Multivariate Discriminants I

Harrison B. Prosper
Florida State University

School Of Statistics

Insitut Pluridisciplinaire Hubert Curien, Strasbourg
30 June 2008 – 04 July 2008

SOS '08

Outline

- Introduction
- Computing Multivariate Discriminants
- Grid Searches
- Quadratic & Linear Discriminants
- Summary

SOS '08

Introduction

SOS '08

Optimal Discrimination

Examples where optimal discrimination, or **classification**, could be useful:

- good/bad run
- normal/bad calorimeter cell
- real/fake lepton
- real/fake jet
- real/fake photon
- heavy/light-jet
- isolated/non-isolated lepton
- signal/background
- etc…

SOS '08

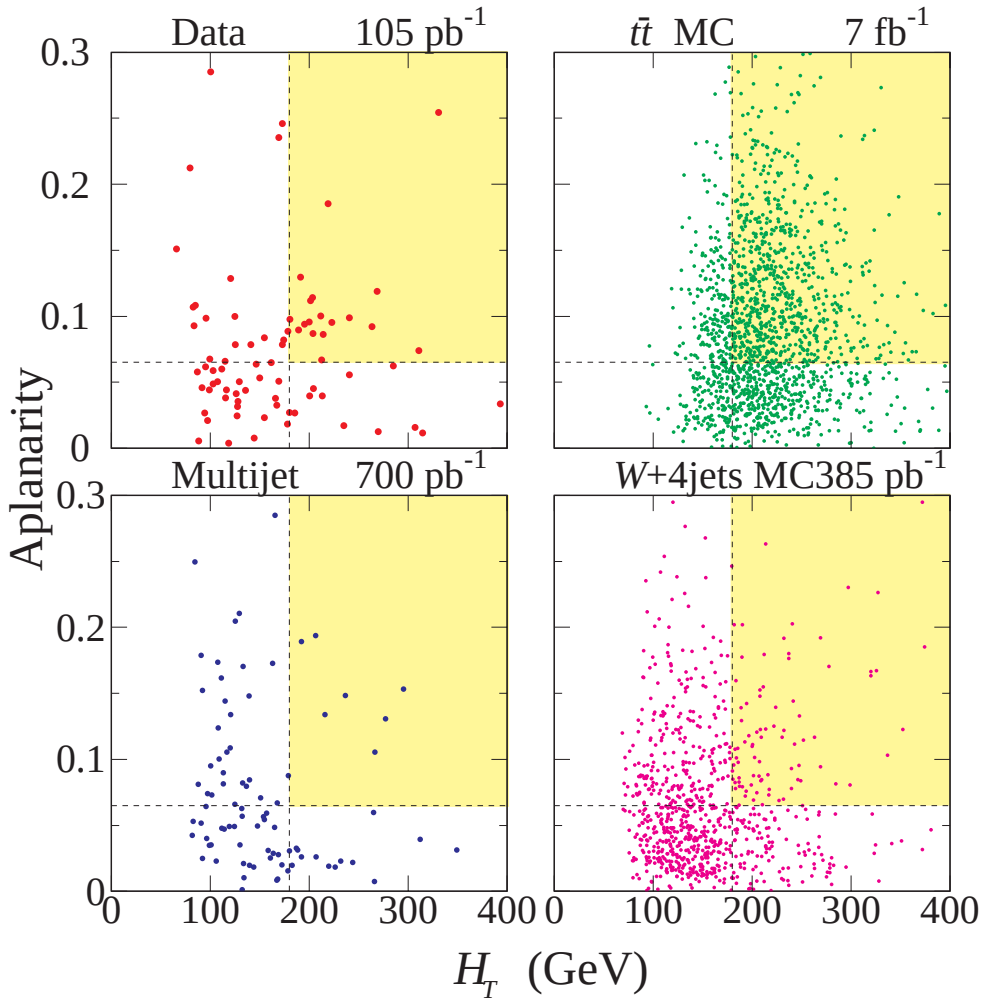
Optimal Discrimination

Note, however, that interesting data are usually multivariate:

$$x = (x_1, x_2, \ldots, x_n)$$

Example:
DØ data, 1995,
top discovery

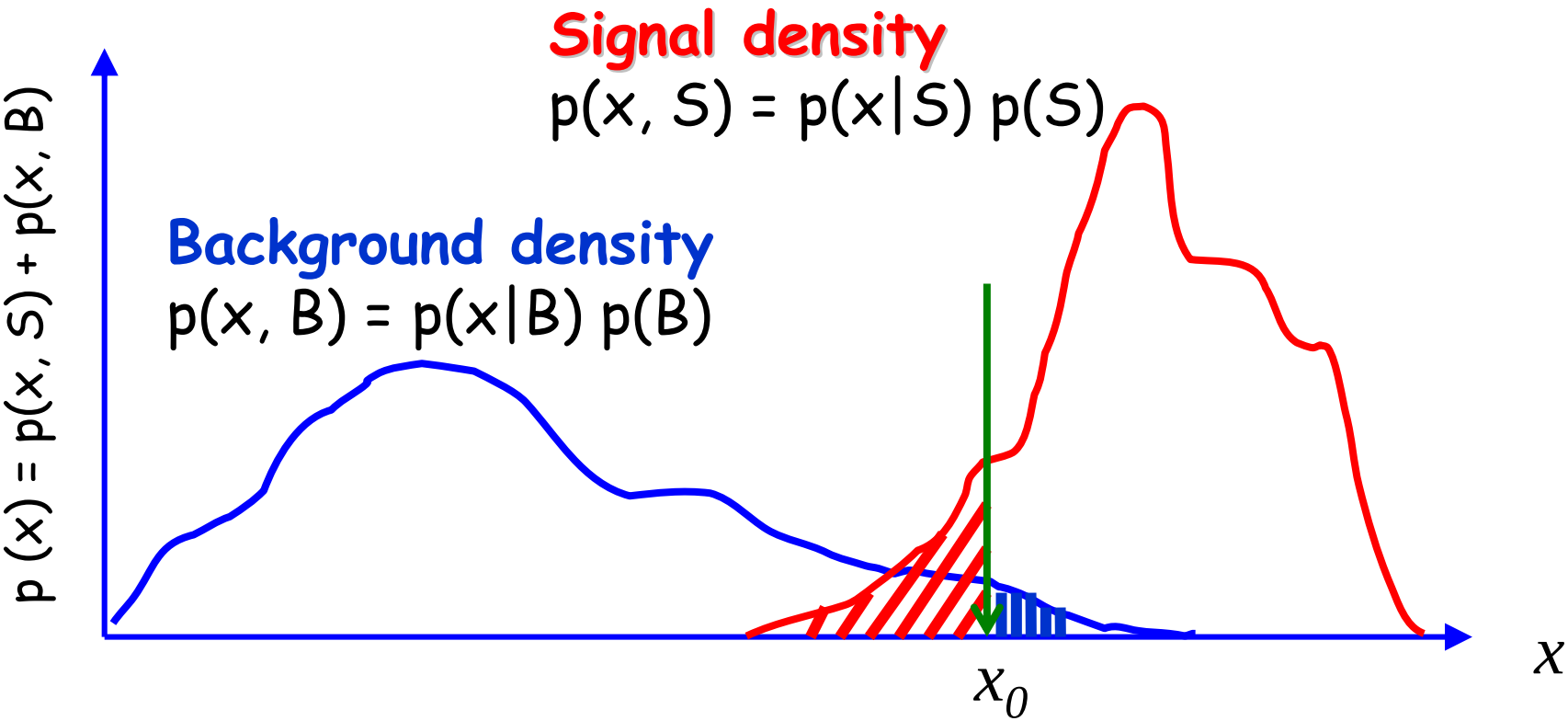
$$p\bar{p} \rightarrow t\bar{t} \rightarrow l + jets$$



