Statistical Wars: The Driven Force – Classification

⟨한국통계학회⟩ 2021년 춘계학술논문발표회 튜토리얼

신승준 (sjshin@korea.ac.kr)

고려대학교 통계학과

2021.05.27

Contents

Contents

Classification in Statistics I

- Classification is a type of statistical analysis to classify the data into several classes.
- In statistics, Data $\mathbf{y} = (y_1, y_2, \cdots, y_n)$ are realizations of

$$Y \sim \mathcal{P}_Y$$

- Statistical analysis is essentially the process of uncovering data generating process (DGP) of Y, $\mathcal{P}_Y$.

Classification in Statistics II

- If there exists a p-dimensional covariate $\mathbf{X}$ associated with Y, our goal is often to learn their relationship. i.e.,

$$Y \mid \mathbf{X} \sim \mathcal{P}_{Y|\mathbf{X}}$$

  - Regression refers to the case of continuous Y.
  - Classification refers to the case of categorical Y.
- In statistics, regression is more popular than classification
- In recent applications, however, (binary) classification becomes a standard.

DGP in Binary Classification, $p(\mathbf{x})$

- In binary classification, the DGP of interest is

$$p(\mathbf{x}) = P(Y = 1 \mid \mathbf{X}) = 1 - P(Y \neq 1 \mid \mathbf{X})$$

  which we call class probability.

- Classification rule for a given $\mathbf{X} = \mathbf{x}$ is

$$\hat{y} = \underset{y \in \{0,1\}}{\operatorname{argmin}}\, P(Y = y \mid \mathbf{X} = \mathbf{x}) = \begin{cases} 1 & \text{if } p(\mathbf{x}) \geq 0.5 \\ 0 & \text{otherwise} \end{cases}$$

- Binary classification is a process of learning $p(\mathbf{x})$ from the data.

Examples I

- Linear Discriminant Analysis (LDA) assumes

$$\mathbf{X} \mid Y = y \sim N(\boldsymbol{\mu}_y, \boldsymbol{\Sigma}), \qquad y \in \{0, 1\} \tag{1}$$

- Corresponding Classification Rule:

$$\frac{p(\mathbf{x})}{1 - p(\mathbf{x})} > 1 \qquad \Leftrightarrow \qquad \mathbf{w}^T \mathbf{x} > \mathbf{c},$$

where $\mathbf{w} = \boldsymbol{\Sigma}^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)^T$ and $\mathbf{c} = \mathbf{w}^T(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)/2$.

Examples II

- Logistic regression assumes

$$Y \mid \mathbf{X} = \mathbf{x} \sim \text{Bernoulli}\{p(\mathbf{x})\},$$

with

$$\log\left\{\frac{p(\mathbf{x})}{1 - p(\mathbf{x})}\right\} = \beta_0 + \boldsymbol{\beta}^T \mathbf{x}.$$

- The logit transformation is justified under (1).
- k-Nearest Neighbor and Naive Bayes are two elementary examples as well.

However ...

- Two goals of data analysis:
  - Interpretation
  - Prediction
- Uncovering $p(\mathbf{x})$ is more related to the goal of interpretation.
- Modern applications often focus on the prediction accuracy, and it suffice to have a good classification rule.

Classification Function, $f(\mathbf{x})$ I

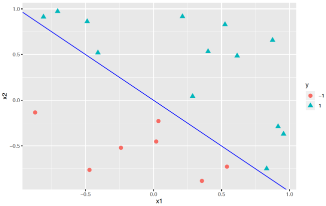
- Assume $y \in \{-1, 1\}$, WLOG.
- One can directly tackle the classification rule induced by $f(\mathbf{x})$.
- Classification rule based on $f(\mathbf{x})$ is given by

$$\hat{y} = \begin{cases} 1, & \text{if } f(\mathbf{x}) > 0 \\ -1, & \text{if } f(\mathbf{x}) < 0 \end{cases} = \text{sign}\{f(\mathbf{x})\}$$

- Classification can be viewed as a process of learning $f(\mathbf{x})$.

Geometric Approach for $f(\mathbf{x})$ I

- Consider a linearly separable case of $(y_i, \mathbf{x}_i), i = 1, \cdots, n$.



- The goal is to find a hyperplane (i.e., blue line)

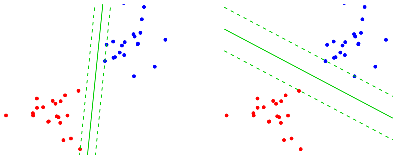
$$f(\mathbf{x}) = \beta_0 + \boldsymbol{\beta}^T \mathbf{x} = 0.$$

  that satisfies $y_i f(\mathbf{x}_i) > 0$, $i = 1, \cdots, n$.

- Perceptron is the oldest algorithm to find a separating hyperplanes.

Geometric Approach for $f(\mathbf{x})$ II

- One can try to find the optimal separating hyperplane, $f(\mathbf{x}) = 0$



(a) non-optimal f (b) Optimal f

- Optimal separating hyperplane is the solution of

$$\min_{\beta_0, \boldsymbol{\beta}} \frac{1}{2}\|\boldsymbol{\beta}\|^2, \quad \text{subject to} \;\; y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{x}_i) \geq 1$$

Geometric Approach for $f(\mathbf{x})$ III

- (Linear SVM) In linearly non-separable case, we need to relax the constraints.

$$\min_{\beta_0, \boldsymbol{\beta}} \frac{1}{2}\|\boldsymbol{\beta}\|^2 + C\sum_{i=1}^{n} \xi_i \tag{2}$$

$$\text{subject to } \; y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{x}_i) \geq 1 - \xi_i; \xi_i \geq 0$$

where $\xi_1, \cdots, \xi_n \geq 0$ are slack variables and $C > 0$ is the cost.

