STOR 767 Spring 2019 Hw6

Due on 03/27/2019 in Class

Problem Set

Remark. This homework focuses on tree-based methods, boosting and unsupervised learning methods.

Instruction.

- **Theoretical Part and Computational Part are respectively credited 60 points. At most 100 points in total will be accounted for this homework.**
- Submission of handwritten solution for the **Theoretical Part** of this homework is allowed.
- Please use **RMarkdown** to create a formatted report for the **Computational Part** of this homework.
- Some of the problems are selected or modified from the textbook [Friedman et al., 2009].

Theoretical Part

1. (*15 pt*) (Variance Reduction via Bagging [Bühlmann and Yu, 2002, Section 2.1]) Let $\{Y_i\}_{i=1}^{\infty} \overset{iid}{\sim} (0,1)$. Denote a size-$n$ sample as $\mathcal{Y}_n := \{Y_i\}_{i=1}^n$, a Bootstrap sample (*i.e.* sampling with replacement) from $\mathcal{Y}_n$ as $\mathcal{Y}_n^*$. Consider the parameter of interest

$$\theta_n(\mathcal{Y}_n) := \mathbb{1}(\bar{Y}_n \leqslant 0)$$

where $\bar{Y}_n := \frac{1}{n}\sum_{i=1}^n Y_i$ is the sample average of $\mathcal{Y}_n$, and the Bootstrap estimate

$$\theta_n^*(\mathcal{Y}_n) := \mathbb{E}_{\mathcal{Y}_n^*|\mathcal{Y}_n}\theta_n(\mathcal{Y}_n^*)$$

where $\mathbb{E}_{\mathcal{Y}_n^*|\mathcal{Y}_n}$ takes the (conditional) expectation on the Bootstrap sample $\mathcal{Y}_n^*$ give $\mathcal{Y}_n$[1].

(a) Show that

$$\theta_n(\mathcal{Y}_n) \xrightarrow[n\to\infty]{d} \mathbb{1}(Z \leqslant 0)$$

where $Z \sim \mathcal{N}(0,1)$.

[1] In practice we take finite Bootstrap approximation to $\mathbb{E}_{\mathcal{Y}_n^*|\mathcal{Y}_n}$. That is, consider B (conditional) *i.i.d.* Bootstrap samples $\{\mathcal{Y}_{nb}^*\}_{b=1}^B$ given $\mathcal{Y}_n$. Then

$$\mathbb{E}^*_{\mathcal{Y}_n^*|\mathcal{Y}_n}\theta_n(\mathcal{Y}_n^*) = \lim_{B\to+\infty}\frac{1}{B}\sum_{b=1}^B \theta_n(\mathcal{Y}_{nb}^*) \quad a.s.$$

by Strong Law of Large Number.

Hint (Continuous Mapping Theorem)**.** If $X_n \xrightarrow[n\to\infty]{d} X$, $\mathbb{P}(X \in D_f) = 0$ where D_f denotes the discontinuous set of f, then $f(X_n) \xrightarrow[n\to\infty]{d} f(X)$.

(b) Argue that

$$\mathbb{E}_{\mathcal{Y}_n}\theta_n(\mathcal{Y}_n) \overset{n\to\infty}{\longrightarrow} \Phi(0)$$

where Φ denotes the CDF of $\mathcal{N}(0,1)$.

(c) Show that

$$\theta_n^*(\mathcal{Y}_n) \xrightarrow[n\to\infty]{d} \Phi(-Z) \sim \textbf{Uniform}[0,1].$$

Hint. Note that

$$\theta_n(\mathcal{Y}_n^*) = \mathbb{1}(\bar{Y}_n^* \leqslant 0) = \mathbb{1}\left\{\sqrt{n}(\bar{Y}_n^* - \bar{Y}_n)/S_n \leqslant -\sqrt{n}\bar{Y}_n/S_n\right\}$$

where $S_n^2 := \frac{1}{n}\sum_{i=1}^n (Y_i - \bar{Y}_n)^2$ is the sample variance. Conditional on $\mathcal{Y}_\infty$, take $\mathbb{E}_{\mathcal{Y}_n^*|\mathcal{Y}_n}$ and send $n \to \infty$[2].

