

CTR prediction contest

Avazu

Avazu Click-Through Rate Prediction

Late submission solution by Pavel Troshenkov

Moscow, Russia, 2018

Table of contents

- Task
- Data observation
- Feature engineering
- Models
- Result
- Area for improvement

Task

In this competition we are asked for CTR problem. We need to predict the probability of user's click on advertisement

Metrics

- Logloss

Dataset

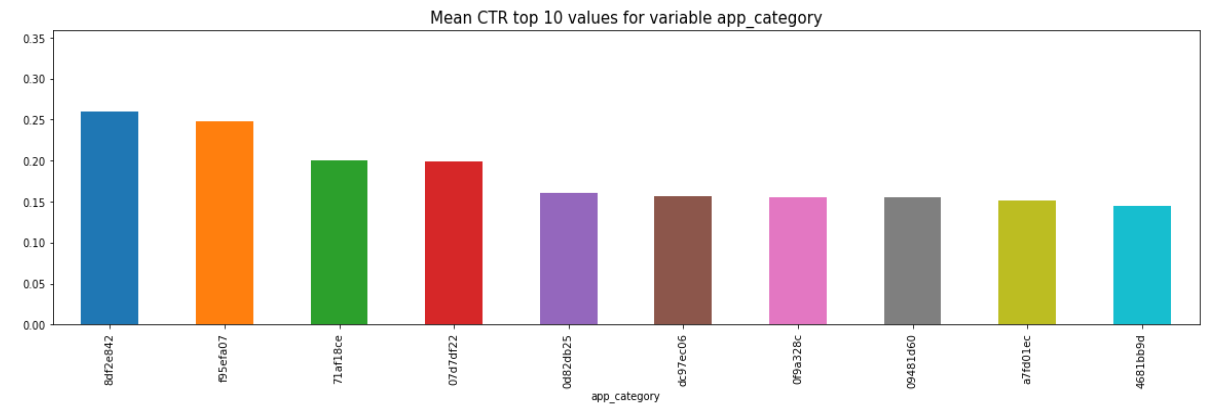
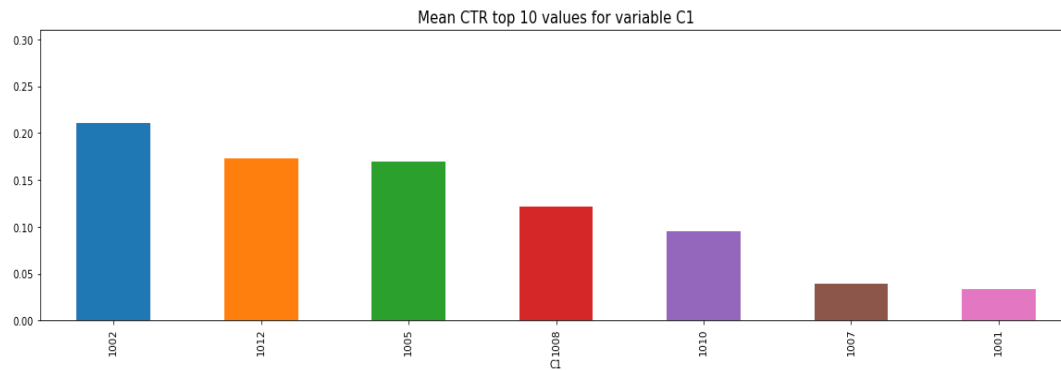
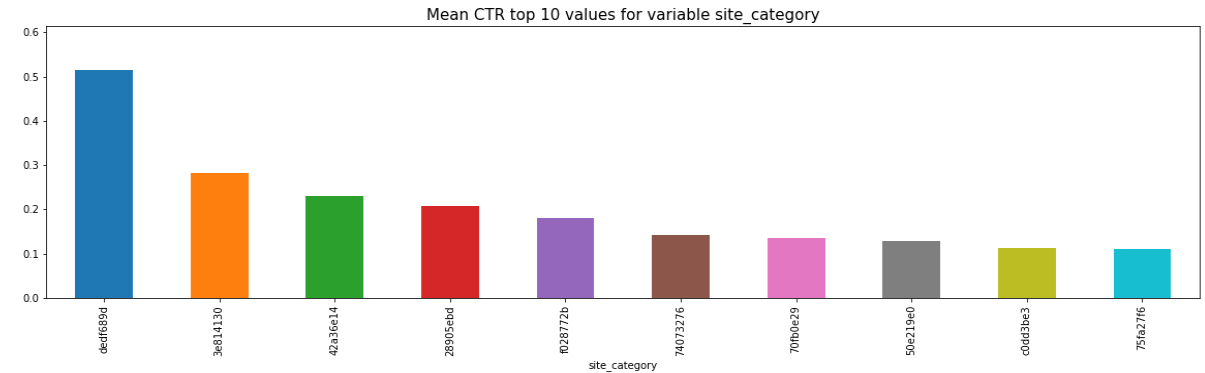
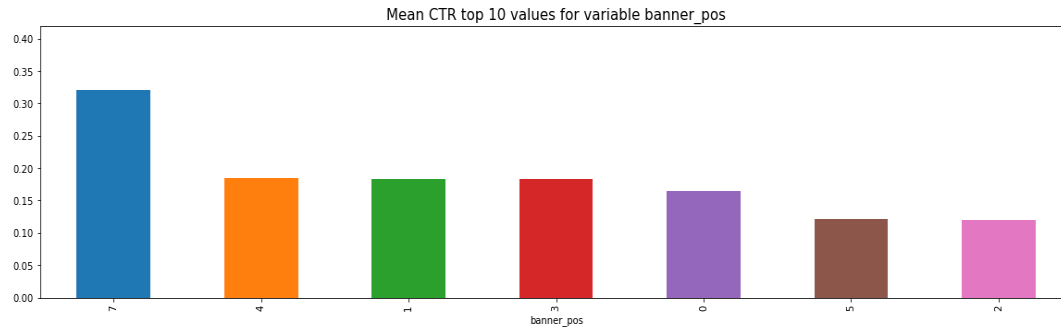
- 40 kk samples for train and 4 kk for test
- 22 categorical features

