kaggle Search kaggle Competitions Datasets Kernels Discussion Jobs

Featured Prediction Competition

Corporación Favorita Grocery Sales Forecasting

Can you accurately predict sales for a large grocery chain?

$30,000
Prize Money

Corporación Favorita · 1,675 teams · 8 days ago

Overview Data Kernels Discussion Leaderboard Rules Team My Submissions New Topic

antklen
12th place

12th place solution

posted in Corporación Favorita Grocery Sales Forecasting 7 days ago

4

Code with main parts of the solution is here https://github.com/antklen/kaggle_favorita-grocery-sales.

Feature engineering

I used the same features as in public scripts, namely

- mean sales during several last periods for item/store pairs
- mean sales during several last periods for item/store pairs for each day of week
- mean ompromotion during several last periods for item/store pairs but as for time windows, I used more history - up to 1 year

plus some other grouping averages

- mean sales during several last periods for item/store pairs for each day of month
- mean number of zero sales days during several last periods for item/store pairs
- mean sales during several last periods for store/item family
- mean sales during several last periods for store/item class
- mean sales during several last periods for item/store city

- mean sales during several last periods for item/store stat
- mean sales during several last periods for item/store type
- mean sales during several last periods for item/store cluster
- mean sales during several last periods group by only item
- mean number of zero sales days during several last periods group by only stores

plus some differences between mean features with different time window (for example, mean_7days - mean_28days)

plus linear regression on each item/store pair time series (for incorporating possible time trend)

plus mean promo for all 16 days (mean of promo_0, promo_1, ... in Ceshine Lee's script).

as for categorical features (store and item info - family, type, etc.)

- standard label encoding for lightgbm
- embeddings for neural network

To be honest, base mean, mean by day of week and promo features do almost all the job, my additional features added not much value.

Onpromotion issue

As for problem with missing onpromotion data, I didn't come up with any clever solution. But taking as less promo based features as possible helped on public LB and helped to have more stable relation between validation and LB (but still not very good). Firstly, I made some extra features based on promo and got stuck around public scripts, but when I removed them, I was able to climb up on the leaderboard.

Training and validation scheme

For validation nothing fancy, I used just the same scheme as in Ceshine Lee's script - 26.07-10.08 for validation, several previous 16-days periods for train. May be it would be useful to have several periods for validation (Giba and CPMP reported that it helped), but we had a lot of data here and training took a lot of time, so I decided to stay with only one period.

What helped to improve score is just to take more data for training. Switching from 6 16-days periods to 20-25 periods helped. During last several days I was running the same models but for even longer training history - 52, 75 and 104 periods (1 year, 1.5 year and 2 year for training). Taking more history consistently improved validation score, taking too long (104 periods) made public score worse, but really surprisingly on private LB even 52 periods was too much - 25 periods was enough, I have the best private LB score on single models with 25 periods.

Models

I used LightGBM and neural network with categorical embeddings on the same set of features.

Also I used two different approaches:

1. Train one general model for all lags (very memory-consuming=)
2. Train separate model for each lag. But in a little different way than in public scripts. I didn't use the same set of features for all lags. For each lag I constructed a little different set of features on-the-fly. The motivation for that is that I thought that it's not a good idea to feed into the model all day of week mean features when we train model only for one day of week. For example, if we train model for monday, why we add features for all other weekdays? It probably adds some noise.

Switching to these two schemes from scheme used in public scripts really improved my score and led me to the top.

Final models

So in the end I had 4 types of models (lightgbm or neural network + general model for all lags or separate model for each lag), trained on several training sets with different length. My final models were simple averages of some of these models.

Averaging models helped a lot here. Public and private score of my single models have very bad correlation. The best single public model (0.507) has only 0.519 on private, while best private model (0.514) had 0.510 on public. But my blending submissions are much more consistent. All my last blending submissions (about 10 submissions) are in the interval [0.506, 0.508] on public and in the interval [0.513, 0.516] on private.

Options

Comments (2)

Sort by Hotness

Click here to enter a comment...

CPMP • 6 days ago • Options • Reply

0

Thanks for sharing and congrats on the gold medal!

antklen • (12th in this Competition) • 5 days ago • Options • Reply

0

Thanks and congrats on one more step towards Grandmaster=)

Our Team Terms Privacy Contact/Support