Project Тачки 2

Prediction of car prices on a marketplace

Participants:

Быков Егор

Махаев Дмитрий

Некрасова Мария

Павловская Екатерина

Серочкин Егор

Relevance of the study

Under the conditions of sanctions pressure from Western partners, official sales of foreign car manufacturers in Russia were suspended, which caused unprecedented demand on the car resale market.

That is why we consider car price prediction a highly urging problem

Overview

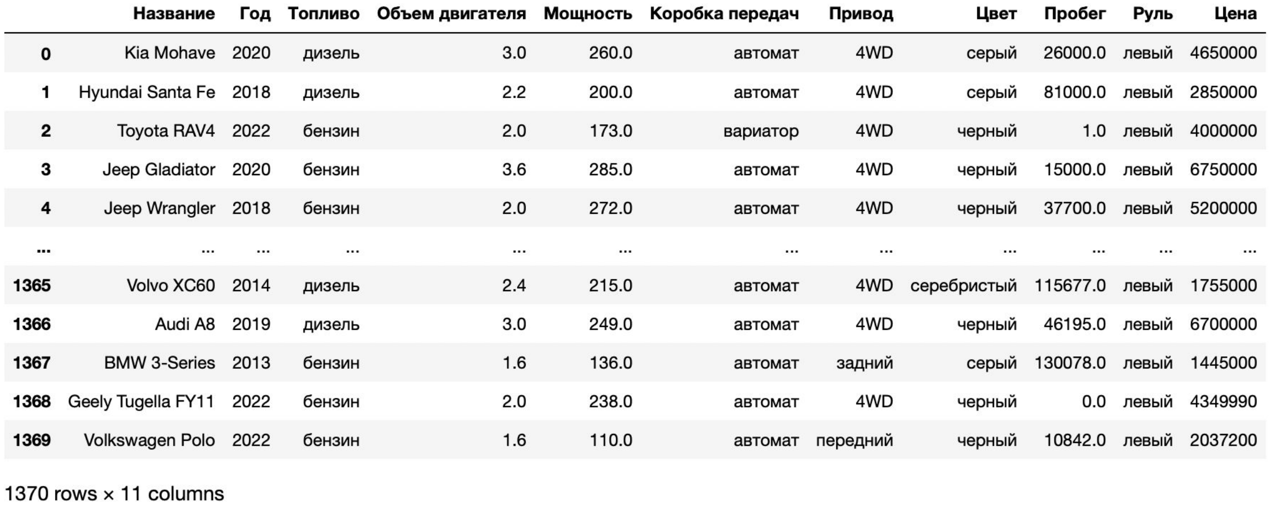
- Data, where it came from
- How we processed data
- Chosen metric
- Models used
  - Versions of Linear regression
  - Trees: random forest, boosting
  - Neural network

A little bit about our data

It was parsed from Russian online market drom.ru in May 2023

Later on, we modified the data so that it could be used for Machine Learning:

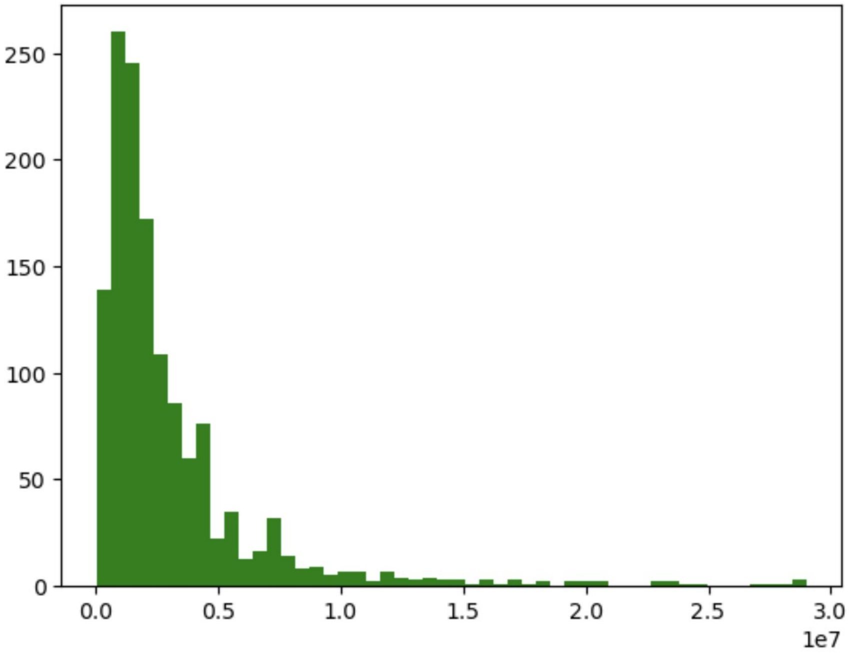
- OneHotEncoding was used
- NaNs’ replaced
- New features were added: country of manufacture
- Normalization with log