SOS '08

Optimal Discrimination

For simplicity, consider **event** classification in 1-dimension



Definition of **optimal**: minimum misclassification cost

Optimal Discrimination

The cost of misclassification is given by

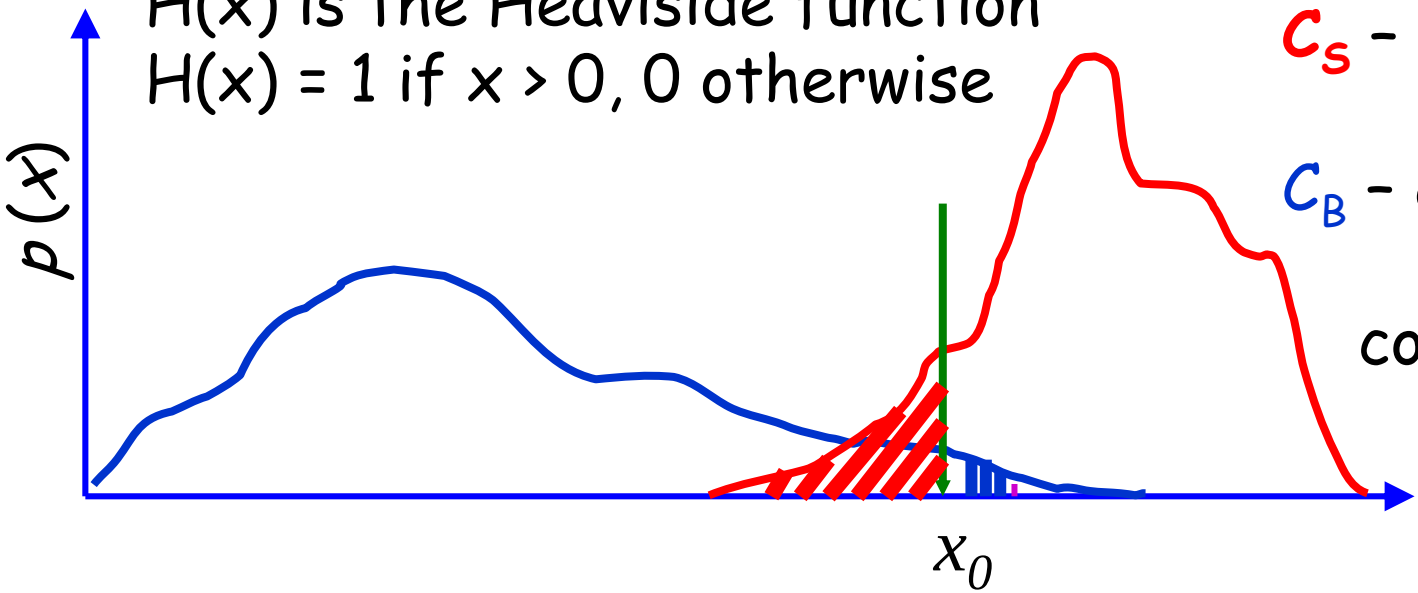
$$C = C_S \int H(x_0 - x)\, p(x, S)\, dx$$

Signal loss

$$+ C_B \int H(x - x_0)\, p(x, B)\, dx$$

Background contamination

H(x) is the Heaviside function
H(x) = 1 if x > 0, 0 otherwise



C_S – cost of signal loss

C_B – cost of background contamination

SOS '08

Optimal Discrimination

Minimizing the cost

$$C(x_0) = C_S \int H(x_0 - x)\, p(x, S)\, dx + C_B \int H(x - x_0)\, p(x, B)\, dx$$

with respect to the **boundary** x_0

$$0 = C_S \int \delta(x_0 - x)\, p(x, S)\, dx - C_B \int \delta(x - x_0)\, p(x, B)\, dx$$
$$= C_S\, p(x_0, S) - C_B\, p(x_0, B)$$

gives the **Bayes discriminant**

$$BD = \frac{C_B}{C_S} = \frac{p(x_0 \mid S)\, p(S)}{p(x_0 \mid B)\, p(B)}$$

SOS '08

Optimal Discrimination

The same form holds when x is multi-dimensional

$$BD = B\frac{p(S)}{p(B)} \quad \text{where} \quad B = \frac{p(x\,|\,S)}{p(x\,|\,B)}$$

is the **Bayes factor**, which is identical to the **likelihood ratio** when there are no unknown parameters

The Bayes discriminant is so called because it is related to **Bayes theorem**

$$p(S\,|\,x) = \frac{BD}{1+BD}$$

A classifier that achieves the minimum cost, and fewest mistakes, is said to have reached the **Bayes limit**

SOS '08

Optimal Discrimination

Note: to achieve optimal discrimination, it is ***not*** necessary to use the correct prior signal to background ratio **k** = p(S) / p(B). Suppose, you chose **k** = 1.

