# Bayesian LAGO for Statistical Detection Problems

**Wanhua Su**
**Department of Statistics and Actuarial Science**
**University of Waterloo**

**Joint work with Dr. Hugh Chipman and Dr. Mu Zhu**

# Outline

- The statistical detection problem.
- LAGO—An adaptive detection method.
- BLAGO—Bayesian LAGO.
- Conclusions and discussions.

# The Statistical Detection Problem

- Set-up:$\{\mathbf{x}_i, y_i\}_{i=1}^n$, $\mathbf{x}_i \in \mathbb{R}^d$ is a vector of predictors, $y_i \in \{0, 1\}$ is class label. (binary classification?).
- Class frequencies: **extremely unbalanced**.
- Objective: Detect those class-1 observations rather than assign class label to new data.

# Performance Measure

- Misclassification rate (MR) is not proper.

| True label | classification result 0 | classification result 1 |
|---|---|---|
| 0 | a (true negative) | b (false positive) |
| 1 | c (false negative) | d (true positive) |

And the misclassification rate (MR) is given by

$$\mathrm{MR} = \frac{b+c}{a+b+c+d}$$

- For detection problem, Average Precision (AP) is more suitable.

$$\mathrm{AP} = \frac{1}{n_1} \sum_{i=1}^{n} \frac{y_{(i)} h(i)}{i}$$

# Examples

- Google search. Prefer searching algorithms that return the relevant documents as early as possible. (1,0)=(relevant, irrelevant).

- Drug discovery. Prefer a models that detect those active drug agent for a certain disease as soon as possible. (1,0)=(active compound, inactive compound).

# LAGO: An Adaptive Method

- LAGO—Locally Adjusted GO estimate.

- By Bayes' rule:

$$p(Y=1|\mathbf{z}) = \frac{p(\mathbf{z}|Y=1)p(Y=1)}{p(\mathbf{z}|Y=1)p(Y=1)+p(\mathbf{z}|Y=0)p(Y=0)} = \frac{af(\mathbf{z})}{af(\mathbf{z})+1}, \tag{1}$$

where $a = \frac{p(Y=1)}{p(Y=0)}$ and $f(\mathbf{z}) = \frac{p(\mathbf{z}|Y=1)}{p(\mathbf{z}|Y=0)}$.

# Step 1 in Estimating $f(\mathbf{z})$

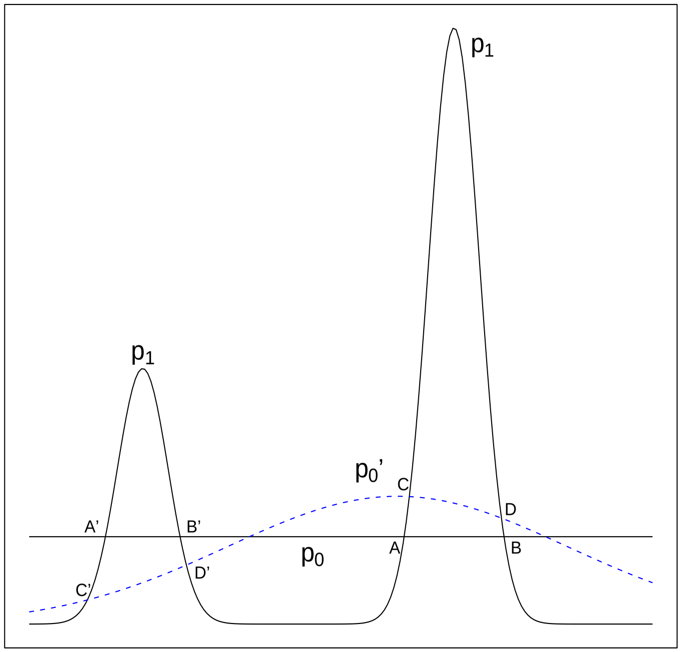
- Estimate $p(\mathbf{z}|Y=1)$ by kernel density estimate