- We call $y_i(\beta_0 + \boldsymbol{\beta}^T \mathbf{x}_i)$ margin which is
  - distance from the separating hyperplane and thus measures the quality/performance the classifier.
  - similar to the residual in the regression context.

$$m_i = y_i f(\mathbf{x}_i) \qquad \Leftrightarrow \qquad r_i = y_i - f(\mathbf{x}_i)$$

Probabilistic Approach for $f(\mathbf{x})$ I

- Suppose Y is a random variable.
- The error rate of a classifier f is

$$P[Y \neq \text{sign}\{f(\mathbf{x})\}] = P\{Yf(\mathbf{x}) < 0\} = E[\mathbb{1}\{Yf(\mathbf{x}) < 0\}]$$

- Define the zero-one loss function of the margin $m = yf$.

$$L_{0\text{-}1}(m) = \mathbb{1}\{m < 0\},$$

- The optimal classifier that minimizes the classification error is known as Bayes Classifer:

$$f^{\text{Bayes}} = \min_{f \in \mathcal{F}} E[L_{0\text{-}1}\{Yf(\mathbf{x})\}] \tag{3}$$

Probabilistic Approach for $f(\mathbf{x})$ II

- The goal is to find f^{Bayes} in (3)
- Given $\{y_i, \mathbf{x}_i\}_{i=1}^n$, the sample version of (3) is

$$\hat{f}^{\text{Bayes}} = \underset{f \in \mathcal{F}}{\operatorname{argmin}}\, \mathbb{E}_n[L_{0\text{-}1}\{yf(\mathbf{x})\}] \tag{4}$$

  where $\mathbb{E}_n$ denotes the empirical expectation (i.e., sample average).
- Yet, (4) is not tractable due to
  - Infinite dimensionality of $\mathcal{F}$ $\quad\Rightarrow\quad$ (Add Constraints on $\mathcal{F}$)
  - Irregularity of the loss $L_{0\text{-}1}$ $\quad\Rightarrow\quad$ (Relaxation of $L_{0\text{-}1}$)

Probabilistic Approach for $f(\mathbf{x})$ III

- Empirical Risk Minimization (ERM) Formulation:

$$\min_{f \in \mathcal{F}} \mathbb{E}_n[L(yf(\mathbf{x})] + \lambda_n J(f) \tag{5}$$

  where $J(f)$ is a penalty functional and $\lambda_n > 0$ is a tuning parameter.

- Different choices of L (or $\mathcal{F}$) correspond to different classifiers.
- Similar to the regression problem.
- A standard empirical process theory can be exploited to study the asymptotic properties.

Examples I

- Logistic Regression solves

$$\min_{\beta_0,\boldsymbol{\beta}} \mathbb{E}_n[\log\{1+\exp\{-yf(\mathbf{x})\}]$$

  with $\mathcal{F}$ being the space of linear functions of $\mathbf{x}$.

- Kernel extension (KLR) becomes straightforward

$$\min_{f\in\mathcal{H}_K} \mathbb{E}_n[\log\{1+\exp\{-yf(\mathbf{x})\}] + \lambda_n\|f\|_{\mathcal{H}}^2$$

  where $\mathcal{H}_K$ denotes the RKHS generated by a kernel function K.

Examples II

- Recall that Linear SVM solves

$$\min_{\beta_0,\boldsymbol{\beta}} \frac{1}{2}\|\boldsymbol{\beta}\|^2 + C\sum_{i=1}^{n}\xi_i, \quad \text{s.t } y_i(\beta_0 + \boldsymbol{\beta}^T\mathbf{x}_i) \geq 1 - \xi_i; \xi_i \geq 0 \tag{6}$$

- (6) is equivalent to

$$\min_{\beta_0,\boldsymbol{\beta}} \mathbb{E}_n\{[1 - yf(\mathbf{x})]_+\} + \frac{\lambda}{2}\|\boldsymbol{\beta}\|^2$$

where $[a]_+ = \max\{0, a\}$.

- Kernel SVM solves

$$\min_{f\in\mathcal{H}_K} \mathbb{E}_n[L_{\text{SVM}}\{yf(\mathbf{x})\}] + \frac{\lambda}{2}\|f\|^2_{\mathcal{H}_K}$$

- Ada-Boosting is another example with $L(m) = \exp(-m)$ (Friedman, 2001).

Examples III

- Most, if not all popular classification methods essentially target f^{Bayes} with differently approximated $L_{0\text{-}1}$.

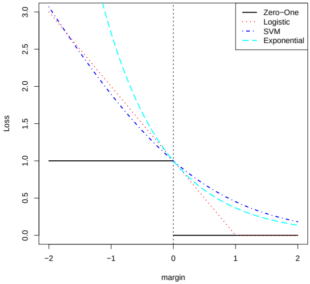


Figure: Convex relaxation of the zero-one loss $L_{0\text{-}1}$. (LR, SVM, and Boosting)

Fisher Consistency: A bridge between $f(\mathbf{x})$ and $p(\mathbf{x})$ I

- Can any convex loss replace $L_{0\text{-}1}$? Certainly not.
- Rhe goal is to find f^{Bayes}.
- Our working target is

$$f(\mathbf{x}) = \underset{f}{\operatorname{argmin}}\, E\{L(yf(\mathbf{x}))\}$$

- Both f^{Bayes} and $f(\mathbf{x})$ must provide identical classification rule. i.e.,

$$\operatorname{sign}\{f(\mathbf{x})\} = \operatorname{sign}\{f^{\text{Bayes}}(\mathbf{x})\}$$

Definition (Fisher Consistency/Classification Calibrated)

A loss function L is Fisher consistent (or classification calibrated) if its population risk minimizer leads the Bayes classification rule.

Fisher Consistency: A bridge between $f(\mathbf{x})$ and $p(\mathbf{x})$ II

- What conditions does L need for Fisher consistency?

Theorem (Bartlett et al., 2006)

Let L is convex. If L is differentiable at $m = 0$ and $L'(0) < 0$, then the convex loss L is Fisher consistent.

- Aforementioned convex loss functions are all Fisher consistent, and there are many variants.
- Although the convexity makes a lot of things simple, it is not essential.

Theorem (Lin, 2004)

If $L(m) < L(-m), \forall m > 0$ and $L(0)' \neq 0$ exists, then the loss L is Fisher consistent

Fisher Consistency: A bridge between $f(\mathbf{x})$ and $p(\mathbf{x})$ III

- For a given $\mathbf{x}$, it can be showed that

$$\operatorname{sign}\{f^{\text{Bayes}}(\mathbf{x})\} = \operatorname{sign}\{p(\mathbf{x}) - 1/2\}$$

- If L is Fisher consistent then

$$\operatorname{sign}\{f(\mathbf{x})\} = \operatorname{sign}\{p(\mathbf{x}) - 1/2\}$$

- Fisher consistency provides a theoretical connection between $p(\mathbf{x})$ and $f(\mathbf{x})$.

Estimation of $p(\mathbf{x})$ from $f(\mathbf{x})$ I

- Consider class-weighted version of the Bayes classifier.

$$f_\pi^{\text{Bayes}} = \min_{f \in \mathcal{F}} E[w_\pi(y) L_{0\text{-}1}(yf(\mathbf{x})]$$

where

$$w_\pi(y) = \begin{cases} 1-\pi & \text{if } y = 1 \\ \pi & \text{if } y = -1 \end{cases}$$

for a given $\pi \in (0,1)$ that controls relative class-importance.

