Modern Data Mining - HW 5

Group Member 1
Group Member 2
Group Member 3

Due: 11:59Pm, 4/10, 2022

Contents

Overview

For the purpose of predictions, a model free approach could be beneficial. A binary decision tree is the simplest, still interpretable and often provides insightful information between predictors and responses. To improve the predictive power we would like to aggregate many equations, especially uncorrelated ones. One clever way to have many free samples is to take bootstrap samples. For each bootstrap sample we build a random tree by taking a randomly chosen number of variables to be split at each node. We then take average of all the random bootstrap trees to have our final prediction equation. This is RandomForest.

Ensemble method can be applied broadly: simply take average or weighted average of many different equations. This may beat any single equation in your hand.

All the methods covered can handle both continuous responses as well as categorical response with multiple levels (not limited to binary response.)

Objectives

- Understand trees
  - single tree/displaying/pruning a tree
  - RandomForest
  - Ensemble idea

- R functions/Packages
  - `tree`, `RandomForest`, `ranger`
- Json data format
- text mining
  - bag of words

Data needed:

- `IQ.Full.csv`
- `yelp_review_20k.json`

Problem 0: Lectures

Please study all three lectures. Understand the main elements in each lecture and be able to run and compile the lectures

- textmining
- trees
- boosting

Problem 1: IQ and successes

Background: Measurement of Intelligence

Case Study: how intelligence relates to one's future successes?

Data needed: `IQ.Full.csv`

ASVAB (Armed Services Vocational Aptitude Battery) tests have been used as a screening test for those who want to join the army or other jobs.

Our data set IQ.csv is a subset of individuals from the 1979 National Longitudinal Study of Youth (NLSY79) survey who were re-interviewed in 2006. Information about family, personal demographic such as gender, race and education level, plus a set of ASVAB (Armed Services Vocational Aptitude Battery) test scores are available. It is STILL used as a screening test for those who want to join the army! ASVAB scores were 1981 and income was 2005.

Our goals:

- Is IQ related to one's successes measured by Income?
- Is there evidence to show that Females are under-paid?
- What are the best possible prediction models to predict future income?

The ASVAB has the following components:

- Science, Arith (Arithmetic reasoning), Word (Word knowledge), Parag (Paragraph comprehension), Numer (Numerical operation), Coding (Coding speed), Auto (Automative and Shop information), Math (Math knowledge), Mechanic (Mechanic Comprehension) and Elec (Electronic information).
- AFQT (Armed Forces Qualifying Test) is a combination of Word, Parag, Math and Arith.
- Note: Service Branch requirement: Army 31, Navy 35, Marines 31, Air Force 36, and Coast Guard 45,(out of 100 which is the max!)

The detailed variable definitions:

Personal Demographic Variables:

- Race: 1 = Hispanic, 2 = Black, 3 = Not Hispanic or Black
- Gender: a factor with levels "female" and "male"

- Educ: years of education completed by 2006

Household Environment:

- Imagazine: a variable taking on the value 1 if anyone in the respondent's household regularly read magazines in 1979, otherwise 0
- Inewspaper: a variable taking on the value 1 if anyone in the respondent's household regularly read newspapers in 1979, otherwise 0
- Ilibrary: a variable taking on the value 1 if anyone in the respondent's household had a library card in 1979, otherwise 0
- MotherEd: mother's years of education
- FatherEd: father's years of education

Variables Related to ASVAB test Scores in 1981 (Proxy of IQ's)

- AFQT: percentile score on the AFQT intelligence test in 1981
- Coding: score on the Coding Speed test in 1981
- Auto: score on the Automotive and Shop test in 1981
- Mechanic: score on the Mechanic test in 1981
- Elec: score on the Electronics Information test in 1981
- Science: score on the General Science test in 1981
- Math: score on the Math test in 1981
- Arith: score on the Arithmetic Reasoning test in 1981
- Word: score on the Word Knowledge Test in 1981
- Parag: score on the Paragraph Comprehension test in 1981
- Numer: score on the Numerical Operations test in 1981

Variable Related to Life Success in 2006

- Income2005: total annual income from wages and salary in 2005. We will use a natural log transformation over the income.

Note: All the Esteem scores shouldn't be used as predictors to predict income

1. EDA: Some cleaning work is needed to organize the data.

- The first variable is the label for each person. Take that out.
- Set categorical variables as factors.
- Make log transformation for Income and take the original Income out
- Take the last person out of the dataset and label it as **Michelle**.
- When needed, split data to three portions: training, testing and validation (70%/20%/10%)
  - training data: get a fit
  - testing data: find the best tuning parameters/best models
  - validation data: only used in your final model to report the accuracy.