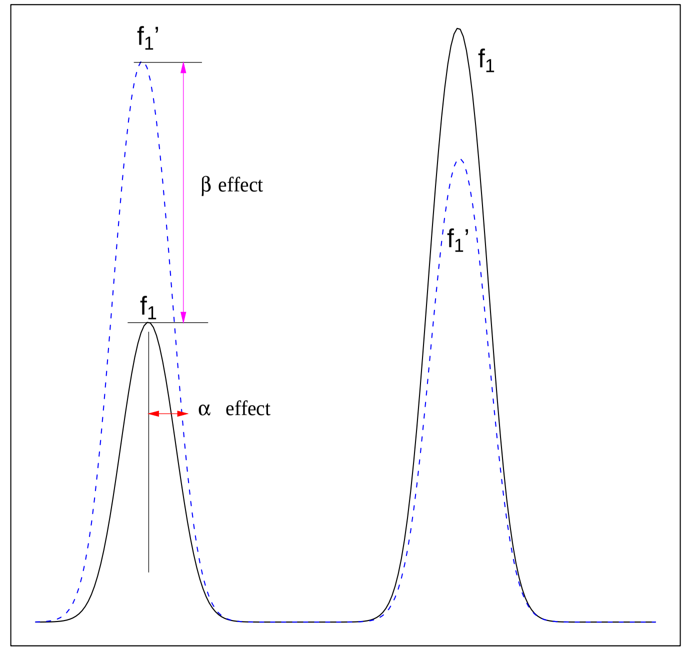
$$p(\mathbf{z}|Y=1) = \frac{1}{N_1}\sum_{i=1}^{N_1}\prod_{j=1}^{d}\frac{1}{r_{ij}}g\left(\frac{z_j - x_{ij}}{r_{ij}}\right),$$

  where $r_{ij} = \frac{1}{K}\sum_{\mathbf{w}\in N(\mathbf{x}_i,K)}|w_{ij} - x_{ij}|$ is the average distance in the $j$-th dimension between a class-1 observation $\mathbf{x}_i$ and its $K$ nearest class-0 neighbors

- If $p(\mathbf{z}|Y=0) = C$, $p(\mathbf{z}|Y=1) \propto f(\mathbf{z})$, and hence $p(\mathbf{z}|Y=1) \propto p(Y=1|\mathbf{z})$.

# Examining the ripple effects




**Note**: Left: The density functions of $p_0$ and $p_1$. Right: The posterior distributions.

# Step 2 in Estimating $f(\mathbf{z})$

- Consider the ripple effects, adjust $p(\mathbf{z}|Y = 1)$ according to the local density of the class-0 observations, we obtain the LAGO estimate

$$\begin{aligned} \hat{f}(\mathbf{z}) &= \frac{1}{N_1} \sum_{i=1}^{N_1} \prod_{j=1}^{d} \frac{1}{\alpha r_{ij}} r_{ij}^{\beta} g\left(\frac{z_j - x_{ij}}{\alpha r_{ij}}\right) \\ &\propto \frac{1}{N_1} \sum_{i=1}^{N_1} \prod_{j=1}^{d} g\left(\frac{z_j - x_{ij}}{\alpha r_{ij}}\right). \end{aligned} \qquad (2)$$

- LAGO paper (2006): Technometrics, 48(2), 193-205.

# Motivation of BLAGO

- Only provides a point estimate, no uncertainty is embodied.
- LAGO score is within (0,1), but not a probability.
- A more precise estimate.