In this case, the discriminant D(x) is given by

$$D(x) = s(x) / [s(x) + b(x)]$$

where s(x) = p(x|S) and b(x) = p(x|B). Then, because of the one-to-one relationship,

$$p(S \mid x) = D(x)\,p(S) / [D(x)\,p(S) + (1 - D(x))\,p(B)]$$

a **cut** on D(x) implies a corresponding cut on p(S|x)

SOS '08

Optimal Signal Extraction

In fact, it is not necessary to apply a cut to extract the signal: the signal can be determined using **event-by-event weighting***. Write the data density as

$$d(x) = \varepsilon\, s(x) + (1-\varepsilon)\, b(x), \quad \varepsilon = \text{signal fraction}$$

Event weighting is simply multiplication by a weight function w(x)

$$w(x)d(x) = \varepsilon\, w(x)s(x) + (1-\varepsilon)\, w(x)b(x)$$

*R. Barlow, "Event Classification Using Weighting Methods," J. Comp. Phys. **72**, 202 (1987)

SOS '08

Optimal Signal Extraction

Compute the expectations

$$\bar{w} = \int dx\, w(x) d(x) \qquad \text{observed data}$$

$$\bar{w}_s = \int dx\, w(x) s(x) \qquad \text{signal}$$

$$\bar{w}_b = \int dx\, w(x) b(x) \qquad \text{background}$$

Then the **signal fraction**, and the **variance** of its estimator are given by

$$\varepsilon = (\bar{w} - \bar{w}_b)/(\bar{w}_s - \bar{w}_b)$$

$$\mathrm{Var}(\hat{\varepsilon}) = \frac{1}{n} \int dx \left(\frac{w - \bar{w}_b}{\bar{w}_s - \bar{w}_b} \right)^2 d(x)$$

where **n** is the number of events

SOS '08

Optimal Signal Extraction

Roger Barlow showed that the signal size is determined with the **smallest variance** when events are weighted with any linear function of

$$w(x) = p(S \mid x) = \frac{s(x)}{s(s) + b(x)/k}$$

Since we do not know **k**, we start with a reasonable guess for it (e.g., a prediction), derive an updated value for **k** through event weighting and repeat the procedure until the value of **k** converges

SOS '08

Computing Multivariate Discriminants

SOS '08

Learning from Examples

Given N examples $(x,y)_1$, $(x,y)_2$,…$(x,y)_N$ the task is to construct an approximation to the discriminant D(x). x are called **feature variables** and y are the **class labels**

There are two general approaches to the problem:

Machine Learning

Teach a "machine" to learn f(x) by feeding it examples, that is, **training data D**.

Bayesian Learning

Infer f(x) given the likelihood for the training data D and a prior on the space of functions f(x).

SOS '08

Machine Learning

Given N examples $(x,y)_1, (x,y)_2, \ldots (x,y)_N$ we specify:

- A function class $F_w = \{ f(x, w) \}$
- A risk function $R(f) = \int L(y, f)\, p(x, y)\, dx\, dy$
- A constraint $C(w)$ on the parameters w

The **loss function** L(y, f) measures how much we lose if we make a poor choice from the function class.

In practice, we minimize the **empirical risk** plus the **constraint**

$$E(w) = \frac{1}{N} \sum_{i=1}^{N} L(y_i, f(x_i, w)) + C(w)$$

SOS '08

Bayesian Learning

Ingredients:

Pr(D|f) the likelihood (of training data)

Pr(f) the prior (over functions)

Then compute:

$$\Pr(f|D) = \Pr(D|f)\,\Pr(f)/\Pr(D)$$

In practice, we work with some function class

$$F_w = \{ f(x, w) \}$$

and make inferences on the parameters:

$$\Pr(w|D) = \Pr(D|w)\,\Pr(w)/\Pr(D)$$

SOS '08

Bayesian Learning

Write

$D = \mathbf{x}, \mathbf{y}$

$\mathbf{x} = \{x_1, \ldots x_N\}$, $\mathbf{y} = \{y_1, \ldots y_N\}$

of N training examples

$$P(AB) = P(A|B)\,P(B) = P(B|A)\,P(A)$$

$$P(A|B) = P(B|A)\,P(A)/P(B)$$

Then Bayes' theorem becomes

$$p(w|\mathbf{x}, \mathbf{y}) = p(\mathbf{x}, \mathbf{y}|w)\,p(w) / p(\mathbf{x}, \mathbf{y})$$

$$= p(\mathbf{y}|\mathbf{x}, w)\,p(\mathbf{x}|w)\,p(w) / p(\mathbf{y}|\mathbf{x})\,p(\mathbf{x})$$

SOS '08

Bayesian Learning

The data **x** do not depend on **w** since they are generated independently of the particular function class we are using. *Consequently*, $p(\mathbf{x}|\mathbf{w}) = p(\mathbf{x})$ and, therefore,

$$p(\mathbf{w}|\mathbf{x}, \mathbf{y}) = p(\mathbf{y}|\mathbf{x}, \mathbf{w})\, p(\mathbf{x}|\mathbf{w})\, p(\mathbf{w}) \,/\, p(\mathbf{y}|\mathbf{x})\, p(\mathbf{x})$$
$$= p(\mathbf{y}|\mathbf{x}, \mathbf{w})\, p(\mathbf{w}) \,/\, p(\mathbf{y}|\mathbf{x})$$

The likelihood for the training data is $p(\mathbf{y}|\mathbf{x}, \mathbf{w})$, the probability density of the class labels, or **targets** **y**, *given* data **x**, evaluated for a given training sample

We now consider two possible forms for $p(\mathbf{y}|\mathbf{x}, \mathbf{w})$

SOS '08

Bayesian Learning

Likelihood for **regression** (with $y_i \in R$)

$$p(\mathbf{y}|\mathbf{x}, \mathbf{w}) = \Pi_i \sqrt{(2\pi/\tau)} \exp[-\tfrac{1}{2}\tau (y_i - f(x_i, \mathbf{w}))^2] \qquad (1)$$

Likelihood for **classification** (with $y_i \in \{0, 1\}$)

$$p(\mathbf{y}|\mathbf{x}, \mathbf{w}) = \Pi_i f(x_i, \mathbf{w})^y [1 - f(x_i, \mathbf{w})]^{1-y} \qquad (2)$$

Note: If events are weighted, then each term must be raised to the power of the associated event weight w_E

SOS '08

Bayesian Learning

Consider the logarithm of the "regression" likelihood

$$E(\mathbf{w}) = (1/N) \sum [y_i - f(x_i, \mathbf{w})]^2 + [2/(N\tau)] \ln p(\mathbf{w})$$

empirical risk constraint

where we have re-scaled $E \rightarrow (2/N\tau) E$.