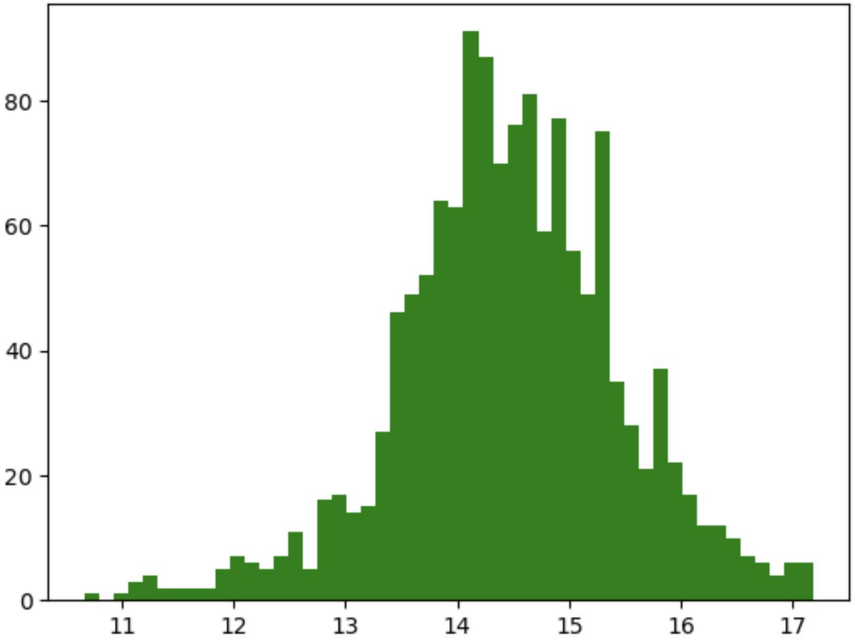
	Название	Год	Топливо	Объем двигателя	Мощность	Коробка передач	Привод	Цвет	Пробег	Руль	Цена
0	Kia Mohave	2020	дизель	3.0	260.0	автомат	4WD	серый	26000.0	левый	4650000
1	Hyundai Santa Fe	2018	дизель	2.2	200.0	автомат	4WD	серый	81000.0	левый	2850000
2	Toyota RAV4	2022	бензин	2.0	173.0	вариатор	4WD	черный	1.0	левый	4000000
3	Jeep Gladiator	2020	бензин	3.6	285.0	автомат	4WD	черный	15000.0	левый	6750000
4	Jeep Wrangler	2018	бензин	2.0	272.0	автомат	4WD	черный	37700.0	левый	5200000
...	...	...	...	...	...	...	...	...	...	...	...
1365	Volvo XC60	2014	дизель	2.4	215.0	автомат	4WD	серебристый	115677.0	левый	1755000
1366	Audi A8	2019	дизель	3.0	249.0	автомат	4WD	черный	46195.0	левый	6700000
1367	BMW 3-Series	2013	бензин	1.6	136.0	автомат	задний	серый	130078.0	левый	1445000
1368	Geely Tugella FY11	2022	бензин	2.0	238.0	автомат	4WD	черный	0.0	левый	4349990
1369	Volkswagen Polo	2022	бензин	1.6	110.0	автомат	передний	черный	10842.0	левый	2037200

1370 rows × 11 columns

Normalization



Price distribution



log(Price) distribution

Additional data processing

As the features we decided to use

- Colour
- Engine volume
- Years old
- Mealige
- Transmission
- Gear
- Wheel side
- Fuel type