(d) Compare $\lim_{n\to\infty} \mathbb{E}_{\mathcal{Y}_n}\theta_n(\mathcal{Y}_n)$ versus $\lim_{n\to\infty} \mathbb{E}_{\mathcal{Y}_n}\theta_n^*(\mathcal{Y}_n)$, and $\lim_{n\to\infty} \textbf{Var}_{\mathcal{Y}_n}\theta_n(\mathcal{Y}_n)$ versus $\lim_{n\to\infty} \textbf{Var}_{\mathcal{Y}_n}\theta_n^*(\mathcal{Y}_n)$[3].

2. (*25 pt*) (Boosting) Consider the Forward Stagewise Additive Modeling (Textbook Section 10.3, Algorithm 10.2) framework.

(a) (AdaBoost) Assume exponential loss and ± 1-valued (classification) trees as individual classifiers. Develop the **AdaBoost** algorithm from scratch.

(b) (Boosting Tree) Relax individual classifiers in (a) to arbitrarily valued (regression) trees. Develop the general **boosting tree** (Textbook Section 10.9) algorithm from scratch. You might need to derive: 1) the prediction given a tree partition, *i.e.* Textbook Equation (10.32); 2) the criterion that the tree partition optimizes; and 3) the weight update.

Punchline. Textbook Equation (10.32) is **INCORRECT**. It should be divided by 2.

(c) (GBM Logistic Regression) Assume deviance loss (negative log-likelihood of Logistic regression) for two-class classification. Develop the **gradient boosting Logistic regression tree** algorithm (which is also known as a Generalized Boosted Regression Model (GBM) for Logistic regression) from scratch. General steps within one iteration are as follows:

1) perform an in-sample gradient step to identify the targets;

[2]For a rigorous proof, we need to consider an appropriate Skorokhod representation $(\mathcal{Y}'_\infty, \mathcal{Y}'^*_\infty) \overset{d}{=} (\mathcal{Y}_\infty, \mathcal{Y}^*_\infty)$, $Z' \overset{d}{=} Z$, $\sqrt{n}\bar{Y}'_n \overset{n\to\infty}{\longrightarrow} Z'$ *a.s.*, hence conditional on $\mathcal{Y}'_\infty$ with $Z'^*|\mathcal{Y}'_\infty \sim \mathcal{N}(0,1)$,

$$\mathbb{1}\left\{\sqrt{n}(\bar{Y}'^*_n - \bar{Y}'_n)/S'_n \leqslant -\sqrt{n}\bar{Y}'_n/S'_n\right\} \xrightarrow[n\to\infty]{d} \mathbb{1}(Z'^* \leqslant -Z')$$

by Slutsky Theorem and Continuous Mapping Theorem. You might skip such rigor.

[3]Bühlmann and Yu [2002] provided a nice insight that the limiting distribution of $\theta_n(\mathcal{Y}_n)$ is discontinuous in Z while $\theta_n^*(\mathcal{Y}_n)$ has a (smooth) probit link to Z (shown in their FIG. 1). Hence the bagging estimate has lower asymptotic variance.

2) perform a regression tree on the targets to extract a partition;

3) perform a boosting tree step to optimize the prediction given partition;

4) update the prediction function.

Hint. Ex. 10.8 and 10.9 have developed Algorithm 10.4 for a K-class classification problem. Develop a simpler algorithm of the same structure. Note that in step 3), the optimization involves another iterative subprogram. Algorithm 10.4 only performs one-step iteration and then proceeds to step 4).