Now take the limit $N \rightarrow \infty$. In that limit, the contribution of the prior goes to zero and we obtain

$$E(\mathbf{w}) = \int dx \int dy\, [y - f(x, \mathbf{w})]^2\, p(x, y)$$
$$= \int dx\, p(x) \int dy\, [y - f(x, \mathbf{w})]^2\, p(y|x)$$

SOS '08

Bayesian Learning

IF the class F_w, to which $f(x, w)$ belongs, is large enough then it will contain a function $f(x, w^*)$ which minimizes $E(w)$. This minimum occurs at

$$f(x, w^*) = \int y\, p(y|x)\, dy$$

that is, $f(x, w^*)$ is the **conditional expectation** of the target y.

Exercise: Prove this

SOS '08

Bayesian Learning

Suppose we use the "regression" likelihood with only two values for y, 0 or 1.

In this case, $p(y|x) = \delta(y-1)\, p(1|x) + \delta(y-0)\, p(0|x)$, so

$$f(x, w^*) = \int y\, p(y|x)\, dy$$
$$= p(1|x)$$
$$= p(x|1)\, p(1) \,/\, [p(x|1)\, p(1) + p(x|0)\, p(0)]$$

which is just the Bayes' discriminant, disguised as Bayes' theorem!

SOS '08

Verification of Discriminants

To verify, in full generality, that q(x) is a satisfactory approximation of the discriminant D(x) = s(x)/[s(x) + b(x)] is a very challenging problem

However, some simple and useful heuristics exist, such as one suggested by event weighting

SOS '08

Verification of D(x)

Weight **equal** numbers of signal and background events (*using events not from the training sample*) by q(x), that is, compute

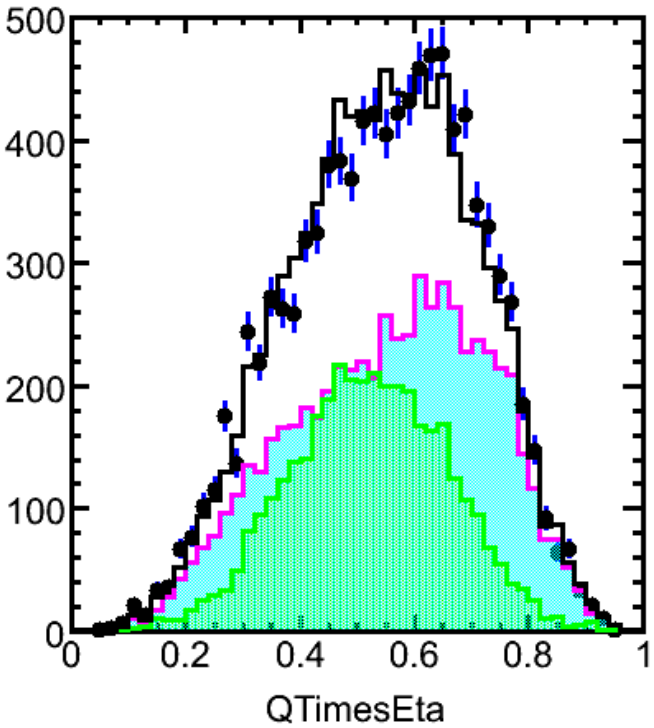
$$s_q(x) = s(x)\, q(x) \text{ and } b_q(x) = b(x)\, q(x)$$

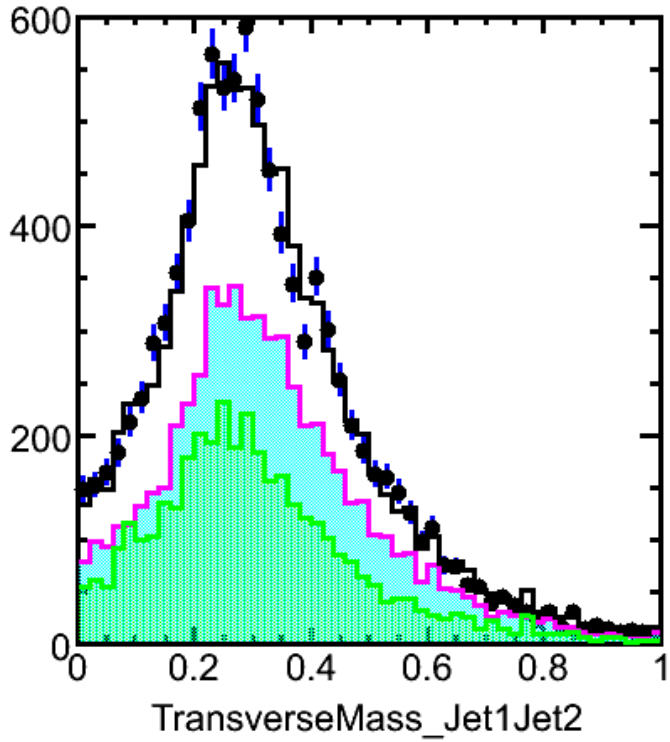
Then, if $q(x) \approx D(x)$, the **sum** of the weighted distributions, $s_q(x)$ and $b_q(x)$, should recover the signal density s(x)

$$s_q(x) + b_q(x) \approx s(x)$$

SOS '08

Verification of D(x)





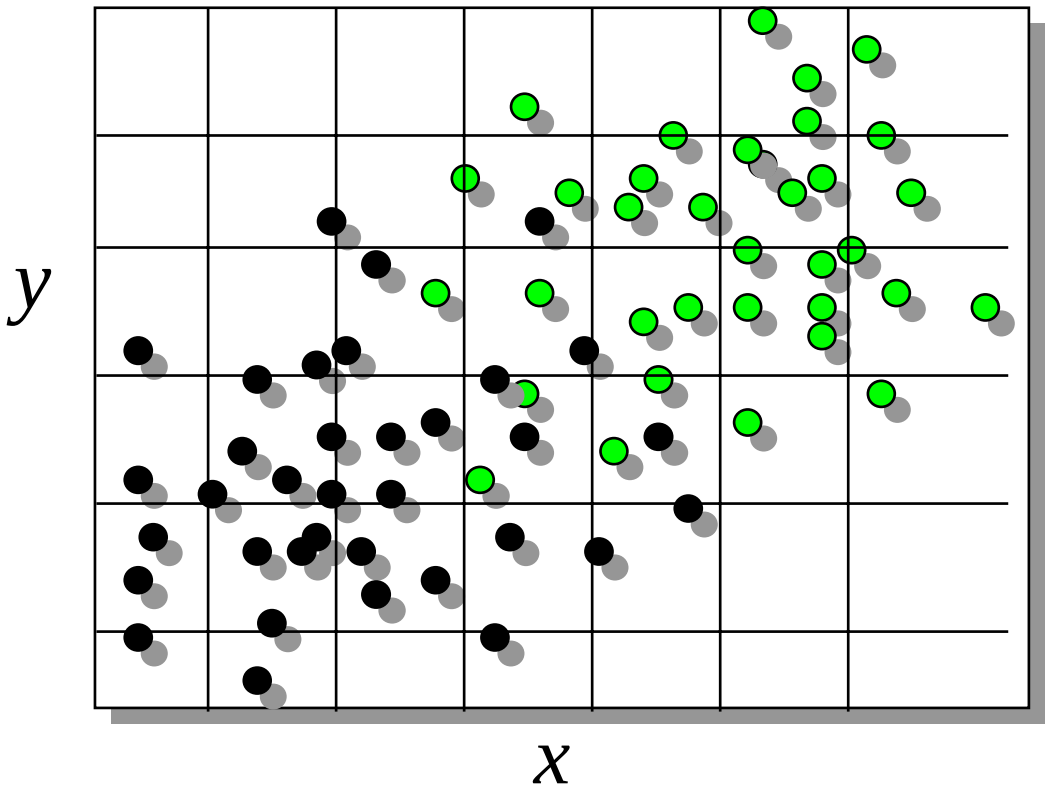
Two of the variables used in the DØ search for single top quarks, illustrating the verification of D(x). Shown are $s_q(x)$, $b_q(x)$, $d_q(x) = s_q + b_q$ and s(x) (the dots).

