Predicting dropouts in MOOC
Course Project

COMP 4331 Data Mining, Fall 2018

Due: 2 Dec 2018, 11:59pm

1 Submission Guidelines

- Project should be sumbmitted to comp4331fall18@gmail.com as attachments.
- You need to zip the following two files together:
  - Proj_itsc_groupid_report.pdf/.docx: Please put all your reports in this file. (Attachments should be original .pdf or .docx, NOT compressed)
  - Proj_itsc_stuid_code.zip: The zip file contains all your source codes for the first assignment.
- The group size is AT MOST 3. Each group is required to send your group member information to comp4331fall18@gmail.com before 11 Nov 2018.
- All attachments, including report and codes, should be named in the format of: Proj_itsc_stuid.zip. E.g., for a student with itsc account: sdiaa, student id: 20171234, the project can be named as: Proj_sdiaa_20171234.zip. The student should be the member who send group information in above term.
- Submissions not following the rules above are NOT accepted.
- 20 marks will be deducted for every 24 hours after the deadline.
- Your grade will be based on the correctness, efficiency and clarity.
- The email for Q&A: sdiaa@connect.ust.hk or yzhangee@connect.ust.hk.
- **Plagiarism will lead to zero mark.**

2 Introduction

2.1 Background

Students' high dropout rate on MOOC platforms has been heavily criticized, and predicting their likelihood of dropout would be useful for maintaining and encouraging students' learning activities.

Therefore, in this project, we will **predict dropout** on XuetangX, one of the largest MOOC platforms in China.

2.2 Description

The competition participants need to predict whether a user will drop a course within next 10 days based on his or her prior activities. If a user C leaves no records for course C in the log during the next 10 days, we define it as dropout from course C For more details about log, please refer to the Data Descriptions `https://github.com/comp4331fall18/sample-code-DM/blob/master/project/DataDescription.txt`.

In short, you need to predict **whether** a user will drop a course within next 10 days based on his or her *prior activities*.

2.3 About XuetangX

XuetangX, a Chinese MOOC learning platform initiated by Tsinghua University, was officially launched online on Oct 10th, 2013. In April 2014, XuetangX signed a contract with edX, one of the biggest global MOOC learning platform co-founded by Harvard University and MIT, to acquire the exclusive authorization of edX's high-quality international courses. In December 2014, XuetangX signed the Memorandum of Cooperation with FUN, the national MOOC platform in France, to make bilateral effort in course construction, platform development and other aspects. So far, there are more than 100 Chinese courses and over 260 international courses available on XuetangX.

3 Task Description

You are provided with a list of information:

- Course information: duration, category, chapters, sections, objects, ...
- Enrollment: a user U enrolled in a course C.
- Behavior/Event: time, source, event type, object.
- Dropout: whether the enrollment will dropout.

The whole dataset can be downloaded in `https://drive.google.com/open?id=1jeEZsB62LSu5WGHcT0h90gQBXH7xVRbD`.

3.1 Load from CSV

The following script is a toy example about loading data in **csv** file row by row. You can also use python packages like **csv** or **pandas** to read data.

```
fin = open(filename)
fin.next()
for line in fin:
    print(line.strip().split(','))
```

We suggest you to use *python dictionary* to store the information of enrollment, user, class, and object, etc. A sample code of loading the 'enrollment_train.csv' information is given in `https://github.com/comp4331fall18/sample-code-DM/blob/master/project/read_data.py`.

Given a date in format yyyy-mm-dd or yyyy/mm/dd, you can use the following script to convert the date into an index. Besides, this structure can help you to quickly compute the duration between two dates.

```
import pandas as pd
def get_time_dict():
    rng = pd.date_range('2013-10-27', '2014-08-01')
    time_dict = pd.Series(np.range(len(rng)), index=rng)
    print(time_dict['2013/10/30'])
    return time_dict
```

3.2 Data preprocessing

After loading the csv files into some data structure, you are required to

- describe what information you gather from which csv file;
- decide and describe which feature you will use, like time duration, source, events;

- describe what kind of data pre-processing techniques you use on each feature, like data transformation or discretization;
- build up the feature vector and describe the meaning of each dimension.

After this procedure, you can get a feature vector for each enrollment.

3.3 Prediction

Given the vectorized features, you are required to

- use at least 3 classifiers in scikit-learn to predict based on the features in previous step. Labels of each enrollment are given in 'true_train.csv' file;
- clearly describe the settings (like kernel for SVM, hyper-parameters) you use for building each classifier;
- we suggest you to use cross-validation to pick up the optimal hyper-parameters;
- you can also use the other classifier beyond those in scikit-learn as the basic model. But you should state clearly about the details of the algorithm you use. A reasonable and powerful classifier will be regarded as bonus.
- Based on the classifiers you use, you are required to use boosting to ensemble these classifiers.
- report the precision and recall on both training and testing sets of each classifier, including the ensemble method.

4 Grading Scheme

- Loading data (5 marks) — read_data.py
- Data preprocessing (20 marks) — preprocess.py
- Classification (20 marks) — main.py
- Evaluation (5 marks): — main.py
  - Performance should be evaluated by the precision and recall
  - You should print the two values for each classifier under optimal parameters.
- Essential comment will be helpful for your grading.
- Project report and representation (50 marks):
  - The report should be well-organized and clearly written.
  - Explain what and why you do for each step.

  - The presentation will be arranged at the last tutorials. Every group member is required to attend.
  - State the division of work among the members in your group.

- Penalty:
  - 20 marks will be deducted for every 24 hours after the deadline.
  - 20 marks will be deducted for every 1 member beyond the maximum group size 3.
  - Plagiarism will lead to zero mark.

- Bonus:
  - Advanced feature processing. (5 marks)

    If you use any advanced or novel pre-processing techniques, you can state out and give enough explanation about what advantage you gain by using this technique.
  - Imbalanced data. (5 marks)

    There are more 'non-dropout' label than its counterpart, you can think about how to solve this imbalance.
  - Good presentation (5 marks)

    The mostly impressed presentation will gain this.

- The total mark will not exceed 100, including the bonus.

While you may discuss with your classmates on general ideas about the assignment, your submission should be based on your own independent effort. In case you seek help from any person or reference source, you should state it clearly in your submission. Failure to do so is considered plagiarism which will lead to appropriate disciplinary actions.