Feature space

- We are given with only categorical features with thousands of unique values

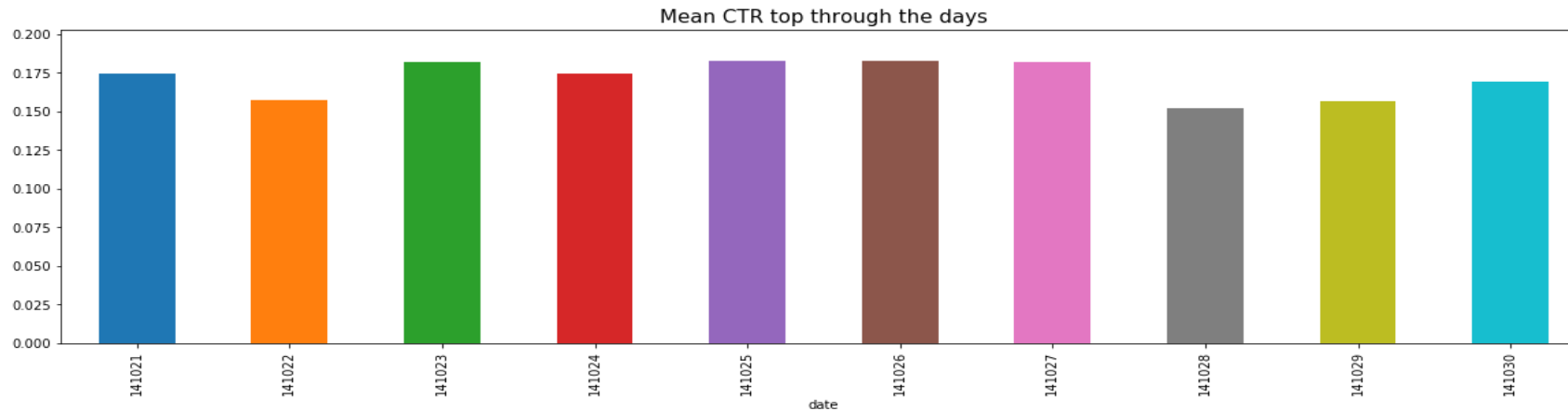
Data observation (1/3)