- Bayes Classification rule is

$$\text{sign}\{f_\pi^{\text{Bayes}}(\mathbf{x})\} = \text{sign}\{p(\mathbf{x}) - \pi\}, \qquad \text{for a given } \pi \in (0,1)$$

Estimation of $p(\mathbf{x})$ from $f(\mathbf{x})$ II

- Let

$$f_\pi(\mathbf{x}) = \operatorname*{argmin}_f E[w_\pi(Y)L\{Yf(\mathbf{x})\}]$$

- Weighted version of the Fisher consistency states that

$$\text{sign}\{f_\pi(\mathbf{x})\} = \text{sign}\{p(\mathbf{x}) - \pi\}, \qquad \text{for a given } \pi \in (0,1)$$

- By Fisher consistency, we have

$$\begin{cases} p(\mathbf{x}) > \pi & \Leftrightarrow \quad f_\pi^*(\mathbf{x}) > 0 \\ p(\mathbf{x}) = \pi & \Leftrightarrow \quad f_\pi^*(\mathbf{x}) = 0 \\ p(\mathbf{x}) < \pi & \Leftrightarrow \quad f_\pi^*(\mathbf{x}) < 0 \end{cases}$$

Estimation of $p(\mathbf{x})$ from $f(\mathbf{x})$ III

- Estimation of $p(\mathbf{x})$ from $f_\pi(\mathbf{x})$. (Wang et al. 2008)
  1. Consider a grid of π, $\{0 < \pi_1 < \pi_2 < \cdots \pi_H < 1\}$.
  2. Solve a series of (7) for different values of $\pi_h, h = 1, \cdots, H$:

$$\hat{f}_{\pi_h} = \min_{f \in \mathcal{F}} \mathbb{E}_n[w_{\pi_h}(y) L\{y f(\mathbf{x})\}] + \lambda_n J(f) \tag{7}$$

     with a Fisher consistent loss L.
  3. One can estimate

$$\hat{p}(\mathbf{x}) = \frac{\hat{\pi}_+ + \hat{\pi}_-}{2}$$

     where

$$\hat{\pi}_+ = \max\{\pi_h : \hat{f}_{\pi_h}(\mathbf{x}) > 0\}, \quad \text{and} \quad \hat{\pi}_- = \min\{\pi_h : \hat{f}_{\pi_h}(\mathbf{x}) < 0\}.$$

Remarks on Part I

- Binary Classification can be viewed as a process of

  (1) Learning $p(\mathbf{x})$

  - Often require the assumptions on $(\mathbf{X}, Y)$.
  - Provide complete picture on DGP.

  (2) Learning $f(\mathbf{x})$

  - Goal is to minimize the error rate.
  - Cast into the ERM problem:

$$\min_{f \in \mathcal{F}} \mathbb{E}_n[L\{yf(\mathbf{x})\}] + \lambda_n J(f),$$

    which is very familiar to statisticians.

- Fisher consistency of L plays an important role to link $f(\mathbf{x})$ and $p(\mathbf{x})$.

Contents

Introduction I

- Large-scale data is frequently encountered.
  - Large p: High-dimensional data - Dimension Reduction
  - Large n: Data stream - Scalable Algorithms
- Types of Dimension Reduction
  - Feature (Variable) Selection
  - Feature (Variable) Screening
  - Feature Extraction

Variable Selection

- One of the most popular topic in modern statistical learning.
- Target:

$$\mathcal{S} = \{j : F(Y \mid \mathbf{X}) \text{ functionally depends on } X_j, j = 1, \cdots, p.\}$$

- A reasonable variable selection method yields an estimator $\hat{\mathcal{S}}_n$ that consistently estimate $\mathcal{S}$, i.e.,

$$P(\hat{\mathcal{S}}_n = \mathcal{S}) \to 1$$

known as Selection Consistency (for fixed p / $p = O(n^{\xi}), \xi > 0$).

Variable Selection in Linear Classifier I

- Under the linear model, $f(\mathbf{x}) = \beta_0 + \boldsymbol{\beta}^T\mathbf{x}$, we have

$$\mathcal{S} = \{j : \beta_j \neq 0, j = 1, \cdots, p\}$$

- After LASSO proposed, variable selection becomes straightforward:

$$(\hat{\beta}_{n,0}, \hat{\boldsymbol{\beta}}_n) = \underset{\beta_0, \boldsymbol{\beta}}{\operatorname{argmin}}\ \mathbb{E}_n[L\{y(\beta_0 + \boldsymbol{\beta}^T\mathbf{x})\}] + p_{\lambda_n}(\boldsymbol{\beta}),$$

  where $p_\lambda(\boldsymbol{\beta})$ denotes the sparsity-pursuing penalty such as LASSO (Tibshirani, 1996) with a tuning parameter λ.

- We have

$$\hat{\mathcal{S}}_n = \{j : \hat{\beta}_{n,j} \neq 0, j = 1, \cdots, p\}$$

Variable Selection in Linear Classifier II

- Popular choices include (for both regression and classification)
  - L_q-penalty:
    $$p_\lambda(\boldsymbol{\beta}) = \lambda \|\boldsymbol{\beta}\|_q, q \geq 0$$
    – $q = 0$ corresponds to the best subset selection.
    – $q = 1$ is LASSO and $q = 2$ is ridge.
  - Variants of LASSO:
    – Adaptive LASSO
    – Elastic net, Group LASSO, $\cdots$
  - Non-convex penalty: SCAD / MCP penalty
- Sparsity is a special case of homogeneity (Ke et al., 2012).
  - Fused LASSO, total variation penalty, Hybrid penalty, $\cdots$

Variable Selection for Nonlinear Classifier I

(COmponent Shrinkage and Selection Operator)

- Consider SS-ANOVA of $f \in \mathcal{F} = \{1\} \oplus \mathcal{F}^1 \oplus \cdots \oplus \mathcal{F}^p$.

$$f(\mathbf{x}) = \beta_0 + \sum_{j=1}^{p} f_j(x_j), \quad f_j \in \mathcal{F}^j$$

  where denotes the second order Sobolev space for X_j.

- COSSO (Lin and Zhang, 2006) solves

$$\min_f \mathbb{E}_n L[\{yf(\mathbf{x})\}] + \lambda \sum_{j=1}^{p} \theta_j \|P^j f\|$$

  where P^j denotes the projection operator to $\mathcal{F}^j, j = 1, \cdots, p$.