# Bayesian Methods in Machine Learning

- Bayesian Neural Network (Dr. Radford Neal)
- Bayesian Tree (Dr. Hugh Chipman)
- Bayesian K-nearest-neighbor (Dr. Chris Holmes)

# Likelihood Function

- LAGO score is not probabilistic, change it to probability scale by a logistic transformation

$$\theta_i = p(y_i = 1|\mathbf{z}_i) = \frac{e^{\beta_0+\beta_1 \hat{f}(\mathbf{z}_i)}}{1 + e^{\beta_0+\beta_1 \hat{f}(\mathbf{z}_i)}}, \qquad (3)$$

where $(\mathbf{z}_i, y_i)$ is the $i$th test point and $\beta_1 > 0$.

- The pseudo-likelihood is

$$\mathrm{L}(\boldsymbol{\theta}; \mathbf{Y}) = \prod_{i=1}^{n} \theta_i^{y_i} [1 - \theta_i]^{1-y_i}. \qquad (4)$$

# Prior Distributions

- $K \sim \text{UNIF}(1, 2, \cdots, K_{\text{max}})$.
- $\alpha \sim \text{UNIF}(0, \infty)$.
- $(\beta_0, \beta_1) \sim \text{BVN}(\boldsymbol{\mu}_0, \boldsymbol{\Sigma}_0)$ with $\beta_1 > 0$.

# MCMC Steps

**Step 0.** Set the initial $K_{old}$, $\alpha_{old}$;

**Step 1.** Propose new $K_{new}$ and $\alpha_{new}$ by

$$K_{new} = K_{old} \pm U[0,1,2,3], \;\; \alpha_{new} = \alpha_{old} + N(0, \sigma_\alpha^2).$$

**Step 2.** Generate a random number $\gamma \sim U(0,1)$. Accept the new state $K_{new}, \alpha_{new}$ if $\gamma \leq J$ with accepted probability

$$J = min\left\{\frac{p(K_{new}, \alpha_{new}|\mathbf{Y})}{p(K_{old}, \alpha_{old}|\mathbf{Y})}, 1\right\}. \qquad (5)$$

**Step 3.** Repeat steps 2 and 3 until burn-in.

# Laplace Approximation

The well-known Laplace approximation is given by

$$\begin{aligned} I_n &= \int_a^b b(\boldsymbol{\beta}) exp\{-nr(\boldsymbol{\beta})\} dt \\ &\approx b(\hat{\boldsymbol{\beta}})(\frac{2\pi}{n})^{\frac{p}{2}} (\det\Sigma_{\hat{\boldsymbol{\beta}}})^{\frac{1}{2}} e^{-nr(\hat{\boldsymbol{\beta}})}, \end{aligned} \tag{6}$$

where $p$ is the dimension of $\boldsymbol{\beta}$; and $\hat{\boldsymbol{\beta}}$ is the solution to $\nabla r(\hat{\boldsymbol{\beta}}) = 0$; and $\Sigma_{\hat{\boldsymbol{\beta}}} = [r^{''}(\hat{\boldsymbol{\beta}})]^{-1}$.

# Evaluating $p(K,\alpha|\mathbf{Y})$

$$
\begin{aligned}
p(K,\alpha|\mathbf{Y}) &\propto p(\mathbf{Y}|K,\alpha)\pi(K,\alpha) \\
&\propto \int p(\mathbf{Y}|K,\alpha,\boldsymbol{\beta})\pi(\boldsymbol{\beta}|K,\alpha)d\boldsymbol{\beta} \\
&= \int L(\boldsymbol{\beta})\pi(\boldsymbol{\beta})d\boldsymbol{\beta} \\
&= \int \exp(\mathcal{L}(\boldsymbol{\beta}))d\boldsymbol{\beta} \\
&= \int \exp\{-n(-\frac{1}{n}\mathcal{L}(\boldsymbol{\beta}))\}d\boldsymbol{\beta} \\
&\approx (2\pi)^{p/2}e^{\mathcal{L}(\boldsymbol{\beta}^*|K,\alpha)}|H(\boldsymbol{\beta}^*|K,\alpha)|^{-1/2}. \qquad (7)
\end{aligned}
$$

- $\mathcal{L}(\boldsymbol{\beta}^*)$: Log posterior of $\boldsymbol{\beta}$ evaluated at the posterior mode $\boldsymbol{\beta}^*$
- $H(\boldsymbol{\beta}^*)$: Hessian matrix of the log posterior of $\boldsymbol{\beta}$ evaluated at $\boldsymbol{\beta}^*$.