2. Factors affect Income

We only use linear models to answer the questions below.

i. To summarize ASVAB test scores, create PC1 and PC2 of 10 scores of ASVAB tests and label them as ASVAB_PC1 and ASVAB_PC2. Give a quick interpretation of each ASVAB_PC1 and ASVAB_PC2 in terms of the original 10 tests.

ii. Is there any evidence showing ASVAB test scores in terms of ASVAB_PC1 and ASVAB_PC2, might affect the Income? Show your work here. You may control a few other variables, including gender.

iii. Is there any evidence to show that there is gender bias against either male or female in terms of income in the above model?

We next build a few models for the purpose of prediction using all the information available. From now on you may use the three data sets setting (training/testing/validation) when it is appropriate.

3. Trees

i. fit1: tree(Income ~ Educ + Gender, data.train) with default set up

   a) Display the tree
   b) How many end nodes? Briefly explain how the estimation is obtained in each end nodes and deescribe the prediction equation
   c) Does it show interaction effect of Gender and Educ over Income?
   d) Predict Michelle's income

ii. fit2: fit2 <- rpart(Income2005 ~., data.train, minsplit=20, cp=.009)

   a) Display the tree using plot(as.party(fit2), main="Final Tree with Rpart")
   b) A brief summary of the fit2
   c) Compare testing errors between fit1 and fit2. Is the training error from fit2 always less than that from fit1? Is the testing error from fit2 always smaller than that from fit1?
   d) You may prune the fit2 to get a tree with small testing error.

iii. fit3: bag two trees

```
a) Take 2 bootstrap training samples and build two trees using the
rpart(Income2005 ~., data.train.b, minsplit=20, cp=.009). Display both trees.
b) Explain how to get fitted values for Michelle by bagging the two trees obtained above. Do not use th
c) What is the testing error for the bagged tree. Is it guaranteed that the testing error by bagging th
```

iv. fit4: Build a best possible RandomForest

   a) Show the process how you tune mtry and number of trees. Give a very high level explanation how fit4 is built.
   b) Compare the oob errors form fit4 to the testing errors using your testing data. Are you convinced that oob errors estimate testing error reasonably well.
   c) What is the predicted value for Michelle?

v. Now you have built so many predicted models (fit1 through fit4 in this section). What about build a fit5 which bags fit1 through fit4. Does fit5 have the smallest testing error?

vi. Summarize the results and nail down one best possible final model you will recommend to predict income. Explain briefly why this is the best choice. Finally for the first time evaluate the prediction error using the validating data set.

vii. Use your final model to predict Michelle's income.

Problem 2: Yelp challenge 2019

Note: This problem is rather involved. It covers essentially all the main materials we have done so far in this semester. It could be thought as a guideline for your final project if you want when appropriate.

Yelp has made their data available to public and launched Yelp challenge. More information. It is unlikely we will win the $5,000 prize posted but we get to use their data for free. We have done a detailed analysis in our lecture. This exercise is designed for you to get hands on the whole process.

For this case study, we downloaded the data and took a 20k subset from **review.json**. *json* is another format for data. It is flexible and commonly-used for websites. Each item/subject/sample is contained in a brace *{}*. Data is stored as **key-value** pairs inside the brace. *Key* is the counterpart of column name in *csv* and *value* is the content/data. Both *key* and *value* are quoted. Each pair is separated by a comma. The following is an example of one item/subject/sample.

```
{
  "key1": "value1",
  "key2": "value2"
}
```

Data needed: yelp_review_20k.json available in Canvas.

yelp_review_20k.json contains full review text data including the user_id that wrote the review and the business_id the review is written for. Here's an example of one review.

```
{
    // string, 22 character unique review id
    "review_id": "zdSx_SD6obEhz9VrW9uAWA",

    // string, 22 character unique user id, maps to the user in user.json
    "user_id": "Ha3iJu77CxlrFm-vQRs_8g",

    // string, 22 character business id, maps to business in business.json
    "business_id": "tnhfDv5Il8EaGSXZGiuQGg",

    // integer, star rating
    "stars": 4,

    // string, date formatted YYYY-MM-DD
    "date": "2016-03-09",

    // string, the review itself
    "text": "Great place to hang out after work: the prices are decent, and the ambience is fun. It's a

    // integer, number of useful votes received
    "useful": 0,

    // integer, number of funny votes received
    "funny": 0,

    // integer, number of cool votes received
    "cool": 0
}
```

Goal of the study

The goals are

1) Try to identify important words associated with positive ratings and negative ratings. Collectively we have a sentiment analysis.
2) To predict ratings using different methods.

