Porto Seguro’s Safe Driver Prediction

3rd place solution

Dmitry Altukhov

Problem statement

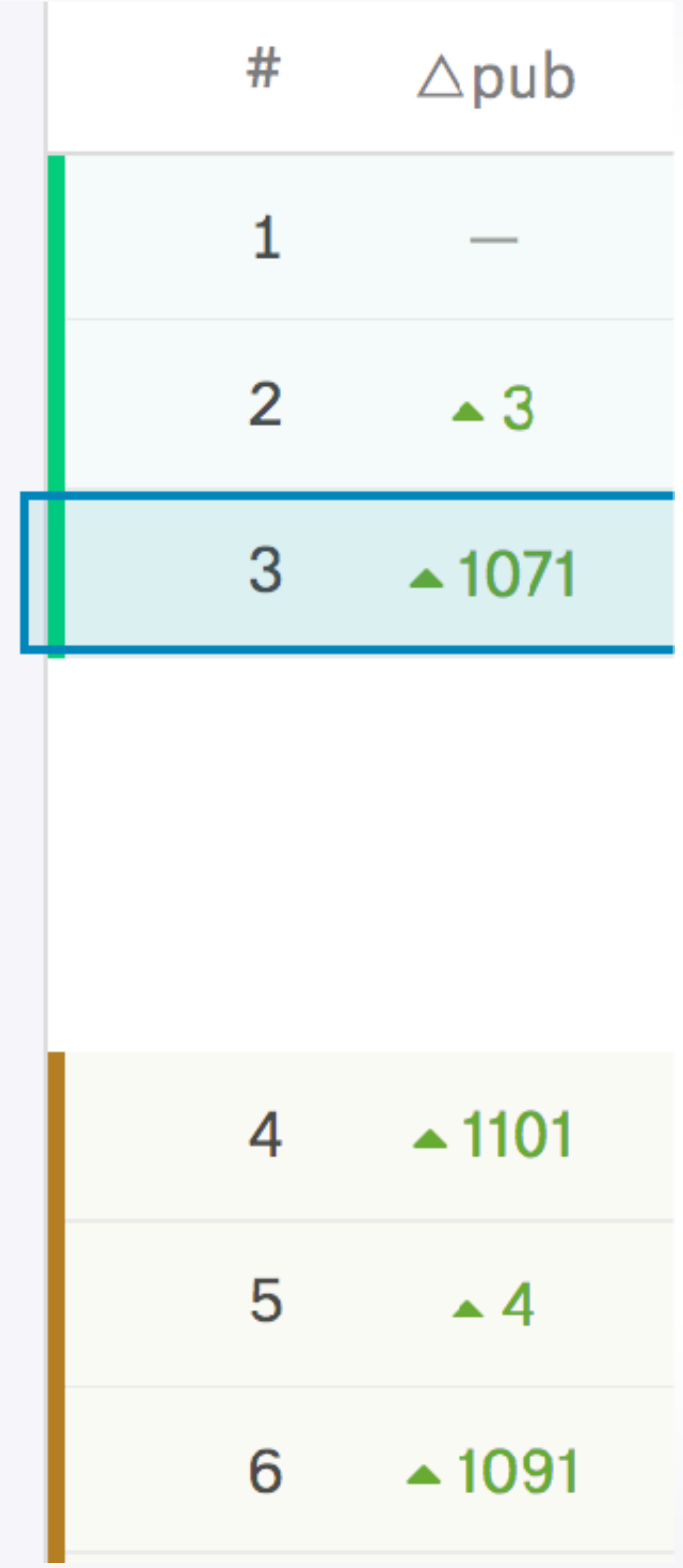
- Imbalanced (~3-4%) binary classification with metric
  - GINI = 2 * AUC - 1
- Semi-anonymized 57 features (could be car model, price, etc.)
- Very similar distributions between train and test
- ~600k rows in train, ~900k rows in test, random 30/70 public/private split

Solution overview

- Results are very close to the baseline. Because of that, participant’s scores are very close => last digit wars on the leaderboard.
- A lot of noisy features. Some of them allegedly automatically generated.
- Important to remove features and one-hot-encode categorical variables.
- Regularized and stable models. 1 LightGBM and 1 neural network.

Validation

- Tune parameters, select features and generate final out-of-fold predictions on different cv splits/ holdouts in order to avoid overfitting.
- Average 4-8 runs with different random seeds to stabilize the results.
- Easy to overfit to particular cv split and/or public leaderboard => better to avoid public scripts.