# Prediction

Suppose

$$\theta(\mathbf{z}_i|\text{train}, k_0, \alpha_0, \boldsymbol{\beta}_0) = \frac{e^{\mathbf{x}_i^T \boldsymbol{\beta}_0}}{1 + e^{\mathbf{x}_i^T \boldsymbol{\beta}_0}}, \tag{8}$$

where $\mathbf{x}_i = [1, \hat{f}(\mathbf{z}_i|\text{train}, k_0, \alpha_0)]^T$. Incorporating uncertainty, the prediction

$$\begin{aligned} \theta(\mathbf{z}_i|\text{train}) &\approx \frac{1}{M}\sum_{j=1}^{M} \theta(\mathbf{z}_i|\text{train}, k^{(j)}, \alpha^{(j)}) \\ &= \frac{1}{M}\sum_{j=1}^{M} \int \theta(\mathbf{z}_i|\text{train}, k^{(j)}, \alpha^{(j)}, \boldsymbol{\beta}) p(\boldsymbol{\beta}|\text{train}, k^{(j)}, \alpha^{(j)}) d\boldsymbol{\beta}, \end{aligned}$$

where $k^{(j)}$ and $\alpha^{(j)}$ are the $j$th ($j = 1, \cdots, M$) sample in the converged chain.

# Conclusions and Discussions

- Strength: BLAGO provides a probabilistic prediction and uncertainty of the prediction.
- Challenges:
  - Choice of the prior distributions.
  - The proposal functions.
  - Performance measure.