1. JSON data and preprocessing data

i. Load *json* data

The *json* data provided is formatted as newline delimited JSON (ndjson). It is relatively new and useful for streaming.

```
{
  "key1": "value1",
  "key2": "value2"
}
{
  "key1": "value1",
  "key2": "value2"
}
```

The traditional JSON format is as follows.

```
[{
  "key1": "value1",
  "key2": "value2"
},
{
  "key1": "value1",
  "key2": "value2"
}]
```

We use `stream_in()` in the `jsonlite` package to load the JSON data (of ndjson format) as `data.frame`. (For the traditional JSON file, use `fromJSON()` function.)

```
pacman::p_load(jsonlite)
yelp_data <- jsonlite::stream_in(file("data/yelp_review_20k.json"), verbose = F)
str(yelp_data)
```

```
## 'data.frame':    19999 obs. of  9 variables:
##  $ review_id  : chr  "Q1sbwvVQXV2734tPgoKj4Q" "GJXCdrto3ASJOqKeVWPi6Q" "2TzJjDVDEuAW6MR5Vuc1ug" "yi0
##  $ user_id    : chr  "hG7b0MtEbXx5QzbzE6C_VA" "yXQM5uF2jS6es16SJzNHfg" "n6-Gk65cPZL6Uz8qRm3NYw" "dac
##  $ business_id: chr  "ujmEBvifdJM6h6RLv4wQIg" "NZnhc2sEQy3RmzKTZnqtwQ" "WTqjgwHlXbSFevF32_DJVw" "ikC
##  $ stars      : num  1 5 5 5 1 4 3 1 2 3 ...
##  $ useful     : int  6 0 3 0 7 0 5 3 1 1 ...
##  $ funny      : int  1 0 0 0 0 0 4 1 0 0 ...
##  $ cool       : int  0 0 0 0 0 0 5 1 0 1 ...
##  $ text       : chr  "Total bill for this horrible service? Over $8Gs. These crooks actually had the
##  $ date       : chr  "2013-05-07 04:34:36" "2017-01-14 21:30:33" "2016-11-09 20:09:03" "2018-01-09 2
```

```
# different JSON format
# tmp_json <- toJSON(yelp_data[1:10,])
# fromJSON(tmp_json)
```

Write a brief summary about the data:

a) Which time period were the reviews collected in this data?

b) Are ratings (with 5 levels) related to month of the year or days of the week? Only address this through EDA please.

ii. Document term matrix (dtm)

Extract document term matrix for texts to keep words appearing at least .5% of the time among all 20000 documents. Go through the similar process of cleansing as we did in the lecture.

a) Briefly explain what does this matrix record? What is the cell number at row 100 and column 405? What does it represent?

b) What is the sparsity of the dtm obtained here? What does that mean?

iii. Set the stars as a two category response variable called rating to be "1" = 5,4 and "0"= 1,2,3. Combine the variable rating with the dtm as a data frame called data2.

Analysis

Get a training data with 13000 reviews and the 5000 reserved as the testing data. Keep the rest (2000) as our validation data set.

2. LASSO

i. Use the training data to get Lasso fit. Choose lambda.1se. Keep the result here.

ii. Feed the output from Lasso above, get a logistic regression.

a) Pull out all the positive coefficients and the corresponding words. Rank the coefficients in a decreasing order. Report the leading 2 words and the coefficients. Describe briefly the interpretation for those two coefficients.

b) Make a word cloud with the top 100 positive words according to their coefficients. Interpret the cloud briefly.

c) Repeat i) and ii) for the bag of negative words.

d) Summarize the findings.

iii. Using majority votes find the testing errors i) From Lasso fit in 3) ii) From logistic regression in 4) iii) Which one is smaller?

3. Random Forest

i. Briefly summarize the method of Random Forest

ii. Now train the data using the training data set by RF. Get the testing error of majority vote. Also explain how you tune the tuning parameters (`mtry` and `ntree`).

4. Boosting

To be determined.

5. PCA first

i. Perform PCA (better to do sparse PCA) for the input matrix first. Decide how many PC's you may want to take and why.

ii. Pick up one of your favorate method above and build the predictive model with PC's. Say you use RandomForest.

iii. What is the testing error? Is this testing error better than that obtained using the original x's?

6. Ensemble model

i. Take average of some of the models built above (also try all of them) and this gives us the fifth model. Report it's testing error. (Do you have more models to be bagged, try it.)

7. Final model

Which classifier(s) seem to produce the least testing error? Are you surprised? Report the final model and accompany the validation error. Once again this is THE only time you use the validation data set. For the purpose of prediction, comment on how would you predict a rating if you are given a review (not a tm output) using our final model?