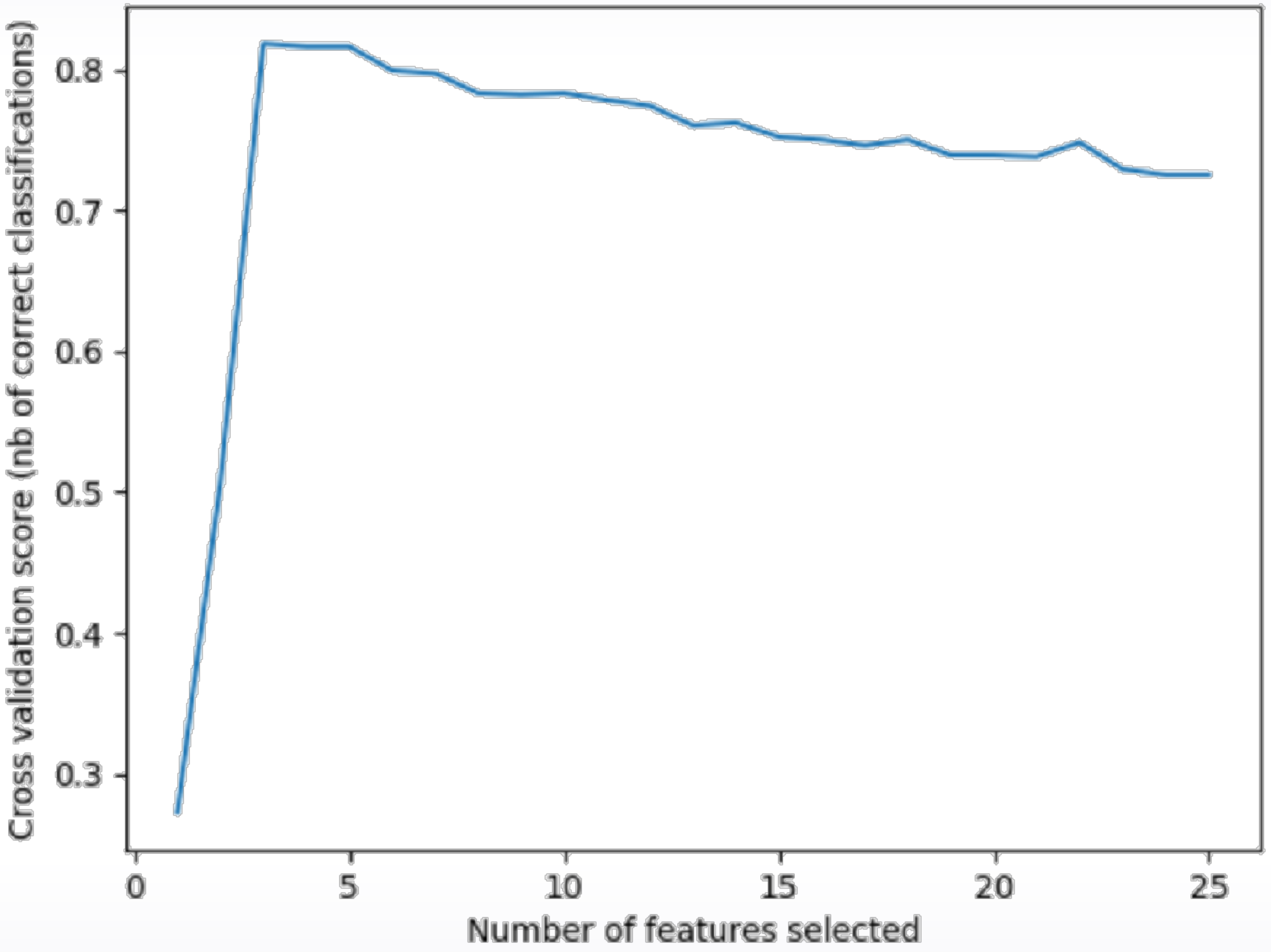


#	△pub
1	—
2	▲3
3	▲1071
4	▲1101
5	▲4
6	▲1091

Feature elimination

- Drop all features with ‘**calc**’ prefix. They seem to be randomly generated.
- Recursively eliminate features until cv score stops improving .

Sample plot from sklearn



Models

0.5 * Boosted trees

- Hot-encode categorical features.
- Regularized parameters: lambda_l1:10, bagging_fraction: 0.5, num_leaves: 16

LightGBM

+

0.5 * Neural Network

- Hot-encode categorical AND numerical features with low number of unique values.
- Regularized architecture: 4096-1024-256 with 0.5 dropout inbetween, first layer has only 2% of nonzero weights

PYTORCH

What didn’t work

- Feature engineering.
- Huge ensembles.

Interesting stuff

- Problem looks like anomaly detection, i.e. rare and unique examples have higher probability of being of class 1.
- Sample-wise reconstruction error of auto-encoder trained on train+test data gives ~0.6 AUC which is pretty high for completely unsupervised method.
- First place solution has an edge because of very good denoising auto-encoder: neural networks are trained on it’s hidden states.

Sellout

- Check our coursera course on competitive data science
- https://www.coursera.org/learn/competitive-data-science

Thank you!