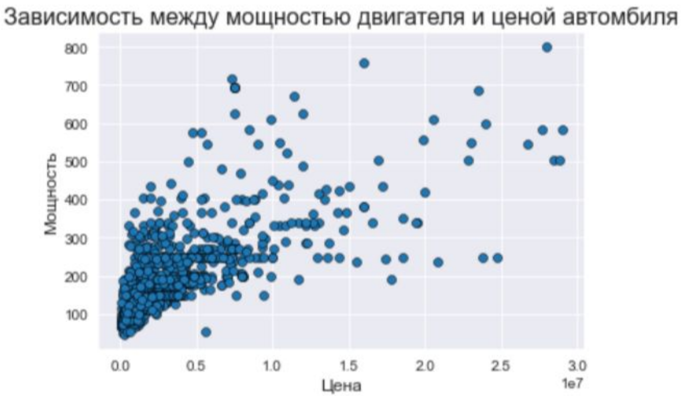
Chosen metric: RMSE

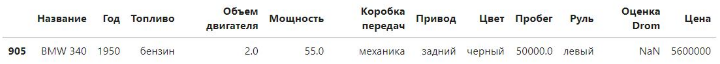
Why?

- Models were trained on MSE, so it’s good to compare them on RMSE
- Also it is easy to interpret.

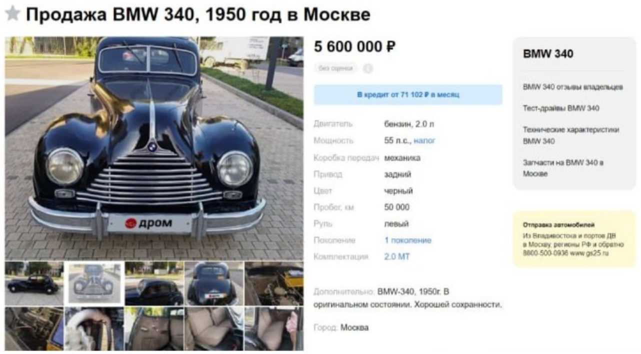
$$\boldsymbol{RMSE = \sqrt{MSE} = \sqrt{\frac{1}{n}\sum(y_i - \hat{y}_i)^2}}$$

Some Data Visualization





	Название	Год	Топливо	Объем двигателя	Мощность	Коробка передач	Привод	Цвет	Пробег	Руль	Оценка Drom	Цена
905	BMW 340	1950	бензин	2.0	55.0	механика	задний	черный	50000.0	левый	NaN	5600000



★ Продажа BMW 340, 1950 год в Москве

5 600 000 ₽

без оценки

В кредит от 71 102 ₽ в месяц

Двигатель бензин, 2.0 л
Мощность 55 л.с., налог
Коробка передач механика
Привод задний
Цвет черный
Пробег, км 50 000
Руль левый
Поколение 1 поколение
Комплектация 2.0 MT

Дополнительно: BMW-340, 1950г. В оригинальном состоянии. Хорошей сохранности.

Город: Москва

BMW 340

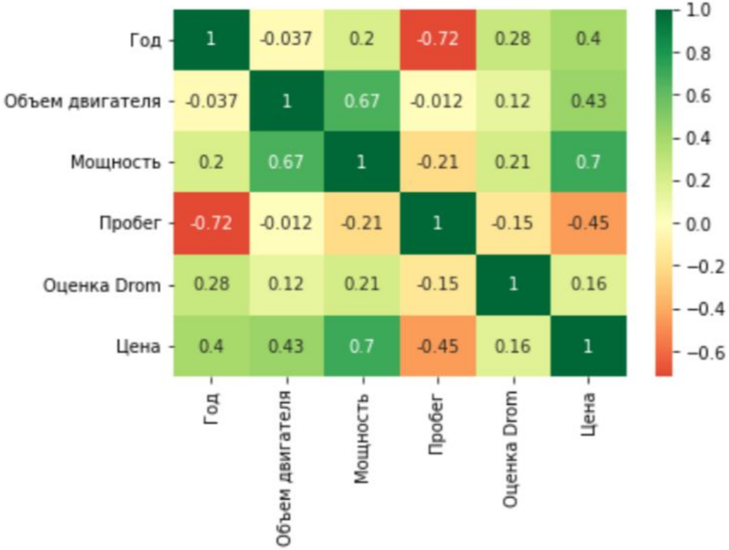
BMW 340 отзывы владельцев

Тест-драйвы BMW 340

Технические характеристики BMW 340

Запчасти на BMW 340 в Москве

Отправка автомобилей
Из Владивостока и портов ДВ в Москву, регионы РФ и обратно
8800-500-0936 www.gs25.ru



Linear models

Hyperparameters and Results

	Lasso	Ridge	ElasticNet	SGDRegressor
Alpha	0.9	0.9	0.1	1.0
l1_ratio			0.9	
Learning_rate				Adaptive
Penalty				l1

