Back to Week 4 | Lessons | Prev | Course Home

Peer-graded Assignment: Kaggle Competition Peer Review

A Gradient Boosting approach to the Instacart Market Basket Analysis challenge

by Fernando Flores

July 16, 2017

like Flag this submission

PROMPT

Part 1: Problem Description. Give the name of the competition you selected and write a few sentences describing the competition problem as you interpreted it. You want your writeup to be self-contained so your peer-reviewer does not need to go to Kaggle to study the competition description. Clarity is more important than detail. What's the overall goal? What does the data look like? How will the results be evaluated?

RUBRIC

Is the problem description clear and comprehensible?

○ 0 pts
The submission makes no real attempt to describe the problem.

○ 2 pts
The submission makes an attempt to describe the problem, I don't really understand what is being described.

Example: "The task is to predict whether a given passenger survived the sinking of the Titanic based on various attributes including age, location of the passenger's cabin on the ship, family members, the fare they paid, and other information. Solutions are evaluated by comparing the percentage of correct answers on a test dataset."

The name of the competition I took a part in is "Instacart Market Basket Analysis". The task to perform consists on predicting the products that will be purchased by a series of users based on their previous orders. The dataset provided includes information about the orders and products, and other supportive information such as aisle and department data that might be used to create a feature set.

● 5 pts
The problem description is clear enough; I understand what is going on.

PROMPT

Part 2: Analysis Approach. Write a few sentences describing how you approached the problem. What techniques did you use?

Example: "I split the data by gender and handled each class separately. For the females, I trivially classified all of them as "survived." For the males, I trained a random forest as a classifier. I ignored the pclass

RUBRIC

Is the approach to the problem described clearly? Do you have some idea how you might employ these techniques to solve the problem?

○ 0 pts
No real attempt was made to describe the approach to the problem.

○ 2 pts

atribute that indicated the location of the passenger's cabin because I didn't think it was relevant."

The data is divided into three datasets: a list of the "n-1" prior orders of each user, and train and test sets containing the "n" order. With this in mind, the way to model user data is to think that we will have two different sets: "train" and "test" users. So it is needed to split the dataset between train and test user ids and extract their corresponding "n-1" orders. I chose to ignore categorical data such as aisle and department id. For model fitting and classification I have chosen a tool called LightGBM, developed by Microsoft. LightGBM is a library that performs Gradient Boosting over decision trees and that whose performance is state-of-the-art in tasks such as classification or ranking.

There's a description, but I don't fully understand what it's saying.

● 5 pts
The description is clear enough; I understand the approach.

PROMPT

Part 3: Initial Solution. Write a few sentences describing how you implemented your approach. What languages and libraries did you use? What challenges did you run into?

Example: "I partitioned the data by gender manually using Excel. I used Weka to build the random forest."

I run my code in Python and used

RUBRIC

Is the initial solution implementation clearly described? Are the tools, languages, and libraries used reasonable?

○ 0 pts
No significant attempt was made to answer the question.

○ 2 pts

Pandas to store data into dataframes and Numpy to format the data columns in an efficient way, in order to make the algorithm run faster. I divided the algorithm into three blocks, the first one generates the datasets needed for LightGBM to work, the second trains the algorithm and the third one tests it, generates predictions and saves the outcomes into a submission file:

1.1. The first task was to split the orders dataset (loaded from a CSV file) into prior, test and train data.
1.2. Later I joined prior data (previous orders for both train and test users) with product data, obtaining a list of previous orders and the products purchased per order.
1.3. My goal is to obtain a list of the products purchased previously by every user, so I grouped the previously mentioned dataframe by user id, obtaining a dataframe with a column that will be a list of these products per each user (all the users are included into the prior dataset) and the user id as index.

2.1. Now it is time to fit the model with train data. But in order to train the classifier I need to label the train dataset. So per every train order and every product purchased previously by the user who ordered it I return a boolean: 0 if the product is not purchased in the train set or 1 if it is.

The implementation is described, but I don't fully understand how it works.