- COSSO with the logistic loss is available in `R`.

Variable Selection for Nonlinear Classifier II

(Variable Selection via Gradient Learning)

- If $X_j \notin \mathcal{S}$, then

$$\frac{\partial f(\mathbf{X})}{\partial X_j} = 0$$

- Taylor expansion of f around $\mathbf{x} \approx \mathbf{u}$ is

$$f(\mathbf{x}) = f(\mathbf{u}) + \nabla f(\mathbf{u})^\top (\mathbf{x} - \mathbf{u})$$

  where $\nabla f(\mathbf{x}) = \{\partial f(\mathbf{x})/\partial x_j, j = 1, \cdots, p\}$.

- Loss can be locally approximated by

$$\frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} L[y_i\{f(\mathbf{x}_j) + \nabla f(\mathbf{x}_j)^\top (\mathbf{x}_i - \mathbf{x}_j)\}]$$

  where $w_{ij} = w_s(\mathbf{x}_i - \mathbf{x}_j)$ is smoothing kernel with a bandwidth s.

Variable Selection for Nonlinear Classifier III

- Assume $f \in \mathcal{H}_K$, and $g_j = \partial f(\mathbf{x})/\partial x_j \in \mathcal{H}_K, j = 1, 2, \cdots, p$, then

$$f(\mathbf{x}) = \alpha_{00} + \sum_{i=1}^{n} \alpha_{i0} K(\mathbf{x}, \mathbf{x}_i), \text{ and}$$
$$g_j(\mathbf{x}) = \alpha_{0j} + \sum_{i=1}^{n} \alpha_{ij} K(\mathbf{x}, \mathbf{x}_i), \quad j = 1, \cdots, p$$

- We can solve

$$\min_{\boldsymbol{\alpha}} \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} w_{ij} L[y_i\{f(\mathbf{x}_j) + \mathbf{g}(\mathbf{x}_j)^\top (\mathbf{x}_i - \mathbf{x}_j)\}] + \sum_{j=0}^{p} \lambda_j \|\boldsymbol{\alpha}_j\|_2$$

where $\mathbf{g}(\mathbf{x}) = (g_1(\mathbf{x}), ..., g_p(\mathbf{x}))^T$ and $\boldsymbol{\alpha}_j = (\alpha_{1j}, ..., \alpha_{nj})$, $j = 1, \cdots, p$.

Independent Screening for Ultra-highdimensional Predictors I

- Ultra-highdimensional Predictor (Fan and Lv, 2008)

$$\mathbf{x} \in \mathbb{R}^p, \text{ with } \log(p) = O(n^\xi), \text{ for some } \xi > 0$$

- Penalized approach often fails due to
  - Accumulated estimation error
  - Computational complexity
- Fan and Lv (2008) proposed a two-stage approach for the ultra-highdimensional data analysis
  - Screening: Quickly filter out most of noise variables ($p \to \tilde{p}$)
  - Selection: Apply penalized variable selection to models with $\tilde{p}$ variables ($\tilde{p} \to d$)

Independent Screening for Ultra-highdimensional Predictors II

- Marginal screening under linear Model

$$y_i = \beta_0 + \mathbf{x}_i^T \boldsymbol{\beta} + \epsilon_i, i = 1, \cdots, n$$

- Compute the marginal utility (correlation)

$$u_j = |\mathbf{y}^T \mathbf{x}_j| \qquad \Leftrightarrow \qquad |\hat{\beta}_j|, \qquad j = 1, 2, \cdots, p$$

- Let

$$\tilde{\mathcal{S}}_n = \{\text{first } d \text{ largest values of } u_j.\}$$

- (Sure Screening Property)

$$P(\mathcal{S} \subset \tilde{\mathcal{S}}_n) \to 1$$

as $n \to \infty$ and $\log(p) = O(n^{\xi})$ for some $\xi > 0$.

Independent Screening for Ultra-highdimensional Predictors III

- Marginal utility u_j measures the relation between y and the jth predictor x_j.
- Distribution-based
  - Two-sample t-test statistics (Fan and Fan, 2009)

$$u_j = |(\bar{x}_j^+ - \bar{x}_j^-)/s_j|$$

  - Komogorov-Smirnov test (Mai and Zou, 2013)

$$u_j = \max \left| \hat{F}_{n,j}^+(x) - \hat{F}_{n,j}^-(x) \right|$$

- Loss-based (Fan et al., 2009, Fan and Song, 2010)
  - Loss

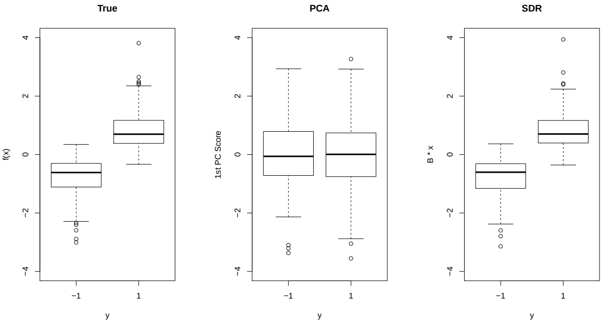
$$u_j = \min_f \mathbb{E}_n[L\{yf(x_j)\}] + \lambda_n J(f) \tag{8}$$

  - Minimizer

$$u_j = \|\hat{f}_j\|^2, \quad \text{where } f_j \text{ denotes the minimizer of (8).}$$

SDR in Binary Classification I

- PCA is a canonical example of feature extraction, but a unsupervised.



(a) True (b) PCA (c) SDR

Figure: Toy example: $y_i = \mathrm{sign}\{\mathbf{B}^T\mathbf{x} + \epsilon_i\}, i = 1, \cdots, 500$ with $\mathbf{B} = \mathbf{e}_1$ (i.e., $\mathbf{B}^T\mathbf{x} = x_1$), where $\mathbf{x} \overset{\text{iid}}{\sim} N_{10}(\mathbf{0}_{10}, \mathbf{I}_{10})$, $\epsilon_i \overset{\text{iid}}{\sim} N(0, 0.2^2)$.