# Reparameterization

- Consider the ripple effect, $\alpha$, the kernel function (d=1) is

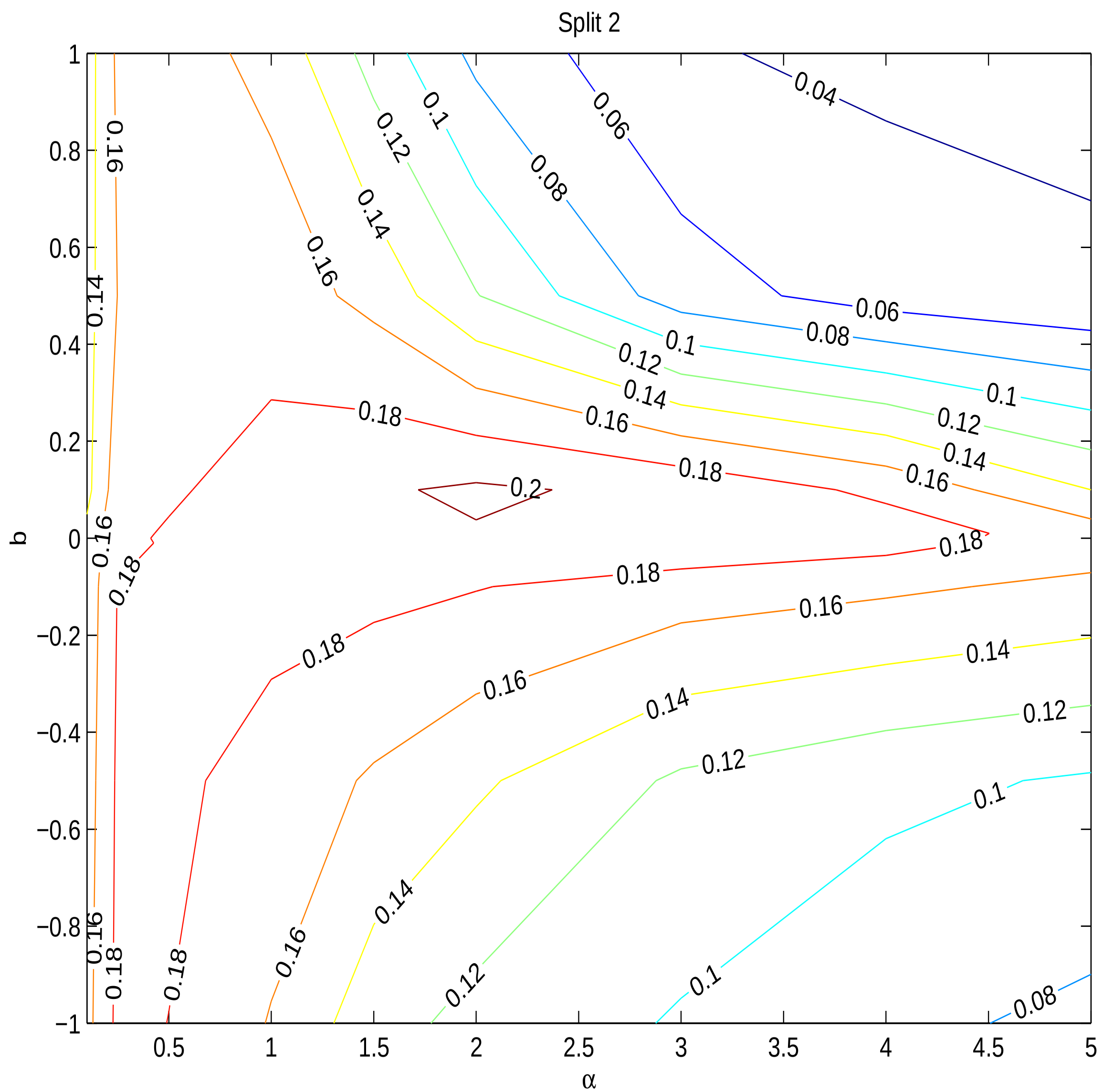
$$\frac{1}{\alpha r_i} g\left(\frac{z - x_i}{\alpha r_i}\right)$$

- Incorporate the other ripple effect, $\beta$, the score function is

$$\hat{p}(\mathbf{z}) = \frac{1}{n_1} \sum_{i=1}^{n_1} \left(\frac{1}{\alpha r_i}\right)(\alpha r_i)^{\beta} g\left(\frac{z - x_i}{\alpha r_i}\right) \propto \frac{1}{n_1} \sum_{i=1}^{n_1} r_i^{\beta-1} g\left(\frac{z - x_i}{\alpha r_i}\right)$$

- An empirical study shows that $b = \beta - 1 \approx 0$.