CTR average values differ by different categorical features

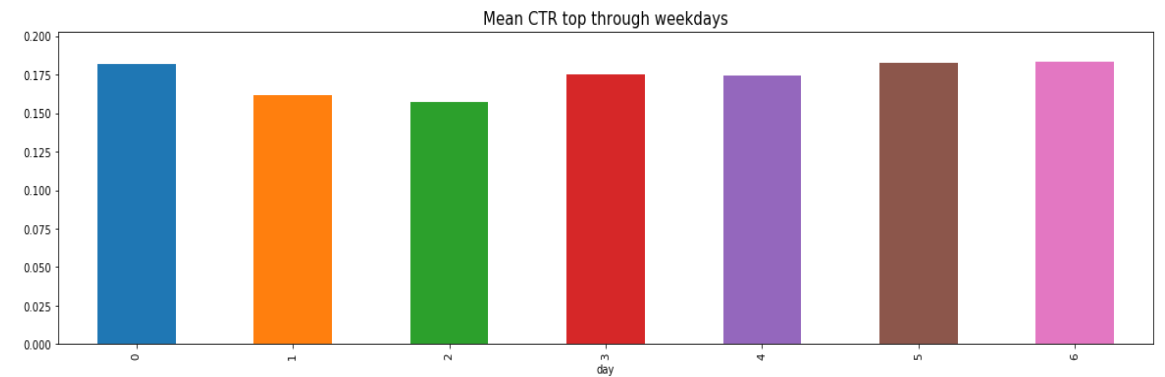
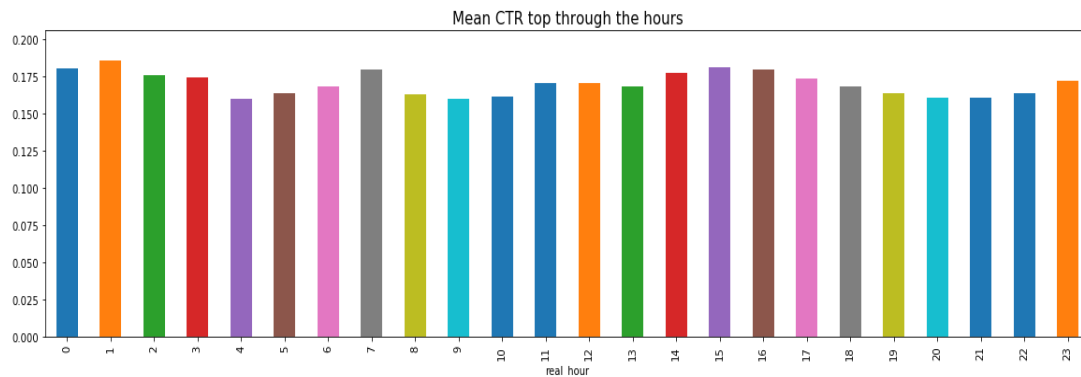


Data observation (2/3)

