STAT 542, Homework 5

April 7, 2018
Due date: Apr 20 (Fri), 5 pm to Compass

Requirements: This homework consists of three problems. You should submit your **report and `R` code** as separate files to Compass2g. The report should be in PDF/html/MS Word format, with no more than 10 pages, including plots. Your report should focus on describing the analysis strategy, demonstrating and interpreting the main result. No excessive `R` code should be included in the report. Font size should be 12pt and plots need to be clearly labeled. If you use R markdown and need a template, there is an example source file provided at our course website. This homework worth 100 points total. Late submission penalty is 5 points for each day (round up) of delay.

Question 1: [40 points] Lets write our own code for a one-dimensional adaboost using a stump model as the weak learner.

a) [20 points] The stump model is a CART model with one split, hence having just two terminal nodes. Since we consider only one dimensional predictor, the only thing that needs to be searched in this tree model is the cutting point. Write a function to fit the stump model with subject weights:

Algorithm 1 A stump model, with weights

Input: A set of data $\{x_i, y_i, w_i\}_{i=1}^n$

1. Search for a splitting rule $I(x \le c)$ that will maximize the weighted reduction of Gini impurity.

$$\texttt{score} = -\frac{\sum_{\mathcal{T}_L} w_i}{\sum w_i}\text{Gini}(\mathcal{T}_L) - \frac{\sum_{\mathcal{T}_R} w_i}{\sum w_i}\text{Gini}(\mathcal{T}_R)$$

where, for given data in a node $\mathcal{T}$, the weighted version of Gini is

$$\text{Gini}(\mathcal{T}) = \widehat{p}(1-\widehat{p}),\ \widehat{p} = (\sum w_i)^{-1}\sum w_i I(y_i = 1)$$

2. Calculate the left and the right node weighted predictions f_L, f_R $\in \{-1, 1\}$ respectively.

Output: The cutting point c, and node predictions f_L, f_R.

---

a) [20 points] Following the lecture slides page 6 in "Boosting.pdf", write your own code to fit the adaboost model using the stump as the base learner. You should also calculate the

exponential error upper bound for each iteration and plot it along with the training error, like page 21. For this implementation, you are not required to do bootstrapping for each tree (you still can if you want); You should generate the following data to test your code and demonstrate that it is correct. Also generate an independent set of testing data using this model to see if there is any potential overfitting. Comment on your findings.

```
n = 300
x = runif(n)
y = (rbinom(n, 1, (sin(4*pi*x)+1)/2) -0.5)*2
```

Note that to obtain the probability from the fitting decision function $F_T(x)$, you can consider using $(1 + \exp(-2F(x)))^{-1}$.

Question 2: [20 points] Prove the property used in the block matrix inverse form — an alternative version of the Sherman-Morrison formula: If $A \in \mathbb{R}^{n\times n}$ is an invertible square matrix, $b \in R^n$ is a column vector. If $A - bb^{\mathrm{T}}$ is invertible, show that its inverse is given by:

$$\left(A - bb^{\mathrm{T}}\right)^{-1} = A^{-1} + \frac{A^{-1}bb^{\mathrm{T}}A^{-1}}{1 - b^{\mathrm{T}}A^{-1}b}$$

Hint: to show $A^{-1} = B$, verify that $AB = I$.

Question 3: [40 points] Take the Tate Collection dataset from Kaggle.

https://www.kaggle.com/rtatman/the-tate-collection

Read the descriptions carefully and perform a clustering analysis. You should consider the following and provide a thorough discussion for each:

- What is your goal when performing this clustering?
- What variable(s) to include? How do you construct/process the variables?
- What clustering algorithm(s) do you use? Find at least one clustering method that is NOT introduced in our class, and implement it. Discuss the advantages/disadvantages of all methods you considered.
- After the clustering is done, summarize and interpret your results.