SDR in Binary Classification II

- Sufficient Dimension Reduction (SDR) is a supervised DR method that seeks $\mathbf{B} \in \mathbb{R}^{p \times d}$ satisfying

$$Y \perp \mathbf{X} \mid \mathbf{B}^T \mathbf{X}.$$

- Dimension Reduction Subspace (DRS) is span$\{\mathbf{B}\}$ (not unique)
- Central Subspace (CS, $\mathcal{S}_{Y|\mathbf{X}}$)is intersection of all DRSes.
- Assuming $\mathcal{S}_{Y|\mathbf{X}} = \text{span}\{\mathbf{B}\}$, the goal of SDR is to identify $\mathcal{S}_{Y|\mathbf{X}}$.

SDR in Binary Classification III

- Slice Inverse Regression (SIR, Li, 1991) is the earliest proposal for SDR in the regression context.
- SIR is based on the fact that

$$E(\mathbf{X} \mid Y) \quad \in \quad \mathcal{S}_{Y \mid \mathbf{X}},$$

  when $\mathbf{X}$ is standardized.
- SIR estimates $E(\mathbf{X} \mid Y)$ by slicing the data based on the observed y_is.

SDR in Binary Classification IV

- SIR algorithm: Assuming $\mathbf{x}_i$s are standardized, WLOG:

1. (Slicing) Slice data using the grid of y, $\{c_1, \cdots, c_H\}$:

$$I_h = \{i : c_{h-1} < y_i < c_h\}, \ h = 1, \cdots, H.$$

2. (Estimation of $E(\mathbf{X} \mid Y)$ Compute the sample within-slice average

$$\bar{\mathbf{x}}_h = n_h^{-1} \sum_{i=1}^{n} \mathbf{x}_i \mathbb{1}\{i \in I_h\}, \ \text{where } n_h = |I_h|.$$

3. (Eigen-decomposition) First d leading eigenvectors of

$$\hat{\mathbf{M}}_n = \sum_{h=1}^{H} \frac{n_h}{n} \bar{\mathbf{x}}_h \bar{\mathbf{x}}_h^\top,$$

estimate (a basis set of) $\mathcal{S}_{Y|\mathbf{X}}$.

SDR in Binary Classification V

- In binary classification, SIR fails when $\dim(\mathcal{S}_{Y|\mathbf{X}}) > 1$ because there is only one slice-structure available.

$$I_1 = \{i : y_i = -1\} \quad \text{vs} \quad I_2 = \{i : y_i = 1\}$$

- Shin et al. (2014) showed

$$\mathcal{S}_{Y|\mathbf{X}} = \mathcal{S}_{p(\mathbf{X})|\mathbf{X}} \tag{9}$$

where $\mathcal{S}_{p(\mathbf{X})|\mathbf{X}}$ is analogously defined as $\mathcal{S}_{Y|\mathbf{X}}$, meaning that $p(\mathbf{x})$ has the same amount of information as y for SDR.

- Probability Enhanced SDR(PRE-SDR) slices the data based on $p(\mathbf{x}_i)$.

SDR in Binary Classification VI

- PRE-SIR replaces Step 1 in SIR algorithm with

$$I_h = \{i : \pi_{h-1} \leq p(\mathbf{x}_i) \leq \pi_h\} \tag{10}$$

for a given $\pi_h, h = 1, \cdots, H$.

- Recall the weighted version of Fisher consistency:

$$\text{sign}\{f_\pi^*(\mathbf{x})\} = \text{sign}\{p(\mathbf{x}) - \pi\}$$

- (10) is equivalent to

$$I_h = \{i : f_h^*(\mathbf{x}_i) > \pi_{h-1} \text{ and } f_h^*(\mathbf{x}_i) > \pi_h\}$$

which can be estimated by solving WSVM with $\pi = \pi_h, h = 1, \cdots, H$.

- Extension to other SDR methods such as SAVE, DR is straightforward.

SDR in Binary Classification VII

- Shin et al. (2017) proposed Principal Weighted Support Vector Machine (PWSVM) that solves

$$\boldsymbol{\beta}_\pi = \underset{\beta,\boldsymbol{\beta}}{\operatorname{argmin}}\, E[w_\pi(Y) L_{\text{SVM}}\{Y f(\mathbf{X})\}] + \lambda \boldsymbol{\beta}^T \boldsymbol{\Sigma} \boldsymbol{\beta}$$

where $f(\mathbf{X}) = \beta_0 + \boldsymbol{\beta}^T(\mathbf{X} - E\mathbf{X})$ and $\boldsymbol{\Sigma} = \text{cov}(\mathbf{X})$.

- Foundation of PWSVM:

$$\boldsymbol{\beta}_\pi \in \mathcal{S}_{Y|\mathbf{X}}, \text{ for any given } \pi \in (0,1).$$

which implies

$$\text{span}\{\boldsymbol{\beta}_{\pi_h}, h = 1, \cdots, H\} \subseteq \mathcal{S}_{Y|\mathbf{X}}$$

- PWSVM estimates $\boldsymbol{\beta}_\pi$ by solving a series of the weighted SVM (extendable to other Fisher consistent losses is possible).

Random Projection Ensemble Classification I

- Idea of Ensemble: *A committee of weak learners is powerful!*
- Construct a weak learner based on random projection. (Cannings and Samworth, 2017)

$$\mathbf{x} \in \mathbb{R}^p \qquad \rightarrow \qquad \mathbf{A}^T\mathbf{x} \in \mathbb{R}^d, \text{ where } d \ll p,$$

  where $\mathbf{A}$ is a random projection matrix $\mathbf{A}^T\mathbf{A} = \mathbf{I}_d$.
- Learn a simple classifier (ex. LDA) of y_i from $\mathbf{A}^T\mathbf{x}_i$ instead of $\mathbf{x}_i$.
- Then combine the results to produce a final prediction.

Random Projection Ensemble Classification II

- Obviously $\mathbf{A}$ cannot be completely arbitrary. But it suffices to choose an $\mathbf{A}$ that is slightly better than the random guess.
- `RandPro` Algorithm for Binary Classification.
  1. Generate B_2 random projection and produce B_2 classifiers based on $(y_i, \mathbf{A}_b^T \mathbf{x}_i), b = 1, \cdots, B_2$.
  2. Choose the projection and corresponding classifier that shows the best prediction performance among B_2 ones.
  3. Repeat Step 1-2 B_1 times to get B_1 classifiers to be used for the final prediction.
- The performance of the RP-ensemble is promising.
- The algorithm is easily parallelizable. (`RandPro` in `R`).

Remarks on Part II

- Dimension reduction is essential in high-dimensional data analysis.
- There are a variety of ways to reduce the predictor dimension.
- The ERM formulation enables us to naturally extend the standard regression technique to the binary classification (ex. variable selection, screening)
- SDR in binary classification is not a trivial extension, since slicing based on the response is not possible.
- Random projection ensemble is a very powerful alternative in high-dimensional binary classification.