CTR average values through dates in train file

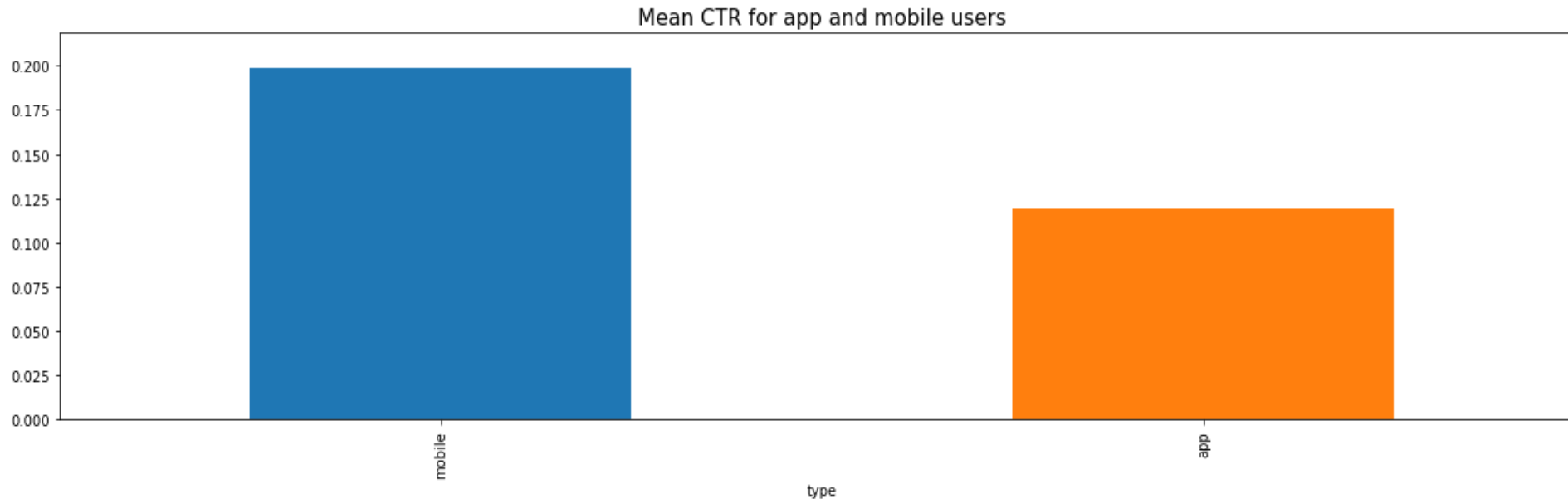


CTR seasonality by weekdays and hours



Data observation (3/3)

- 'site_id' == 85f751fd stands for app users
- Mobile and app users have a huge difference in average CTR



Data observation: Conclusion

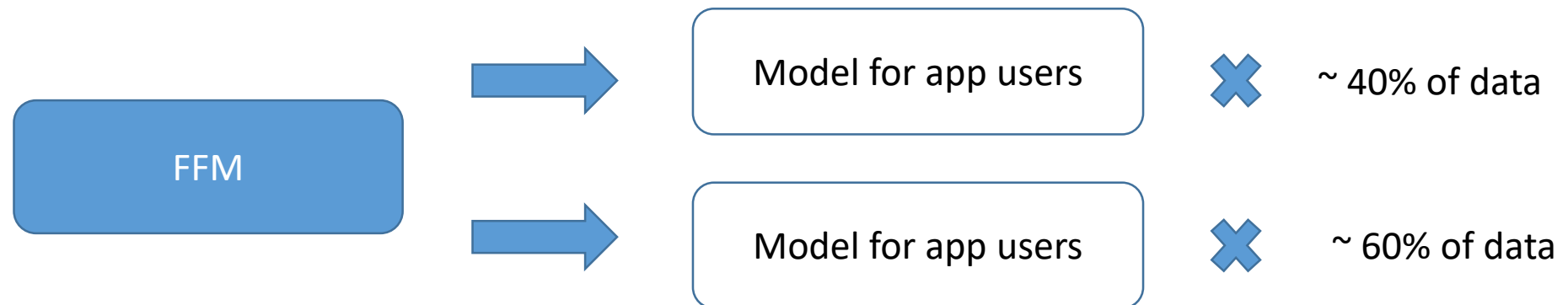
- Use factorization machines model as we have only categorical features with a lot of (> 2kk) unique values
- Since mobile and app users have a huge difference in average CTR, I will set up different models for them