● 5 pts
The implementation is well-described; I understand how it works.

2.2 I train LightGBM over the labelled train data. This step returns a Booster object, that is, a fitted model.

3.1 I use the Booster to perform a prediction for each order and product of the test set. This step will return a list of predictions: normalized predictions on how likely is for the product to be purchased in the order.
3.2 I set a threshold to determine which products are more likely to be purchased per order.
3.3 Save the list of test orders and the prediction into a CSV file.

Help Center

PROMPT

Part 4: Initial Solution Analysis. Write a few sentences assessing your approach. Did it work? What do you think the problems were?

Example: "My approach did not work so well, achieving a score of 0.65. This is less than the sample solution. I suspect I should not have ignored the pclass attribute."

This first approach, yet succesful, did not take a high accuracy along. The LB score (I did not evaluate the performance in terms of precision and accuracy, so I did not compute a F1 metric) is 0.214. I ended up quite low in the leaderboard.

RUBRIC

Is the initial solution analysis comprehensible and reasonable?

○ 0 pts
No significant attempt was made to assess the approach.

○ 2 pts
There's an assessment, but I don't really understand what is being described.

● 5 pts
The assessment is adequately described; I understand what's going on.

PROMPT

Part 5: Revised Solution and Analysis. Write a few sentences describing how you improved on your solution, and whether or not it worked.

Example: "I included the pclass attribute and ignored the ticket number attribute. My score improved to 0.68."

I tried to improve the accuracy rate in four ways: Improve the number of features used to fit the model, change the model internal parameters, perform Cross Validation and a perform a better threshold adjustment.

1. About the features, I added up a series of new features such as frequency of orders per user, number of orders and reorders per product, median basket size per user, most suitable day of the week for an order to be made, etc. This task is critical and a good feature selection will make users to boost their prediction and thus their position in the leaderboard.

2. I changed a few model parameters such as the branching and bagging factors but they seemed not to have a major effect in the performance. What did help was the setting of two concrete

RUBRIC

Is the revised solution and analysis presented clearly?

○ 0 pts
No significant attempt was made to describe an improved solution.

○ 2 pts
There's a description, but I don't fully understand it.

● 5 pts
The improvement is adequately described; I understand it.

parameters: boosting type and the number of leaves of each decision tree. About boosting type, at first I used an standard Random Forest, but it is clearly outperformed by the native Gradient Boosting Decision Tree implemented by LightGBM. A suitable number of leaves varies from 95 to 100.

3. LightGBM supports Cross Validation, and combining it with a reasonable number of boosting rounds will help to return the most suitable model. I used a 10-fold CV approach in 10 boosting rounds to find the best result.

4. In order to define the best threshold I defined an interval between 0.2 to 0.3 (other users recommended different values within this range) with an step size of 0.01, submitting later the prediction performed in each step. I found out the best predictions are returned by a threshold of 0.23.

After applying these changes the LB score raised to 0.318, still way far from the top players but an improvement compared to the initial approach.

OVERALL ASSIGNMENT RUBRIC

Overall evaluation-Given the description provided, do you feel you would be able to reproduce the result?

○ 0 pts
No, the description is too sketchy, incomplete, or unclear for me to be able to reproduce this result.

○ 2 pts
No, but primarily because I don't have a strong enough background in the area; there is a fair amount of jargon used.

○ 4 pts
Yes, I think I could reproduce the result with some effort.

● 5 pts
Yes, the solution is clearly and thoroughly described -- I could follow them easily.

Evaluating the analyses of others' is a great way to pick up new insights and viewpoints. What is one helpful thing you got out of reviewing this submission? (e.g. maybe ideas on analyzing solutions, incentive to go try this competition, etc.)?

ideas on analyzing solutions.

(This question will not affect the submitter's score.) You're not expected to actually try and reproduce the submitter's result, but you may choose to do so. Did you try to reproduce the result?

● No

○ Yes, but my result was different

○ Yes, and my result was the same.

Submit Review

Comments

Comments left for the learner are visible only to that learner and the person who left the comment.

share your thoughts...