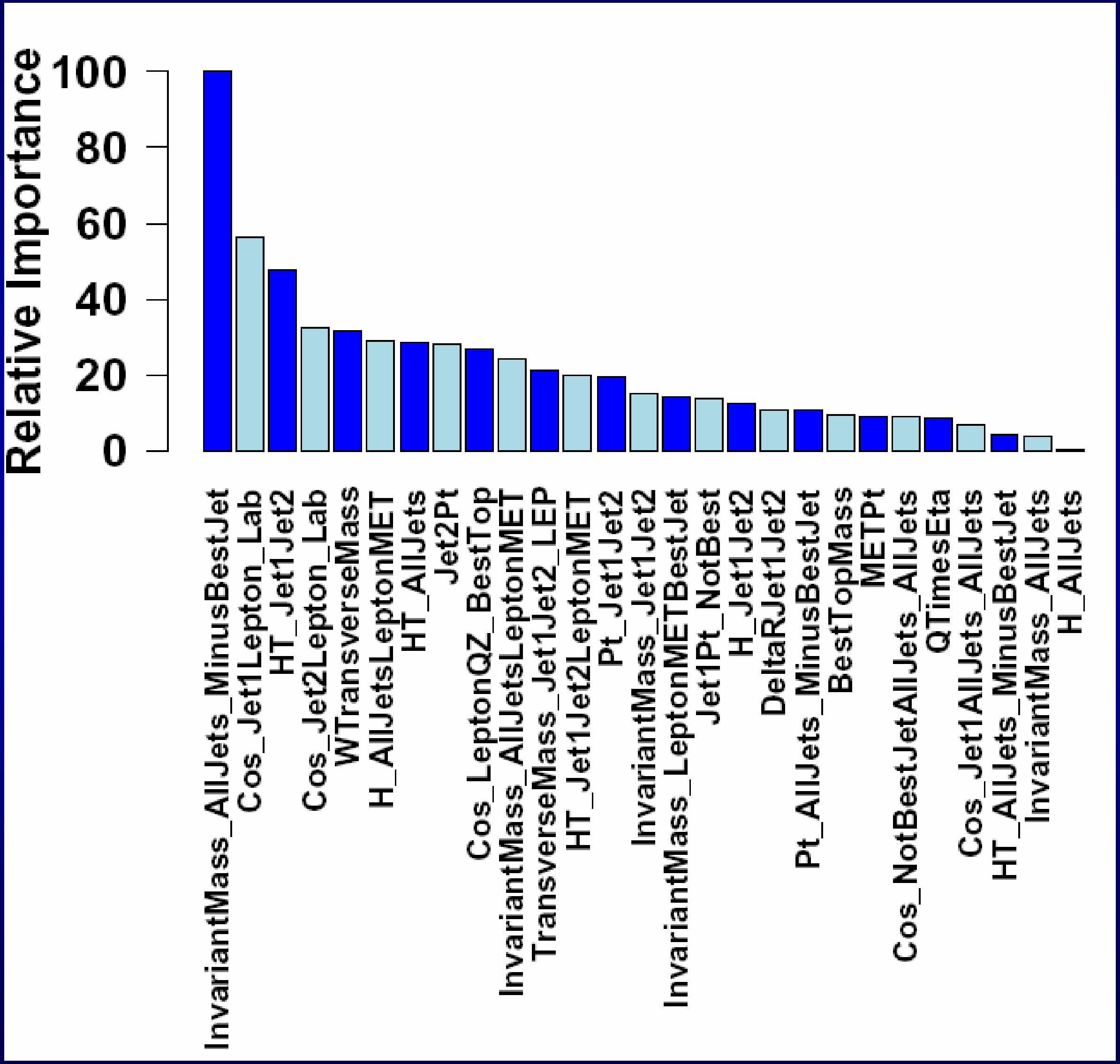
- Data Set (μ, EqOneTag, Ipanema Summer2005)
  - TRAIN sample tb vs QCD+ttbar+Wbb.
    Use ~ 4000 signal + ~ 4000 background events.
- Inputs
  - 27 variables (Shabnam's list)
- RuleFit
  - Use default settings