Contents

Multiclass Classification I

- Response: $Y \in \{1, 2, \cdots, K\}$ with $K > 2$.
- Goal is to learn the class probability

$$p_k(\mathbf{x}) = (Y = k \mid \mathbf{X} = \mathbf{x}), k = 1, \cdots, K$$

- Prediction rule in order to minimize the classification error:

$$\hat{y} = \underset{k}{\operatorname{argmax}}\, p_k(\mathbf{x})$$

- Naive approach based on binary classifiers:
  - One vs One (Pairwise)
  - One vs Rest

Multiclass Classification II

- In the K-class problem, we need K-decision functions

$$\mathbf{f}(\mathbf{x}) := \{f_1(\mathbf{x}), \cdots, f_K(\mathbf{x})\} \in \mathbb{R}^K$$

- We need the sum-to-zero constraint for the identifiability.

$$\sum_{k=1}^{K} f_k(\mathbf{x}) = 0$$

Multiclass Classification III

- A natural idea is

$$\mathbf{f}^*(\mathbf{x}) := \{f_1^*(\mathbf{x}), \cdots, f_K^*(\mathbf{x})\} = \operatorname*{argmin}_{\mathbf{f}} E[L\{\mathbf{f}(\mathbf{x}), Y\}]$$

  for a loss function L.

- We say L is Fisher consistent if

$$\operatorname*{argmax}_k f_k^*(\mathbf{x}) = \operatorname*{argmax}_k p_k(\mathbf{x})$$

- It is NOT easy to derive a general condition of Fisher consistency in multiclass problem.

Multiclass Classification IV

- Multicategory SVM

$$\min_{f \in \mathcal{F}} \mathbb{E}_n[L\{\mathbf{f}(\mathbf{x}), y\}] + \lambda_n J(\mathbf{f})$$

$$\text{s.t} \sum_{k=1}^{K} f_k(\mathbf{x}) = 0$$

- A list of L proposed for multiclass SVM include

a.	(Lee et al., 2004)	$\sum_{k \neq y}[1 + f_k(\mathbf{x})]_+$
b.	(Naive Hinge)	$[1 - f_y(\mathbf{x})]_+$
c.	(Vapnik, 1998)	$\sum_{k \neq y}[1 - (f_y(\mathbf{x}) - f_j(\mathbf{x})]_+$
d.	(Crammer and Singer, 2001)	$[1 - \min_j(f_y(\mathbf{x}) - f_j(\mathbf{x}))]_+$

- All loss function encourages f_y to be the maximum among $\{f_1, \cdots f_K\}$.
- Only (a) satisfies the Fisher consistency (Liu, 2016).

Fisher Consistency in Multiclass Classification I

- Extension based on Multiple Comparison:

$$\mathbf{g}(\mathbf{x}, y) = \{f_y(\mathbf{x}) - f_k(\mathbf{x}), \forall k \neq y\}$$

- A classifier $\mathbf{f} \in \mathbb{R}^K$ yields a correct prediction if

$$\begin{aligned} y = \operatorname*{argmax}_k f_k(\mathbf{x}) \quad &\Leftrightarrow \quad \mathbf{g}(\mathbf{x}, y) > \mathbf{0}_{K-1} \\ &\Leftrightarrow \quad \min\{\mathbf{g}(\mathbf{x}, y)\} > 0 \end{aligned}$$

- Thus, $\min\{\mathbf{g}(\mathbf{f}, y)\}$ can be viewed as a multiclass version of margin.

Fisher Consistency in Multiclass Classification II

- An ERM formulation for the multiclass problem.

$$
\begin{aligned}
&\min_{\mathbf{f}} \mathbb{E}_n[L(\min\{\mathbf{g}(\mathbf{f}, y)\})] + \lambda_n J(\mathbf{f}) \\
&\text{s.t } \sum_{k=1}^{K} f_k(\mathbf{x}) = 0
\end{aligned}
$$

- We need a loss L that yields the Bayes classification rule.

Fisher Consistency in Multiclass Classification III

- Truncation of loss can guarantee the Fisher consistency for multiclass classification.

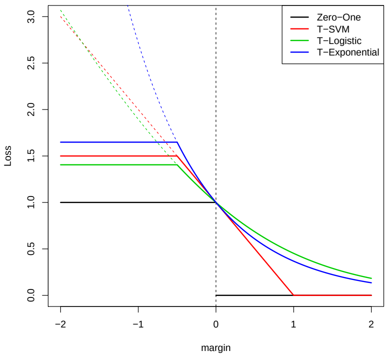


Figure: Truncated loss functions.

Fisher Consistency in Multiclass Classification IV

Theorem (Wu and Liu, 2007)

Assume that a loss $L(m)$ is non-increasing and that $L'(0) < 0$ exists. Let $L_{T_s}(m) = \min\{L(m), L(s)\}$ is a truncated $L(m)$ with $s \leq 0$. Then a sufficient condition for the loss $L_{T_s}(\min \mathbf{g}(\mathbf{f}, y))$ to be Fisher-consistent is that the value of s satisfies

$$\sup_{u: u \geq -s \geq 0} \frac{L(0) - L(u)}{L(s) - L(0)} \geq (k-1).$$

This condition is also necessary if L is convex.

Loss functions		Condition of s for FC
(SVM)	$[1-m]_+$	$-\frac{1}{k-1} \leq s \leq 0$
(Logistic)	$\log(1+\exp\{-m\})$	$-\log(2^{k/(k-1)} - 1) \leq s \leq 0$
(Exponential)	$\exp(-m)$	$\log(1 - \frac{1}{k} \leq s \leq 0$

Angle-based Multiclass Classification I

- However, the computation for the multiclass classification is not easy to optimize even for the convex loss due to the sum-to-zero constraint.
- Truncated loss is even more difficult due to its non-convexity.
- We need a better method.

Angle-based Multiclass Classification II

- Assume that the kth class corresponds to $\{\mathbf{w}_1, \cdots, \mathbf{w}_K\}$ which forms a simplex with K vertices in the $(K-1)$ dimensional space.

$$\mathbf{w}_k = \begin{cases} (K-1)^{-1/2}\mathbf{1}, & k = 1 \\ -\frac{1+K^{1/2}}{(k-1)^{3/2}}\mathbf{1} + \left(\frac{K}{K-1}\right)^{1/2}\mathbf{e}_{k-1}, & k = 2, \cdots, K \end{cases}$$

- Prediction rule based on $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^{K-1}$ is

$$\hat{y} = \underset{k}{\operatorname{argmin}} \angle(\mathbf{f}(\mathbf{x}), \mathbf{w}_k) = \underset{k}{\operatorname{argmax}} \langle \mathbf{f}(\mathbf{x}), \mathbf{w}_k \rangle$$

  where $\angle(\mathbf{v}, \mathbf{u})$ and $\langle \mathbf{v}, \mathbf{u} \rangle$ denote angle and inner product.