SOS '08

Grid Searches

SOS '08

Grid Search



Apply cuts at each grid point

$$x > x_i$$

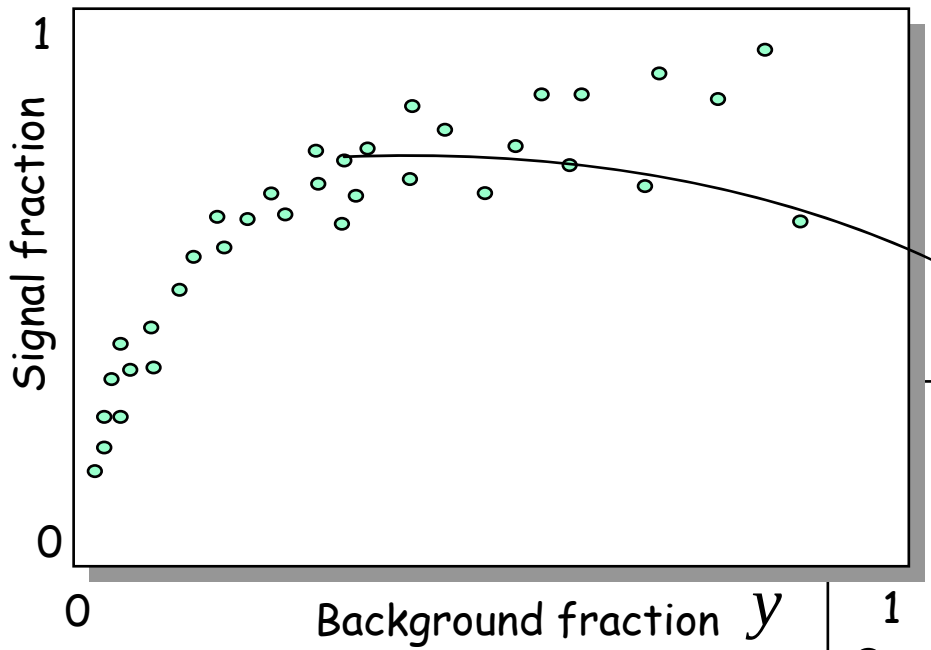
$$y > y_i$$

We refer to (x_i, y_i) as a ***cut-point***

Suffers from the curse of dimensionality ~ $M^{\dim(d)}$

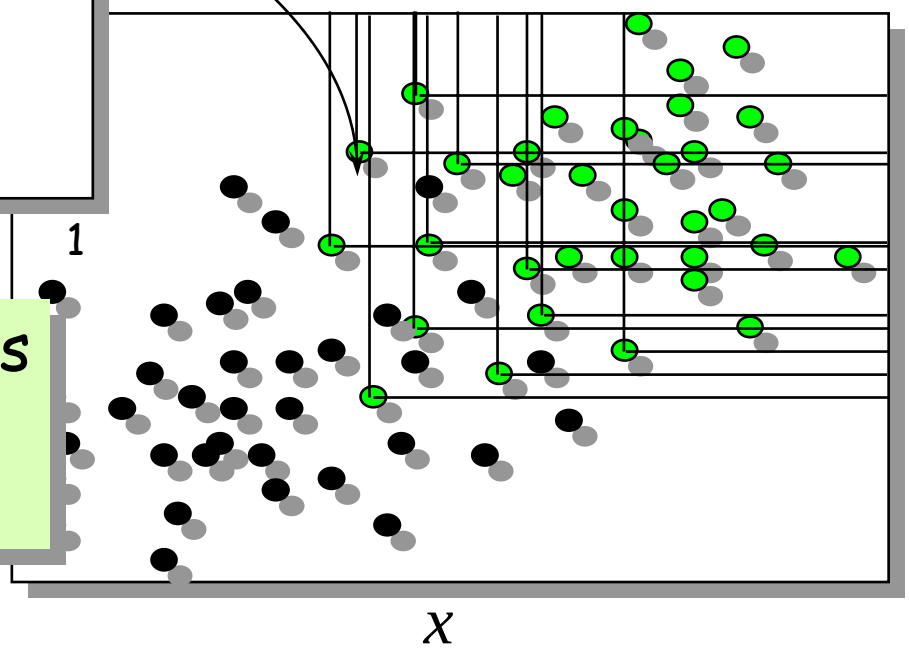
Random Grid Search

Take each point of the signal class as a cut-point



$x > x_i$

$y > y_i$



N_{tot} = # events before cuts
N_{cut} = # events after cuts
Fraction = N_{cut}/N_{tot}

H.B.P. et al., Proceedings, CHEP 1995

SOS '08

Random Grid Search - Example

CMS mSUGRA study

The Focus Point Region

m_0	= 3280 GeV,
$m_{1/2}$	= 300 GeV,
A_0	= 0,
$\tan\beta$	= 10,
sign(μ)	= +1
m_{top}	= 175 GeV

$$pp \to \tilde{g}\ \tilde{g}$$

$$pp \to \chi^{\pm}\ \chi^{0}$$

$$pp \to \chi^{+}\ \chi^{-}$$

$$pp \to \chi^{0}\ \chi^{0}$$

Event selection

- $ME_T > 40$ GeV
- $N_j \geq 5$ jets, with $E_T > 30$ GeV
- $|\eta_{j1}|, |\eta_{j2}| < 2.5$