A Bit of R

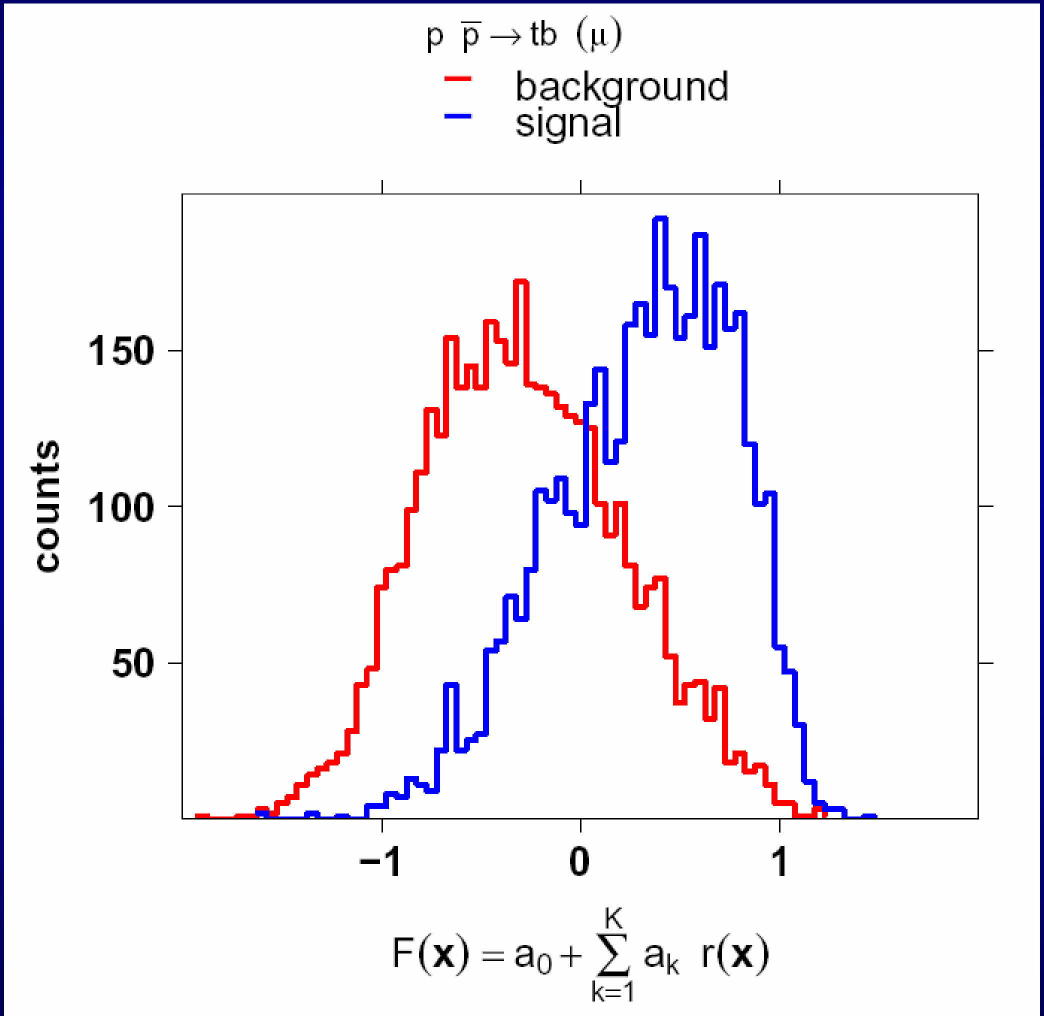
```
#----------------- Initialize RuleFit
ROWS        ← 1:8000;     VARS ← 1:27                    (1)
platform    ← “linux”                                    (2)
rfhome      ← “.”                                        (3)
source( paste(rfhome, “rulefit.r”, sep = “/”) )          (4)
library(akima, lib.loc = rfhome)                         (5)

#----------------- Run RuleFit
d  ← read.table(“mu_tb.dat”); vars ← names(d)[VARS]      (6)
x  ← d[ROWS, vars]                                       (7)
y  ← sapply( d[ROWS, “Target”], function(x){ 2*x-1 } )   (8)
model ← rulefit(x, y, rfmode = “class”)                  (9)
```