3. (*20 pt*) (Classical MDS, Textbook Ex. 14.11) Suppose $\{\boldsymbol{x}_i\}_{i=1}^n \subseteq \mathbb{R}^p$, $\bar{\boldsymbol{x}} = \frac{1}{n}\sum_{i=1}^n \boldsymbol{x}_i$ is their sample mean, $\mathbf{S} = [s_{ii'} = \langle \boldsymbol{x}_i - \bar{\boldsymbol{x}}, \boldsymbol{x}_{i'} - \bar{\boldsymbol{x}} \rangle]_{n\times n}$ is their centered inner-product matrix. Fix $k \leqslant n-1$. Let $\lambda_1 \geqslant \lambda_2 \geqslant \cdots \geqslant \lambda_k$ be the k largest eigenvalues of $\mathbf{S}$, associated with eigenvectors $\mathbf{U}_k := [\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_k] \in \mathbb{R}^{n\times k}$. Denote $\mathbf{D}_k := \mathrm{diag}\{\sqrt{\lambda_j}\}_{j=1}^k$. Consider the **classical MDS** stress function (Textbook Equation (14.100))

$$S_{\mathrm{C}}(\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_n) := \sum_{i=1}^n \sum_{i'=1}^n (s_{ii'} - \langle \boldsymbol{z}_i - \bar{\boldsymbol{z}}, \boldsymbol{z}_{i'} - \bar{\boldsymbol{z}} \rangle)^2. \quad \left(\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_n \in \mathbb{R}^k,\ \bar{\boldsymbol{z}} = \frac{1}{n}\sum_{i=1}^n \boldsymbol{z}_i \right)$$

Show that the rows of $\mathbf{U}_k\mathbf{D}_k$ minimizes S_{C}.

Hint. Consider in matrix form $\mathbf{Z} := [\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_n]^T \in \mathbb{R}^{n\times k}$ and a centering matrix $\mathbf{H}_n := \mathbf{I}_{n\times n} - \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^T$, work on the SVD of $\mathbf{H}_n\mathbf{Z}$.

Computational Part

1. (*20 pt*) **Email Spam Data**

   The *Email Spam data*[4] is introduced in Chapter 1 Example 1 in Textbook. Reproduce the analysis in Section 9.2.5 and 10.8, Figure 15.1, 15.4 and 15.5.

2. (*20 pt*) **Authorship Data**

   The *Authorship data* consists of word counts for stop words from chapters written by four British authors.

   (a) **Visualization:** Create a visual summary of the (a) author texts, (b) words, and (c) both words and author texts. Methods such as PCA, MDS and nonlinear MDS (isomap) with various distances may be helpful. Which methods yield the most interpretable visual representations of the data?

   (b) **Clustering:** Use $K(=4)$-means and hierarchical clustering with several linkages and distances to cluster author texts. Which method is best at correctly grouping texts from the four British authors? Report your clustering accuracy.

[4] Available at https://web.stanford.edu/~hastie/ElemStatLearn/datasets/spam.data and in **R** package `ElemStatLearn`.

3. (*20 pt*) **Microarray Gene Expression Data**

   The *TCGA data* contains the expression of a random sample of 2000 genes for 563 patients from three cancer subtypes: Basal (`Basal`), Luminal A (`LumA`), and Luminal B (`LumB`). Suppose we are only interested in distinguishing Luminal A samples from Luminal B - but alas, we also have Basal samples, and we don't know which is which.

   (a) Perform K-means clustering for various K's and report their within cluster sum of square (WSS) plot. Comment on the WSS at $K = 3$.

   (b) Perform $K(= 3)$-means clustering and visualize the clustering result. What percentage of `Basal`, `LumA`, and `LumB` type samples are in each of the 3 resulting clusters? Did we do a good job in distinguishing `LumA` from `LumB`?

   (c) Use dimension reduction techniques (PCA, MDS and nonlinear MDS with various distances) and perform $K(= 3)$-means on their reduced spaces. Compare the clustering results to (b) visually. Which method shows the best performance to distinguish `LumA` from `LumB`?

   (d) Suppose we might know that the first PC contains information that we aren't interested in. Repeat (b) and (c) on the dataset **partialling out the first PC**.

References

Peter Bühlmann and Bin Yu. Analyzing bagging. *The Annals of Statistics*, 30(4):927–961, 2002. 1, 2

Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. Springer-Verlag, https://web.stanford.edu/~hastie/Papers/ESLII.pdf, second edition, 2009. 1