# Average precision maximized around $b \approx 0$



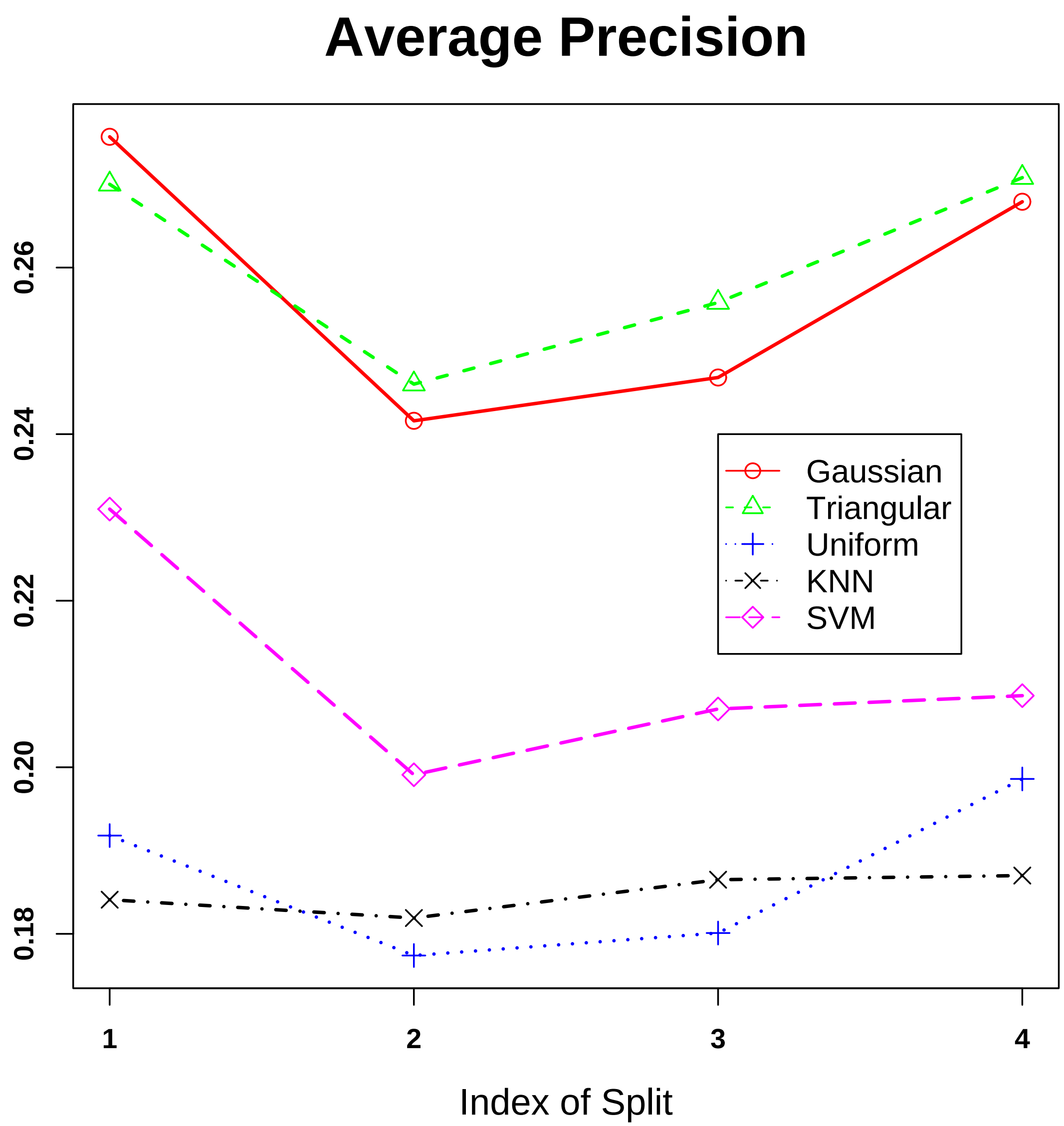
# The NCI database

**AIDS Antiviral Data**–from National Cancer Institute (NCI)

- Target–HIV1 virus;
- Response: $\{0, 1\}$ or $\{\text{inactive}, \text{active}\}$
- Explanatory variables: Molecular descriptors $(\text{BCUT}_1, \ldots, \text{BCUT}_6)$.

| | Training | Test | |
|---|---|---|---|
| Active | 304 | 304 | 608 |
| Inactive | 14,602 | 14,602 | 29,204 |
| Total | 14,906 | 14,906 | 29,812 |

# Results and Comparisons



# A Numeric Summary of Hit Curve

$$\text{Average Precision} = \frac{1}{n_1}\sum_{i=1}^{n}\frac{y_{(i)}h(i)}{i}$$

| Ranking | True Class Label | Precision |
|---|---|---|
| 1 | 1 | 1/1 |
| 2 | 1 | 2/2 |
| 3 | 0 | |
| 4 | 1 | 3/4 |
| 5 | 0 | |
| 6 | 0 | |

$$\text{AP}=\tfrac{1}{3}(1/1+2/2+3/4)\approx 0.9167$$