Data Science Game 2017 finals

lebed i 3 raka

Popov N., Shapovalov N., Soboleva D., Vikulin V.

Lomonosov Moscow State University

November 18, 2017

Overview

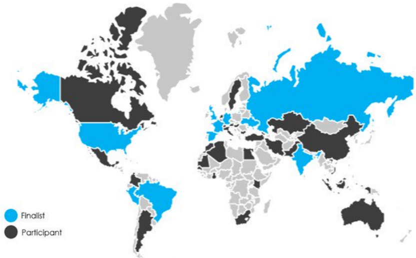
Contest info

- 150 teams
- each team contains 4 students from same university
- 20 teams advanced to the finals in Paris

OUR FINALISTS

WINNING THEIR TICKETS TO PARIS

1	Moscow State University	RUS
2	Higher School of Economics	RUS
3	Skoltech	RUS
4	IIMC	IND
5	Toulouse School of Economics	FRA
6	USP Sao Paulo	BRA
7	IMT Atlantique	FRA
8	Stevens Institute of Technology	USA
9	University of Edinburgh	GBR
10	University of Alfenas	BRA
12	Ukrainian Catholic University	UKR
14	Universidad Nacional de Ingenieria	PER
15	ENSIMAG	FRA
16	St Petersburg University	RUS
17	Université Toulouse Paul Sabatier	FRA
18	HSE NN	RUS
21	UPMC	FRA
23	Humboldt University	DEU
27	USP Sao Carlos	BRA
33	Barcelona Graduate School of Economics	ESP



Data

Data was given from January 2012 to May 2017

Product ID	Country ID	Date	...	Quantity
1	1	02.12.2016	...	1000
2	1	03.06.2012	...	125
2	3	11.09.2013	...	911

Table: input data format example

About 5.5m samples 50 original columns with features
About 38k unique pairs (product, country)

Goal

The **goal** of this challenge is to predict the demand in spare parts for different countries for the next 3 months (June, July, August 2017)

Product ID	Country ID	June 2017	July 2017	August 2017
1	1	100	90	80
2	1	0	0	0
2	3	1000	1	999

Table: output data format example

Metric is MAE overall predictions

Simple solutions

1. predict same quantity as in last month **public:** 11.48 **private:** 12.59
2. linear model on last 6 month **public:** 10.65 **private:** 11.68

1	—	lebed i 3 raka - MSU - RUSSIA	9.26568	39	2mo

Your Best Entry ↑

Your submission scored 9.27239, which is not an improvement of your best score. Keep trying!

2	▾2	Do u wanna tell about our god...	9.44430	55	2mo
3	▾4	TSEureka - TSE - France	9.45072	89	2mo
4	▴2	LSTeAm - Deeper is better - U...	9.45207	97	2mo
5	▾1	Imagouille - ENSIMAG - France	9.48983	95	2mo
6	▴3	ADASE - Skoltech - Russia	9.49184	41	2mo
7	▴2	UCUpnik - UCU - Ukraine	9.52370	45	2mo
8	—	Lab Rats - HSE NN - Russia	9.74954	73	2mo
9	▾2	TheFirstBrazilianSniper - Fed...	10.01052	37	2mo
10	▴1	Mean Predictors - Humboldt ...	10.21265	49	2mo
11	▾2	DataMafia - IIMC - INDIA	10.62162	48	2mo
12	—	Team Maia - USP São Paulo - ...	10.67895	80	2mo

linear model

Data preparation

1. Daily data transform to monthly data by counting statistics (sum, mean, median, std, count non zeros, etc.)
2. Fill in zeros for months without purchases
3. For each month generate 3 target variables
4. For each target fit different model
5. For validation we can use only one last month (May 2017)

Feature engineering

1. last 6 month, previous year
2. statistics by different time periods (3, 6, 9, 12, 24 month)
3. use extra information about items
    - properties of items are *not constant* over time
    - some of the properties are categorical
    - we defined property as mode value over time
4. In the end, we have about 300 features and 2m objects