- By construction, we do NOT need the sum-to-zero constraint.

Angle-based Multiclass Classification III

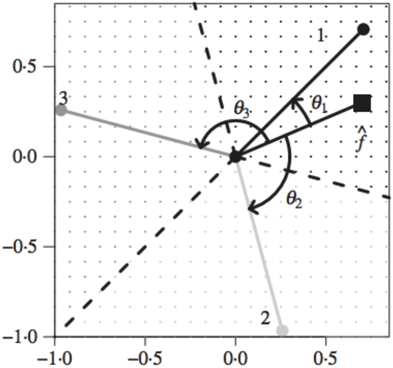


Figure: Angle-based $K = 3$-class Classification.

Angle-based Multiclass Classification IV

- Angle-based multiclass classification solves

$$\min_{\mathbf{f}} \mathbb{E}_n\{L(\langle \mathbf{f}(\mathbf{x}), \mathbf{w}_y \rangle)\} + \lambda_n J(\mathbf{f})$$

Theorem (Zhang and Liu, 2014)

In the angle-based classifier, a loss L is Fisher Consistent if L' exists and $L'(m) < 0$ for all m.

ROC-Optimizing Classification I

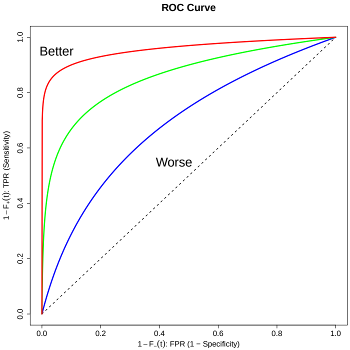
- In unbalanced classification, minimizing error rate may not be desired.
- Optimizing cost can be an alternative. It is equivalent to determine a threshold:

$$p(\mathbf{x}) > \pi \qquad \text{or} \qquad f(\mathbf{x}) > t$$

- However, it is NOT straightforward to choose the optimal threshold in practice.

ROC-Optimizing Classification II

- ROC Curve, a trajectory of $\{\text{TPR}(t), \text{FPR}(t)\}$ for $f(\mathbf{x})$ is a popular way to visualize the classification performance of $f(\mathbf{x})$ regardless of the threshold t.

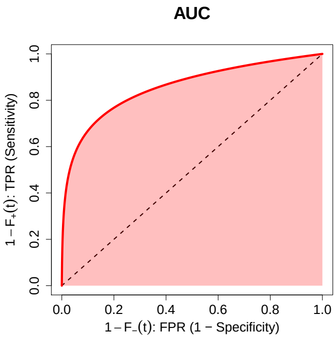


ROC-Optimizing Classification III

- By definition, the population AUC is

$$\mathrm{AUC}(f) = P\left[f(\mathbf{X}^{+}) > f(\mathbf{X}^{-})\right]$$

where $\mathbf{X}_{+} = \mathbf{X} \mid Y = 1$ and $\mathbf{X}_{-} = \mathbf{X} \mid Y = -1$.



ROC-Optimizing Classification IV

- Let $\mathbf{Z} = (Y, \mathbf{X}) \overset{iid}{\sim} \mathcal{P}, i = 1, 2$. The ROC margin is defined by

$$m_{12}(f) := m(\mathbf{Z}_1, \mathbf{Z}_2; f) = \frac{1}{2}\{Y_1 f(\mathbf{X}_1) + Y_2 f(\mathbf{X}_2)\}(1 - Y_1 Y_2).$$

- This is because

$$AUC(f) = P[m_{12}(f) > 0] = 1 - E[\mathbb{1}\{m_{12}(f) \leq 0\}].$$

ROC-Optimizing Classification V

- ROC-optimizing classifier is

$$f^{\text{Bayes}} = \min_f E[\mathbb{1}\{m(\mathbf{Z}_1, \mathbf{Z}_2; f) \le 0\}]$$

- For any given $\mathbf{X}_1 = \mathbf{x}_1$ and $\mathbf{X}_2 = \mathbf{x}_2$, we have

$$\text{sign}\{f^{\text{Bayes}}(\mathbf{x}_1) - f^{\text{Bayes}}(\mathbf{x}_2)\} = \text{sign}\{p(\mathbf{x}_1) - p(\mathbf{x}_2)\}.$$

Definition (Fisher Consistency in ROC-optimizing Classifier)

Let $f^* = \text{argmin}_f E[L\{M(\mathbf{Z}_1, \mathbf{Z}_2; f)\} \mid \mathbf{X}_1 = \mathbf{x}_1, \mathbf{X}_2 = \mathbf{x}_2]$. The loss L is Fisher consistent if for arbitrary given $\mathbf{X}_1 = \mathbf{x}_1$ and $\mathbf{X}_2 = \mathbf{x}_2$

$$\text{sign}\{f^*(\mathbf{x}_1) - f^*(\mathbf{x}_2)\} = \text{sign}\left\{p(\mathbf{x}_1) - p(\mathbf{x}_2)\right\}. \tag{11}$$

ROC-Optimizing Classification VI

Lemma (Sufficient Condition)

Assume that there is no ties (i.e., $p(\mathbf{x}_1) \neq p(\mathbf{x}_2)$ if $\mathbf{x}_1 \neq \mathbf{x}_2$). If $L(m) < L(-m)$ for any given $m > 0$, then the loss function L is Fisher consistent in ROC-optimizing classification.

- Similar to the result in Lin (2004) in the error-minimization context.

ROC-Optimizing Classification VII

- ROC-optimizing classifier solves

$$\min_f \frac{1}{\binom{n}{2}} \sum_{i<j} [L\{m(\mathbf{z}_i, \mathbf{z}_j; f)\}] + \lambda_n J(f)$$

  where L is a Fisher consistent loss function.

- Intercepter is unidentifiable. However, it can be uniquely determined for a given level of TPR and/or TFR.
- Optimization is straightforward especially for the convex loss. (`R` package is available upon request)
- U-process theory helps to explore the asymptotic.