Feature engineering

Feature	Comment
day	extracted from hour as a weekday number
time	extracted from hour
user	device_id + device_model + device_ip
user_count_hour	how many times times user appeared per hour
user_count_date	how many times times user appeared per date
user_nunique_hour_site_category	how many times site_category user visited per hour
user_nunique_hour_app_domain	how many times app_domain user visited per hour
user_nunique_hour_C15	how many times C15 user visited per hour
user_nunique_hour_C16	how many times C16 user visited per hour
user_nunique_hour_C17	how many times C17 user visited per hour
user_nunique_hour_C19	how many times C19 user visited per hour
user_nunique_hour_C21	how many times C21 user visited per hour
user_nunique_date_site_category	how many times site_category user visited per day
user_nunique_date_app_domain	how many times app_domain user visited per day
user_nunique_date_C15	how many times C15 user visited per day
user_nunique_date_C16	how many times C16 user visited per day
user_nunique_date_C17	how many times C17 user visited per day
user_nunique_date_C19	how many times C19 user visited per day
user_nunique_date_C21	how many times C21 user visited per day
place_id	site_id + app_id
place_genre_id	site_id + app_id + site_category + app_category
tech_position	banner_pos + device_conn_type
add_position	place_id + banner_pos
union_category	site_category + app_category
ultra_C_type	C1 + C14 + ... + C21
user_history	cumulative sum of visits of user per day
place_history	cumulative sum of visits of place per day

Models structure

!!! Both models were trained only on 40% of data due to technical reasons



Models based on LibFFM¹

Model for app users was trained with 10 latent features and gave local CV around 0.372 logloss

```
C:\Users\user>"D:\Downloads\avazu_feedzai\libffm-ftrl-master\libffm-ftrl-master\ffm-train.exe" -p "D:\Downloads\avazu_feedzai\ffm_txt\val_app_ffm.txt" -s 4 -k 10 -t 200 --no-rand --on-disk --auto-stop "D:\Downloads\avazu_feedzai\ffm_txt\train_app_ffm.txt"
iter   tr_logloss   va_logloss
  1      0.29883    0.37209
  2      0.27931    0.38885
Auto-stop. Use model at 1th iteration.
```

Model for mob users was trained with 6 latent features and gave local CV around 0.412 logloss

```
Командная строка - D:\Downloads\avazu_feedzai\libffm\ffm-train.exe -p D:\Downloads\avazu_feedzai\val_ffm.txt -s 6 -k 6 -l 2e-05 -t 200 -r 0.2 --auto-stop D:\Downloads\avazu_feedzai\train_ffm.txt
Microsoft Windows [Version 10.0.16299.192]
(c) Корпорация Майкрософт (Microsoft Corporation), 2017. Все права защищены.

C:\Users\user>D:\Downloads\avazu_feedzai\libffm\ffm-train.exe -p D:\Downloads\avazu_feedzai\val_ffm.txt -s 6 -k 6 -l 2e-05 -t 200 -r 0.2 --auto-stop D:\Downloads\avazu_feedzai\train_ffm.txt
iter   tr_logloss   va_logloss
  1      0.44178    0.43197
  2      0.43134    0.42887
  3      0.42875    0.42664
  4      0.42669    0.42475
  5      0.42496    0.41285
  6      0.42314    0.42439
Auto-stop. Use model at 1th iteration.
```