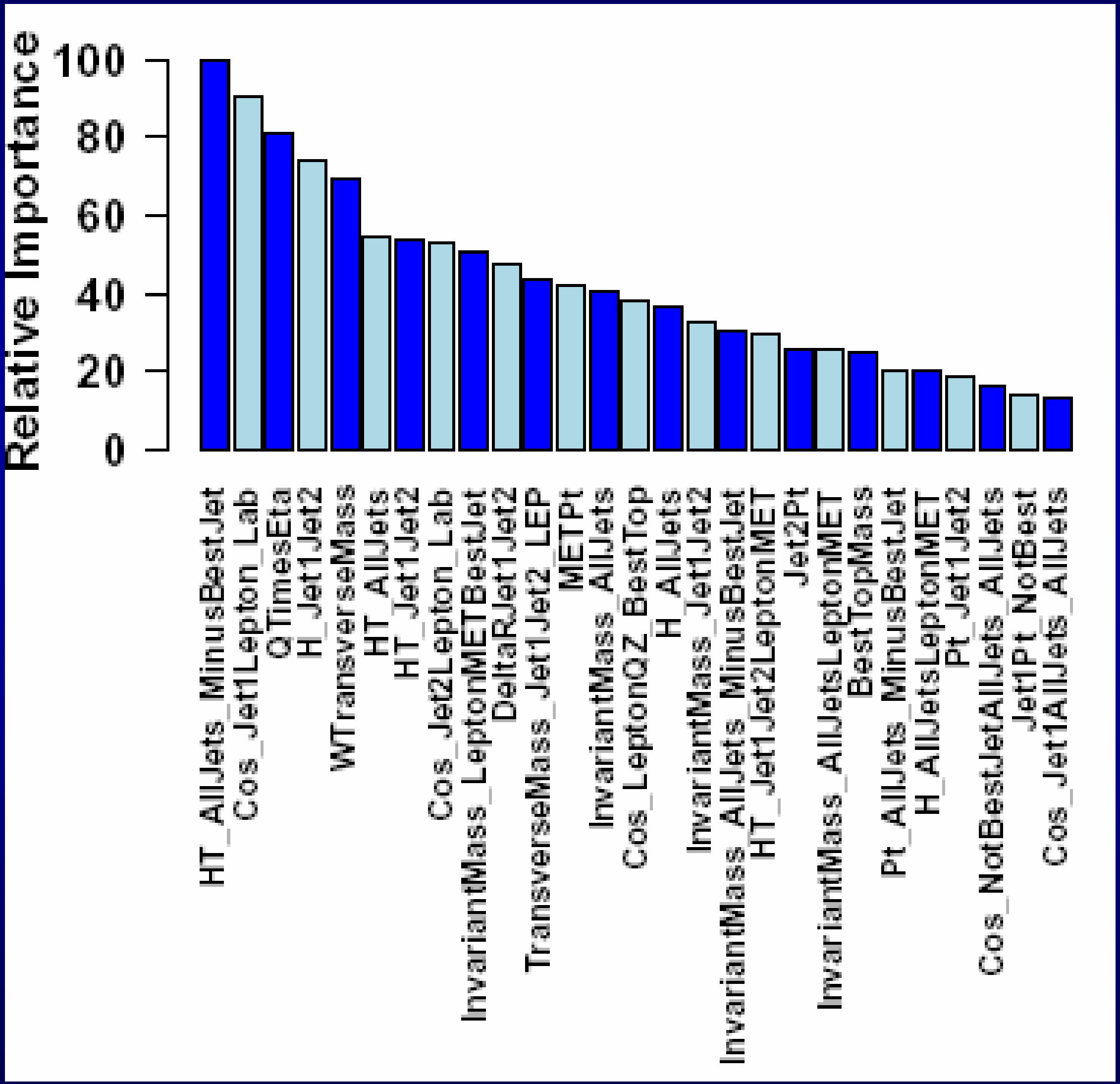
RuleFit – Variable Importance



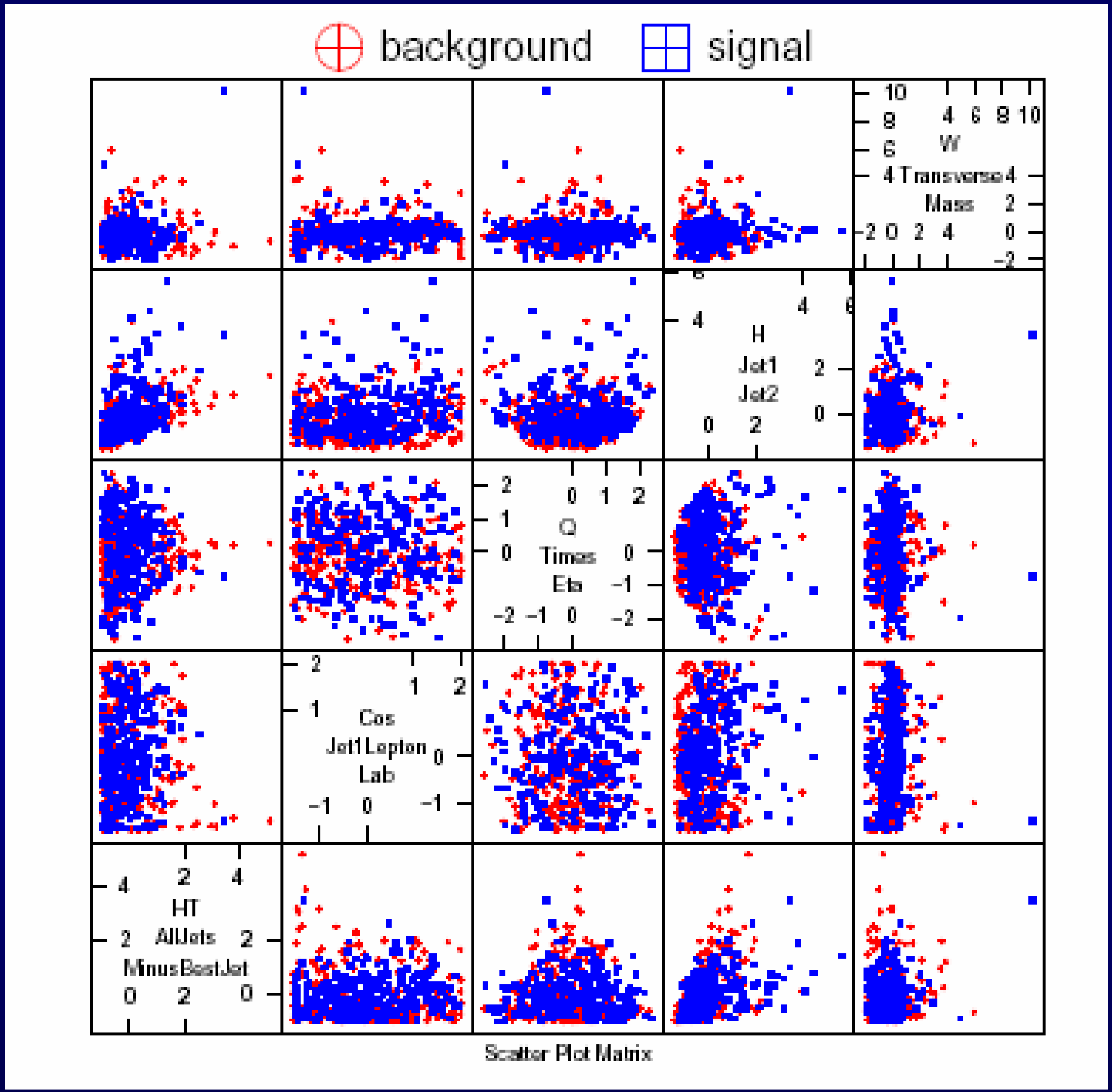
RuleFit – Test



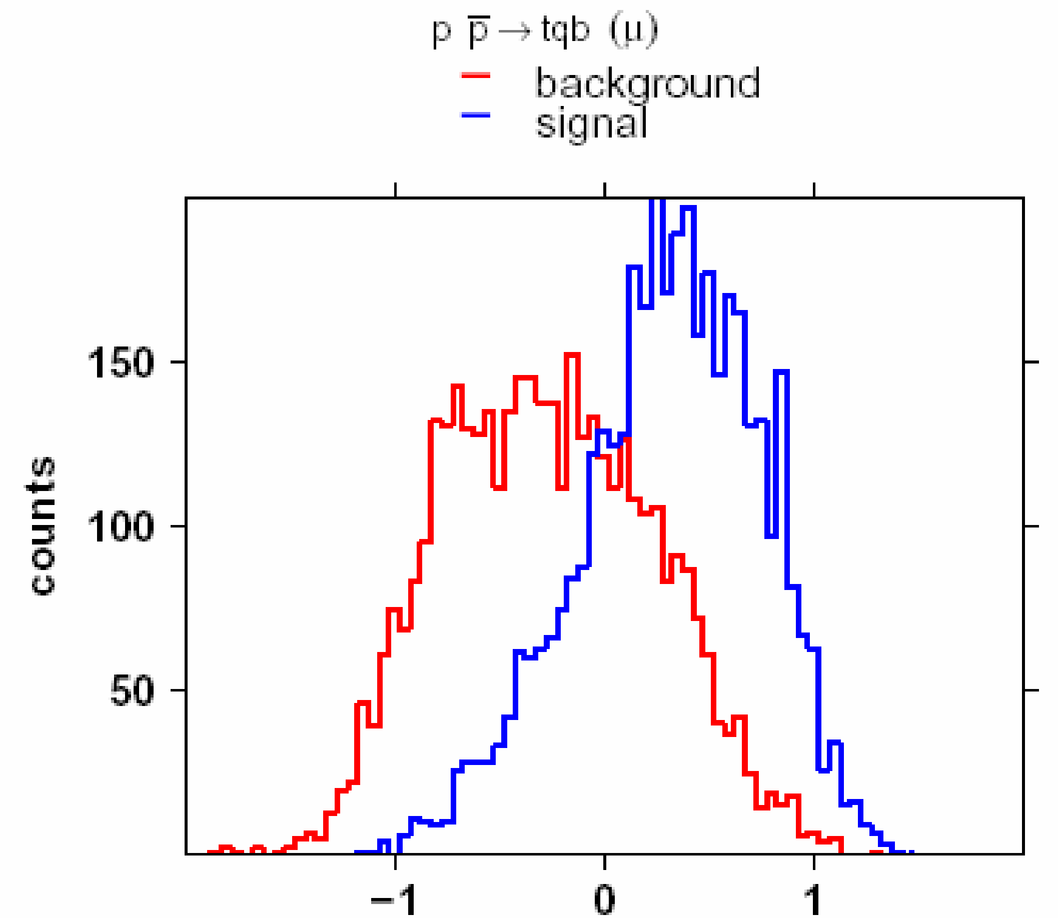
tqb – Variable Importance



tqb – Splom Plot



tqb – Test



$$F(\mathbf{x}) = a_0 + \sum_{k=1}^{K} a_k \; r(\mathbf{x})$$

Conclusions

- PHYSTAT05
  - Acceptance priors should go to zero at zero acceptance!
  - Nonparametric Bayes using KDE may be useful.
  - Variable importance algorithm may be useful.
  - R could be useful for exploration of p17 data.
  - Excellent conference, typical English weather!