Individualized Treatment Regime I

- In randomized treatment framework, we have $(\mathbf{X}, A, R)$ where
  - Action: $A \in \{1, 2, \cdots, K\}$ with a known prior prob. distribution $\pi(A, \mathbf{X})$.
  - Reward: $R \in \mathbb{R}$
  - Covariate: $\mathbf{X} \in \mathbb{R}^p$.
- Value function under the ITR $d(\mathbf{x})$ for a given $\mathbf{x}$ is

$$V(d) = E\{R \mid d(\mathbf{X}) = A\} = E\left[\frac{R}{\pi(A, \mathbf{X})} \mathbb{1}\{A = d(\mathbf{X})\}\right] \qquad (12)$$

- The optimal ITR is defined as the rule of actions that maximizes the value function:

$$d_0(\mathbf{x}) = \underset{d}{\operatorname{argmax}}\, V(d)$$

Individualized Treatment Regime II

- When $K = 2$ (2-armed bandit), we encode treatment A to be 1 or -1.
- Given $(\mathbf{x}_i, a_i, r_i), i = 1, \cdots, n$, the empirical version of $V(d)$ in (12) is

$$\hat{V}_n(d) := \frac{1}{n} \sum_{i=1}^{n} \frac{r_i}{\pi(a_i, \mathbf{x}_i)} \mathbb{1}\{a_i = d(\mathbf{x}_i)\}.$$

- Note that

$$d_0(x) = \text{sign}\{f(\mathbf{x})\}, \qquad \text{for some function } f.$$

- We can rewrite the empirical value function as

$$\hat{V}_n(d) := \frac{1}{n} \sum_{i=1}^{n} \frac{r_i}{\pi(a_i, \mathbf{x}_i)} \mathbb{1}\{a_i f(\mathbf{x}_i) < 0\}.$$

- We can cast ITR into weighted binary classification problem.

Individualized Treatment Regime III

- Zhao et al. (2012) proposed the outcome weighted learning (OWL) by replacing the indicator function with the hinge loss

$$\frac{1}{n}\sum_{i=1}^{n}\frac{r_i}{\pi(a_i,\mathbf{x}_i)}(1-a_i f(\mathbf{x}_i))_+ + \lambda_n J(f)$$

Theorem (Fisher Consistency in OWL, Zhao et al. (2012))

Let f^* be the population solution of the OWL, i.e.,

$$f^* = \underset{f}{\operatorname{argmin}}\, E\left\{\frac{R}{\pi(A,\mathbf{X})}[1 - Af(\mathbf{X}_i) < 0]_+\right\}.$$

Then $d_0(\mathbf{X}) = sign\{f^*(\mathbf{X})\}$ where d_0 denotes the ITR that maximizes population value function.

Individualized Treatment Regime IV

- The goal of ITR is identify the rule of discrete actions that yields the best result (i.e., maximizing the value function).
- There are various ways to solve ITR based on the classification idea.

Remarks on Part III

- We have seen that the idea of binary classification can be extended to various context
  - Multiclass problem
  - ROC-optimizing problem (more generally ranking problem)
  - Individual treatment regime for precision medicine
- Some other extensions (not covered in this tutorial) include
  - Top-k classification.
  - Semi-supervised classification.
  - Anomaly detection (a.k.a. one-class classificaiton)

References I

1. Cristianini and Shawe-Taylor (2000) "An introduction to support vector machines and other kernel-based learning methods" Cambridge university press.
2. Friedman (2001) "Greedy function approximation: a gradient boosting machine" *AOS*, pp.1189-1232.
3. Bartlett, Jordan and McAuliffe (2006) "Convexity, classification, and risk bounds" *JASA*, 101(473), 138-156.
4. Lin (2004) "A note on margin-based loss functions in classification" *Stat & prob letters*, 68(1), pp.73-82.
5. Wang, Shen, and Liu (2007) "Probability estimation for large-margin classifiers" *Biometrika*, 95, 149-167.
6. Tibshirani (1996) "Regression shrinkage and selection via the lasso" *JRSSb*, 58(1), pp.267-288.
7. Ke, Fan and Wu (2015) "Homogeneity pursuit" *JASA*, 110(509), 175-194.
8. Lin and Zhang (2006) "Component selection and smoothing in multivariate nonparametric regression" *AOS*, 34(5), 2272-2297.
9. Fan and Lv (2008) "Sure independence screening for ultrahigh dimensional feature space" *JRSSb*, 70(5), 849-911.
10. Fan and Fan (2008) "High dimensional classification using features annealed independence rules" *AOS*, 36(6), 2605–2637.
11. Fan, Samworth, and Wu (2009) "Ultrahigh dimensional feature selection: beyond the linear model" *JMLR*, 10, 2013-2038.

References II

12. Fan and Song (2010) “Sure independence screening in generalized linear models with NP-dimensionality” *AOS*, 38(6), 3567-3604.
13. Mai and Zou (2013) “The Kolmogorov filter for variable screening in high-dimensional binary classification” *Biometrika*, 100(1), 229-234.
14. Li (1991) “Sliced inverse regression for dimension reduction” *JASA*, 86(414), 316-327.
15. Shin, Wu, Zhang, and Liu (2014) “Probability-enhanced sufficient dimension reduction for binary classification” *Biometrics*, 70(3), 546-555.
16. Shin, Wu, Zhang, and Liu (2017) “Principal weighted support vector machines for sufficient dimension reduction in binary classification” *Biometrika*, 104(1), 67-81.
17. Cannings and Samworth (2015) “Random-projection ensemble classification” *JRSSb*,79(4), 1021-1022.
18. Lee, Lin, and Wahba (2004) “Multicategory Support Vector Machines, theory, and application to the classification of microarray data and satellite radiance data.” *JASA*, 99(465) 67-81.
19. Vapnik (1998) “Statistical learning theory”. Wiley, New York.
20. Crammer and Singer (2001) “On the algorithmic implementation of multiclass kernelbased vector machines” *JMLR*, 2, 265-292.
21. Liu (2007) “Fisher consistency of multicategory support vector machines” *AISTAT*.
22. Wu and Liu (2014) “Wu and Liu (2007) Robust truncated hinge loss support vector machines. *JASA*, 102(479), 974-983.”
23. Zhang and Liu (2014) “Multicategory angle-based large-margin classification” *Biometrika*, 101(3), 625-640.
24. Zhao, Zeng, Rush, and Kosorok (2012) “Estimating individualized treatment rules using outcome weighted learning” *JASA*, 107(499), 1106-1118.