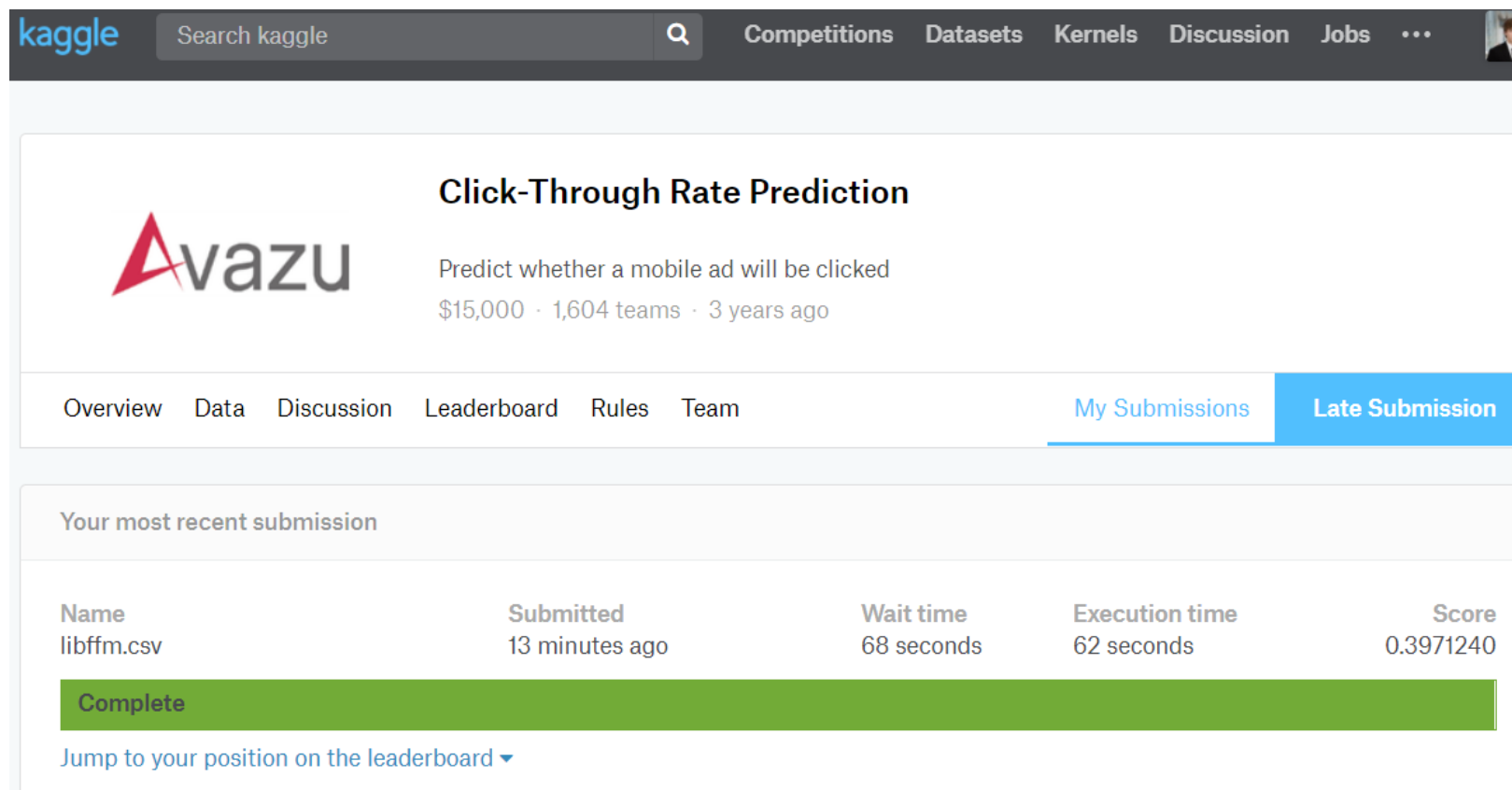
CV validation

- Same features were generated separately for mobile and app users
- Test size was set as 20%

¹ Library can be found at <https://github.com/CNevd/libffm-ftrl>

Results¹

Private LB 0.397 Logloss (~990 place)



The screenshot shows the Kaggle interface for the "Click-Through Rate Prediction" competition by Avazu. The page title is "Click-Through Rate Prediction" with a subtitle "Predict whether a mobile ad will be clicked" and details "\$15,000 · 1,604 teams · 3 years ago". The navigation bar includes "Overview", "Data", "Discussion", "Leaderboard", "Rules", "Team", "My Submissions", and "Late Submission". The "My Submissions" tab is active, showing a table of submissions. The most recent submission is "libffm.csv", submitted 13 minutes ago, with a wait time of 68 seconds, execution time of 62 seconds, and a score of 0.3971240. A green bar indicates the submission is "Complete". A link "Jump to your position on the leaderboard" is also visible.

Name	Submitted	Wait time	Execution time	Score
libffm.csv	13 minutes ago	68 seconds	62 seconds	0.3971240

Complete

[Jump to your position on the leaderboard](#)

¹ This result was obtained on 40% of training data. Using full set should improve accuracy

Area for improvement

- Parameter tuning (latent features and regularization)
- More feature engineering
- More data
- More models (additional splits by other categorical features besides site_id)

Thank you

- Code https://github.com/paveltr/avazu_late_submission
- My profile <https://www.linkedin.com/in/paveltr>