Computing for Data Sciences – 2017

PGDBA, First Year, First Semester, Indian Statistical Institute

Assignment 3

Posted on 20 October 2017 | Clarify by 25 October 2017 | Submit by 30 October 2017

Problem 1 [40 points]

Spam Detection for Text Messages
`http://www.souravsengupta.com/cds2017/evaluation/smsdata.txt`

Background: Text Message Spam are unsolicited text messages (SMS), especially advertising, directed at mobile phones or smartphones. As the popularity of mobile phones surged in the early 2000s, frequent users of text messaging began to see an increase in the number of unsolicited (and generally unwanted) commercial advertisements being sent to their telephones through text messaging (SMS). This can be particularly annoying for the recipient because, unlike in email, some recipients may be charged a fee for every message received, including the spam messages.

In this challenge, you are asked to complete the analysis of what type of text messages are likely to be spam. In particular, you are asked to apply the tools of machine learning to predict which messages in a corpus are spam. You may treat it as a classic case-study of Binary Classification.

Problem statement: Use as training set the labeled (`good` and `spam`) text messages available at `http://www.souravsengupta.com/cds2017/evaluation/smsdata.txt` to build a robust tree-based binary classifier that is capable of distinguishing spam text messages from regular ones. It will be nice if you may use a pruned decision tree, a bagging model, a random forest classifier, and a gradient boosted model, successively, to notice the respective advantages and disadvantages.

Submission: Submit the `R` (or Python) code you wrote to solve this problem as a single program file – `groupXXassign3prob1.R` (or `groupXXassign3prob1.py`), where `XX` is your group number. In the commented section, please mention your problem-solving approach, the main text processing packages you used, and acknowledge any online/offline resources you have consulted.

Dataset: Note that the training set of labeled text messages is structured as follows, where the first element is the label, either `good` or `spam`, and then the text message is posted as raw text.

```
good    Go until jurong point, crazy.. Available only in bugis n great world la e buffet...
good    Ok lar... Joking wif u oni...
spam    Free entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005.
good    U dun say so early hor... U c already then say...
good    Nah I don't think he goes to usf, he lives around here though
spam    FreeMsg Hey there darling it's been 3 week's now and no word back!
good    Even my brother is not like to speak with me. They treat me like aids patent.
```

Dataset available at — `http://www.souravsengupta.com/cds2017/evaluation/smsdata.txt`

Problem 2 [20 points]

Spam Detection for Text Messages
`http://www.souravsengupta.com/cds2017/evaluation/smsdata.txt`

Problem statement: Use as training set the labeled (`good` and `spam`) text messages available at `http://www.souravsengupta.com/cds2017/evaluation/smsdata.txt` to build a robust support vector classifier that is capable of distinguishing spam text messages from regular ones.

Submission: Submit the `R` (or Python) code you wrote to solve this problem as a single program file – `groupXXassign3prob2.R` (or `groupXXassign3prob2.py`), where `XX` is your group number. In the commented section, please mention your problem-solving approach, the main text processing packages you used, and acknowledge any online/offline resources you have consulted.

Problem 3 [40 points]

Handwritten Images Dataset: Digit Recognizer
`https://www.kaggle.com/c/digit-recognizer`

Background: MNIST (Modified National Institute of Standards and Technology) is the de facto "hello world" dataset of computer vision. Since its release in 1999, this classic dataset of handwritten images has served as the basis for benchmarking classification algorithms. As new machine learning techniques emerge, MNIST remains a reliable resource for researchers and learners alike. In this assignment, you will learn the application of SVM on this classic dataset.

In this challenge, your goal is to correctly identify digits from a dataset of handwritten images. In particular, you should apply Support Vector Machines to classify handwritten images to match decimal digits. You may treat it as a classic case-study of Multi-class Classification with images.

Problem statement: Take part in the basic online Kaggle Competition – "Digit Recognizer" – available at `https://www.kaggle.com/c/digit-recognizer`. Submit your predictions online.

Submission: Submit the `R` (or Python) code you wrote for the competition as a single program file – `groupXXassign3prob3.R` (or `groupXXassign3prob3.py`), where `XX` is your group number. In the commented section, please mention the name of your Kaggle team and your final score (standing) on the leaderboard. You may use any public *Kernel* or *Discussion* posted on Kaggle, with proper acknowledgement for each such reference mentioned in the commented section.

Restriction: Please restrict yourselves to Support Vector Machines (SVM) and Kernel Methods for the classification. You may use any kernel of your choice for the SVM, and you may also use Principal Component Analysis and other Low-Dimensional Embedding techniques, if required.

Your submission should be emailed to `sg.sourav@gmail.com` by midnight of 30 October 2017.

Properly acknowledge every source of information that you referred to, including discussions with other groups. Verbatim copy from any source is strongly discouraged, and plagiarism will be heavily penalized. It is strongly recommended that you write the codes completely on your own.