Random Grid Search – Example

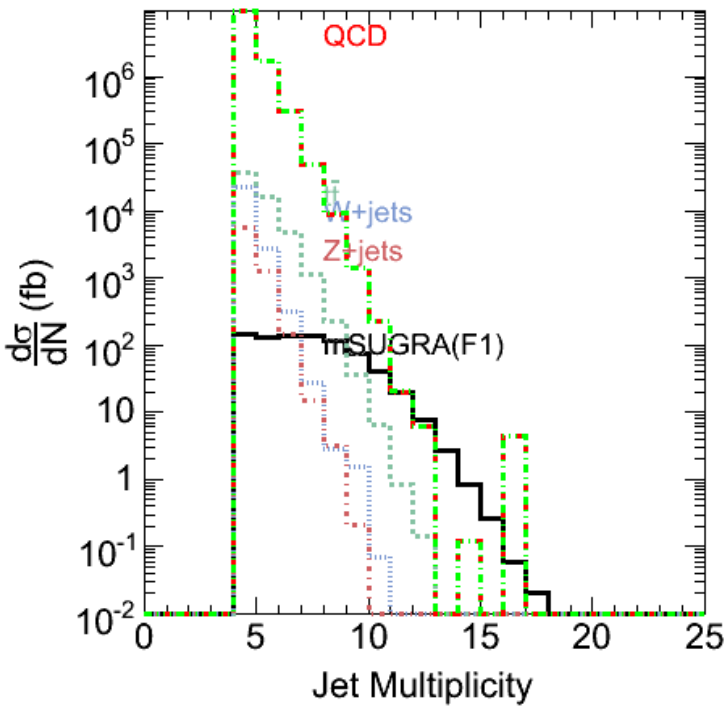
Reaction	B. F. (%)
$pp \to \tilde{g}\ \tilde{g}$	89.0
$pp \to \chi^{\pm}\ \chi^{0}$	6.3
$pp \to \chi^{+}\ \chi^{-}$	2.6
$pp \to \chi^{0}\ \chi^{0}$	0.5

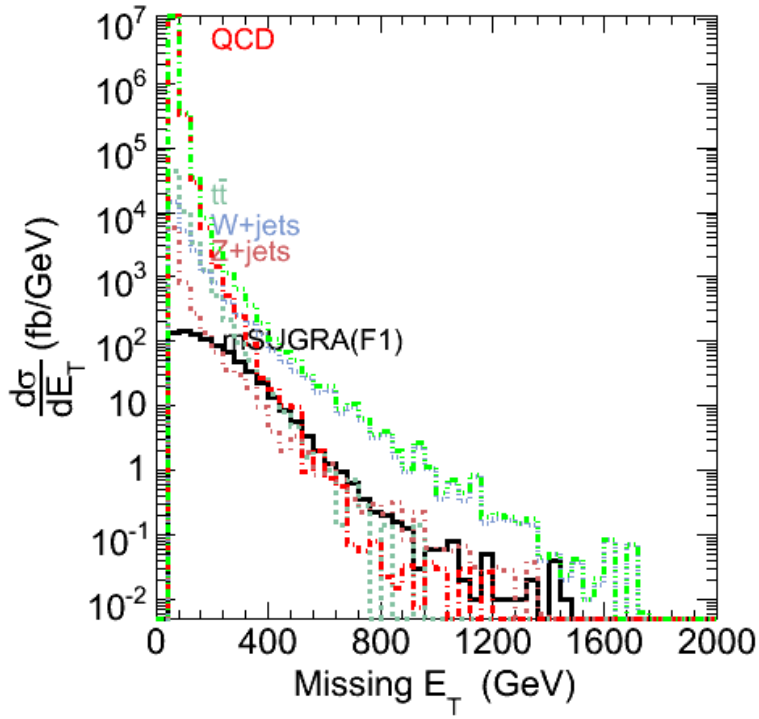
Event Source	σ (fb)
QCD	2.0×10^6
ttbar	2.2×10^4
W+jets	3.1×10^3
Z+jets	1.5×10^3
mSUGRA	6.7×10^2