Computational resources

- 500$ on Microsoft Azure
- 48 hours of usage
- GPU is not required

So we selected 4x machines with 112GB RAM and 16 vCPU

Regression

1. Simple linear regression (e.g. `sklearn.linear_model.ElasticNet`)
2. Tree-based methods:
   - `sklearn.ensemble.RandomForestRegressor`
   - `xgboost.XGBRegressor`
   - `lightgbm.LGBMRegressor` – rule of thumb

Feature selection

- Build simple l_1-reg linear regression for all data to obtain sparse coefficients
- Use only features with non-zero weights
- Not only eliminates some noise, but also speeds up training (especially when add categorical features)

Rocket science: non-linear transformations

- Want to minimize MAE, but conventional regressors minimize MSE
- Make some monotonic transformation of target to reshape its distribution
- Examples: `np.log(1 + target)`, `target ** 0.5`

Rocket science: time to classify

- The data is sparse (most of samples contain zero monthly demand), so try to exploit it!
- First of all, classification **zero** *vs.* **non-zero** demand goes
- For non-zero outcomes run regression to predict actual demand

Moar ideas

- Integer demand vs floating-point predictions: `round`
- For tree-based regressors build models with varying `colsample` and `random_state`
  Make single model from them (simple averaging?)
- Running out of ideas? Combine your submissions!

Some additional thoughts

- Months have different number of days in them; consider it in model
- Good competition on Kaggle with MAE-based regression: Allstate Claims Severity[1]

[1] https://www.kaggle.com/c/allstate-claims-severity

Result-driven programming

transfrom type	feature selection	clf	round	score
`np.log(1+y)`	−	−	−	9.66
`np.log(1+y)`	−	−	+	9.60
`np.log(1+y)`	−	+	+	9.59
`y ** 1/3`	?	+	+	9.40
`np.log(1+y)`	+	+	+	**9.27**

Table: Public LB score for different models. Averaging some models with best public LB gave **9.26**
Note: feature selection implies inclusion of categorical features in model, and vise versa

Aftermath

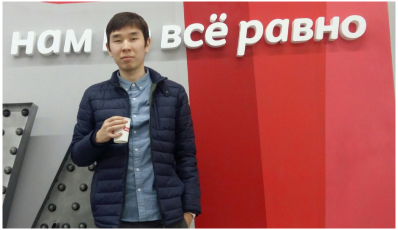
transfrom type	**feature selection**	**clf**	**round**	**public LB**	**private LB**
`np.log(1+y)`	$-$	$-$	$-$	9.66	10.59
`np.log(1+y)`	$-$	$-$	$+$	9.60	10.53
`np.log(1+y)`	$-$	$+$	$+$	9.59	10.29
`y ** 1/3`	?	$+$	$+$	9.40	**9.97**
`np.log(1+y)`	$+$	$+$	$+$	**9.27**	10.05

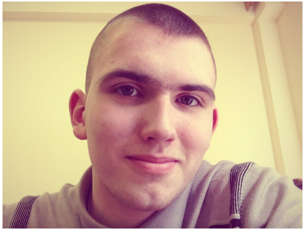
Table: LB scores for different models. Averaging some models with best public LB gave **10.04** - top-1
Note: feature selection implies inclusion of categorical features in model, and vise versa

Timeline

- Sep 29, 8 AM: start of the game
- Sep 29, 12 PM: some nontrivial submission (**11.48**)
- Sep 29, 5 PM: fix problem with overfit on single regression model (**10.9**)
- Sep 29, 7 PM: log-transform + classification + `round` (**9.60**)
- Sep 30, 12 AM: categorical features + feature selection (**9.27**)
- Sep 30, 2 AM - 4 AM: some random nonlinear transforms (**9.40**)
- Sep 30, 5 AM - 8 AM: `colsample` aggregation + averaging best models (**9.26**)
- Sep 30, 9 AM - 10 AM: prepare small presentation for jury
- Sep 30, 12 PM: end of the game

Special thanks to our sponsors





Questions?