Signal : Noise ~ **1 : 3000**

SOS '08

Random Grid Search - Example

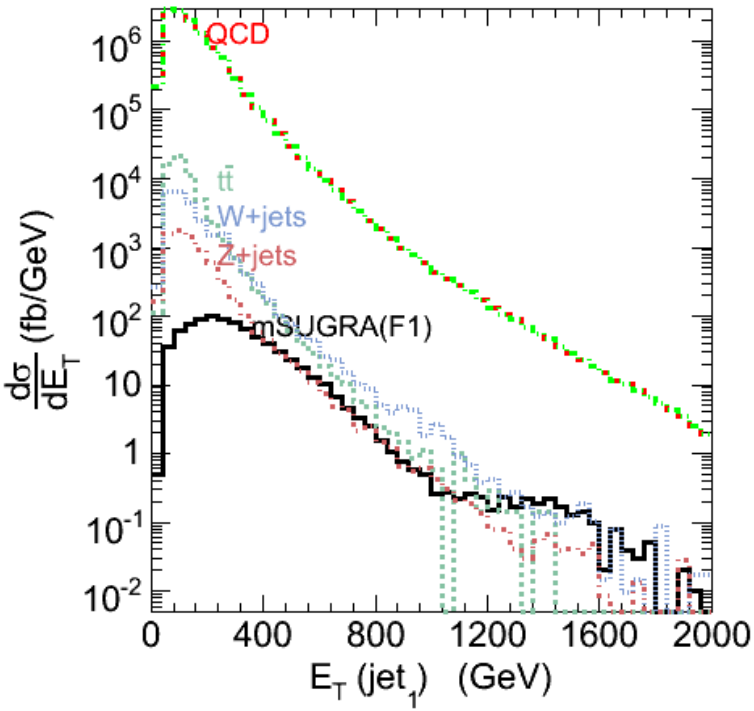


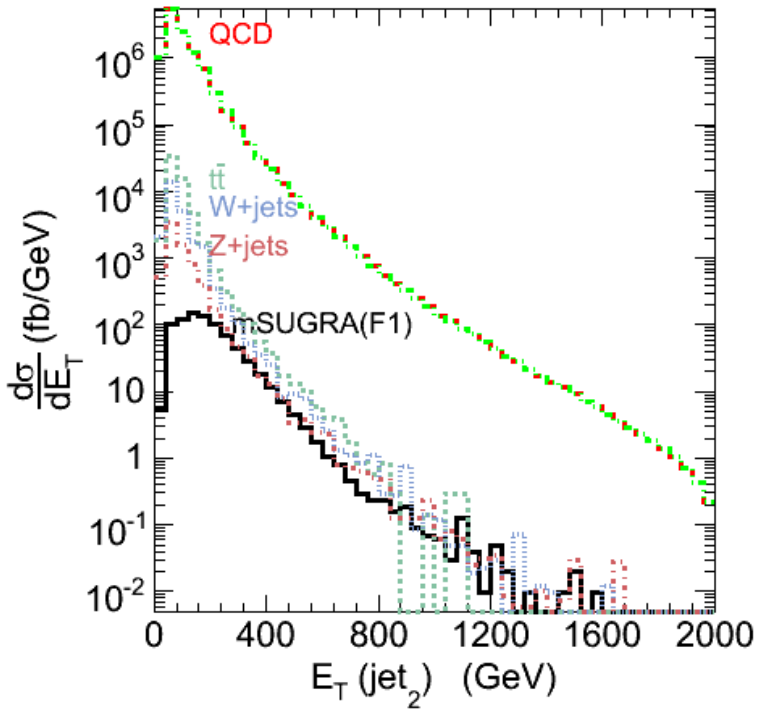


Note: Spectra for ≥ 4 jets

SOS '08

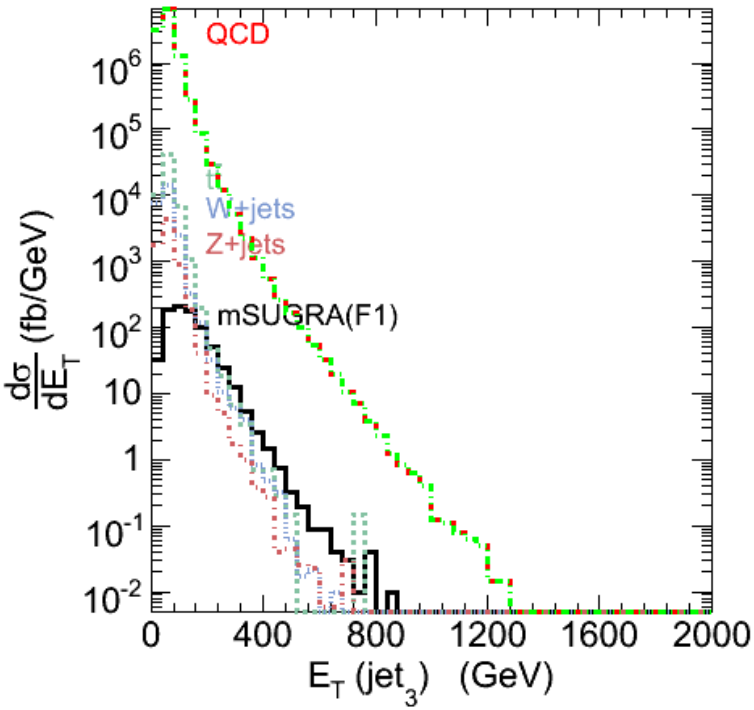
Random Grid Search - Example

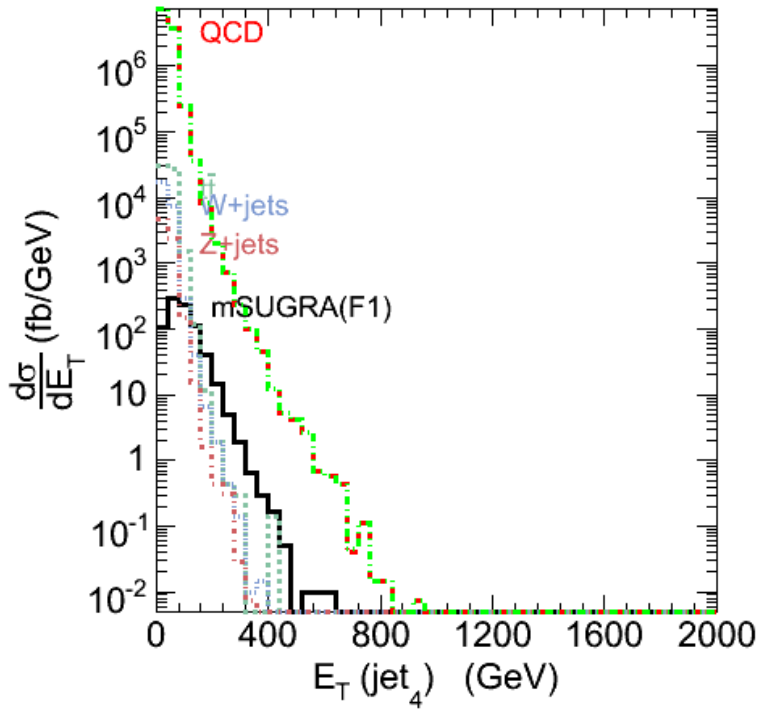




SOS '08

Random Grid Search - Example



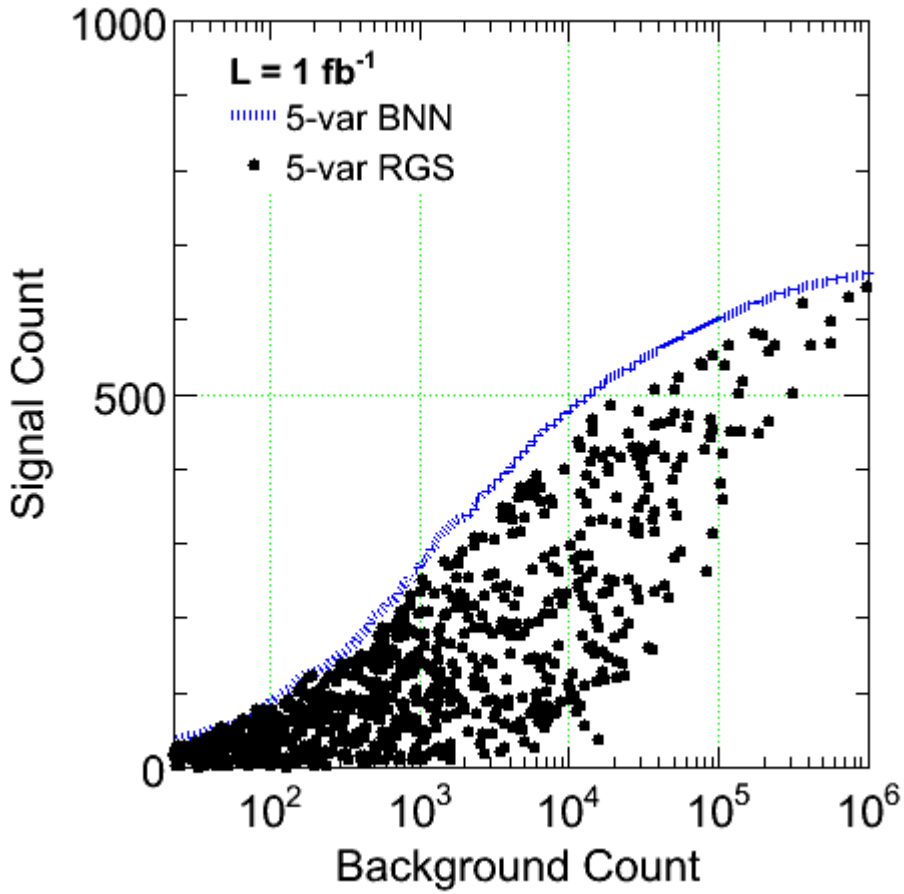


Random Grid Search – Example

Random grid search over 5 variables

ME_T, P_{Tj}, j=1,…,4

assuming 1 fb^{-1}



SOS '08

Quadratic & Linear Discriminants

SOS '08

Quadratic Discriminants

Suppose that each density s(x) and b(x) is a multivariate Gaussian

$$\text{Gaussian}(x \mid \mu, \Sigma) = \frac{\exp[-(x-\mu)^T \Sigma^{-1} (x-\mu)/2]}{(2\pi)^{d/2} \mid \Sigma \mid^{1/2}}$$

where μ is the vector of **means** and Σ is the **covariance matrix**. In this case, can write an explicit expression for the Bayes factor

$$B(x) = s(x)/b(x)$$

SOS '08

Quadratic Discriminants

It is usually more convenient to consider the logarithm of the Bayes factor,

$$\lambda(x) = \ln B(x),$$

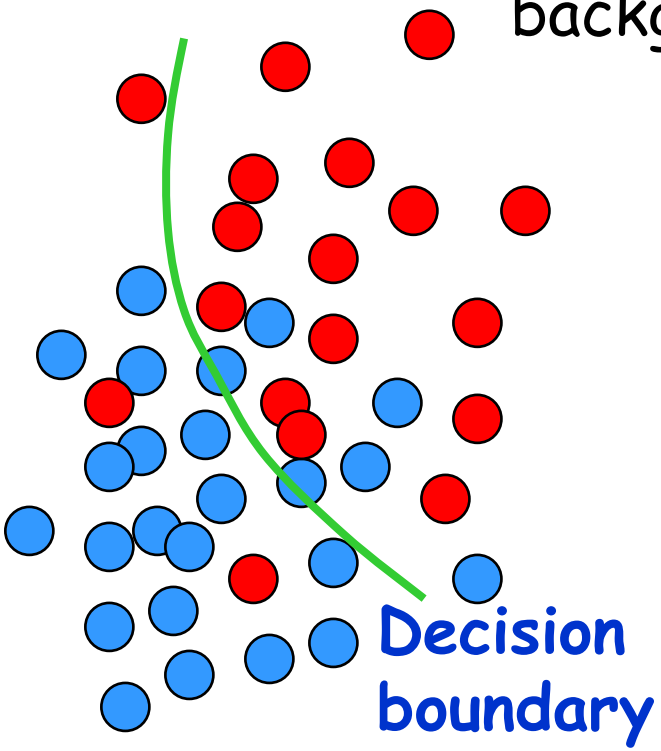
which, after eliminating non-essential constants, can be written as

$$\lambda(x) = (x-\mu_B)^T \Sigma_B^{-1} (x-\mu_B) - (x-\mu_S)^T \Sigma_S^{-1} (x-\mu_S)$$

SOS '08

Quadratic Discriminant

A fixed value of $\lambda(x)$ defines a quadratic hyper-surface that partitions the d-dimensional **feature space** {x} into signal-rich and background-rich regions.



SOS '08

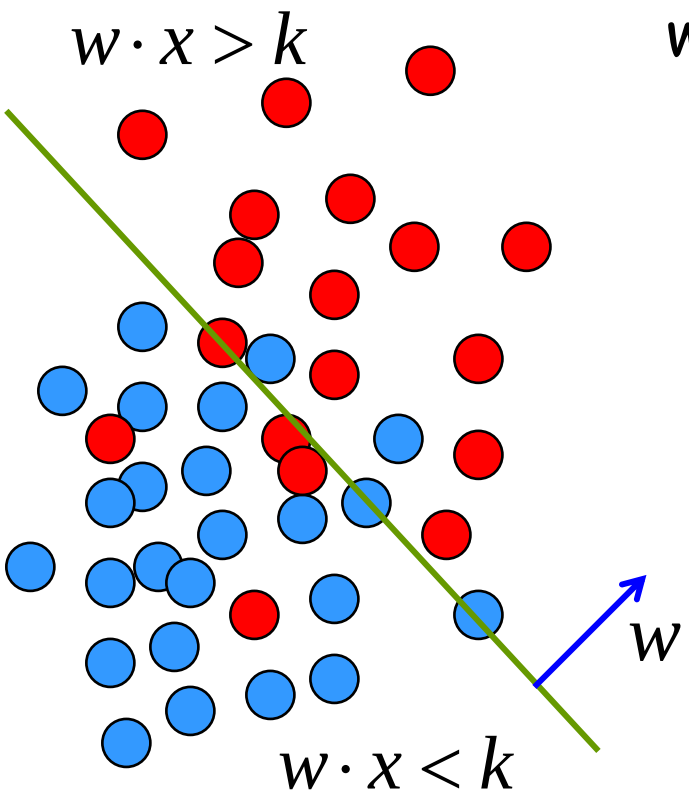
Linear Discriminant

If, in the quadratic function $\lambda(x)$, we use the same covariance matrix for each class of events

e.g., $\Sigma = \Sigma_S + \Sigma_B$

we arrive at

Fisher's Discriminant



$$\lambda(x) = w \cdot x$$

where w is a vector given by

$$w \propto \Sigma^{-1}(\mu_S - \mu_B)$$

SOS '08

Summary

1. If the goal is to classify objects with the fewest mistakes, it is sufficient to apply a threshold, that is, a cut, to the discriminant

$$D(x) = \frac{s(x)}{s(x) + b(x)}$$

2. If the goal is to extract the signal strength with minimum variance, it is sufficient to weight events using the associated weight function

$$w(x) = \frac{D(x)}{D(s) + [1 - D(x)]/k}$$