ElasticNet	Ridge	Lasso	Linear Regression	SGDRegressor
2`217`493.24	2`225`910.85	2`227`131.85	2`227`133.11	2`238`256.24

Naive prediction: 3`452`207.49

Decision Tree

Decision Tree

Decision Tree

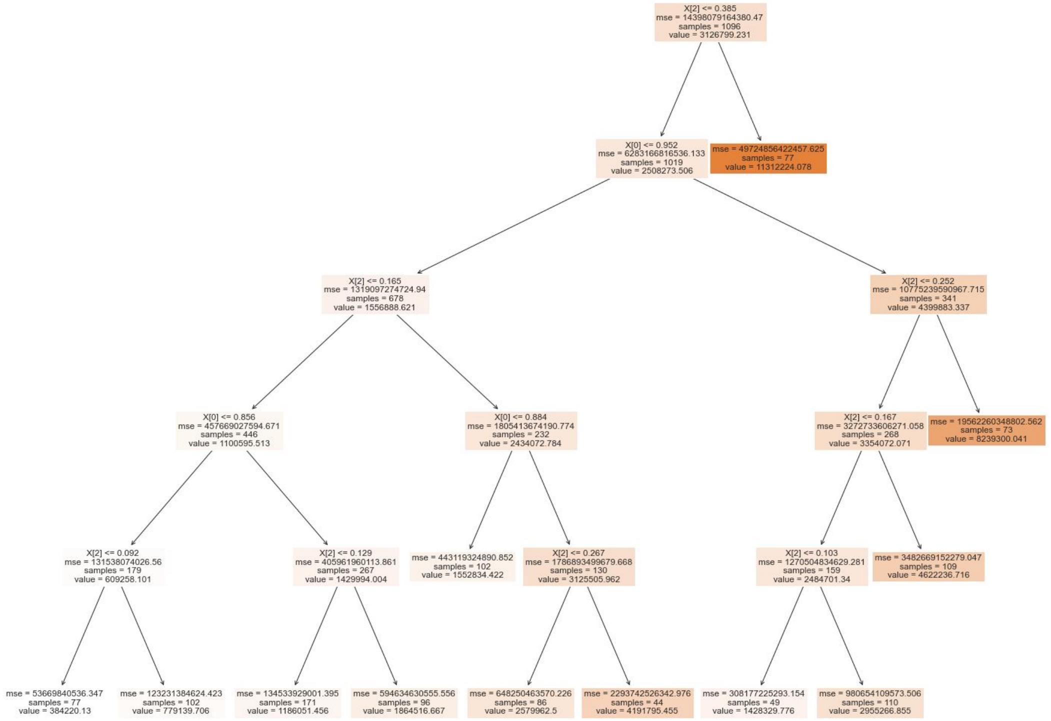
MAE: 1069473.2180

RMSE: 2156379.3430

MAPE: 0.4729

Test Score: 4649971870950.092

Train Score: 5527470174446.3955



Boosting

MAE: 791023.9096

RMSE: 1630064.2893

MAPE: 0.3107

Test Score: 2657109587124.939

Train Score: 989612975465.5034

Random Forest

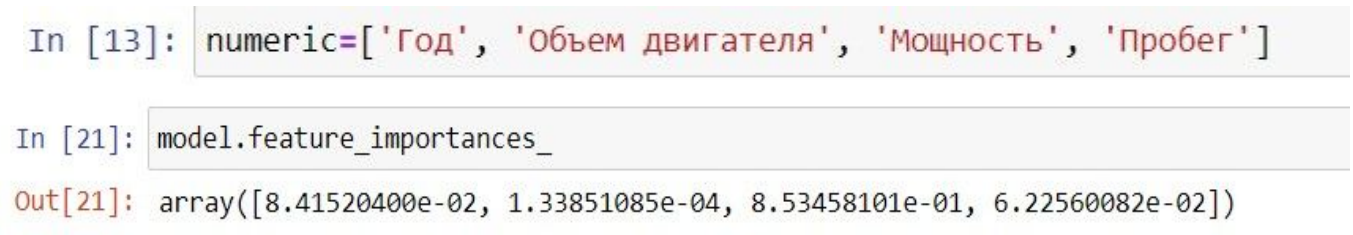
MAE: 1418631.0895

RMSE: 2605073.8150

MAPE: 0.7861

Test Score: 6786409581540.967

Train Score: 8702858594536.699



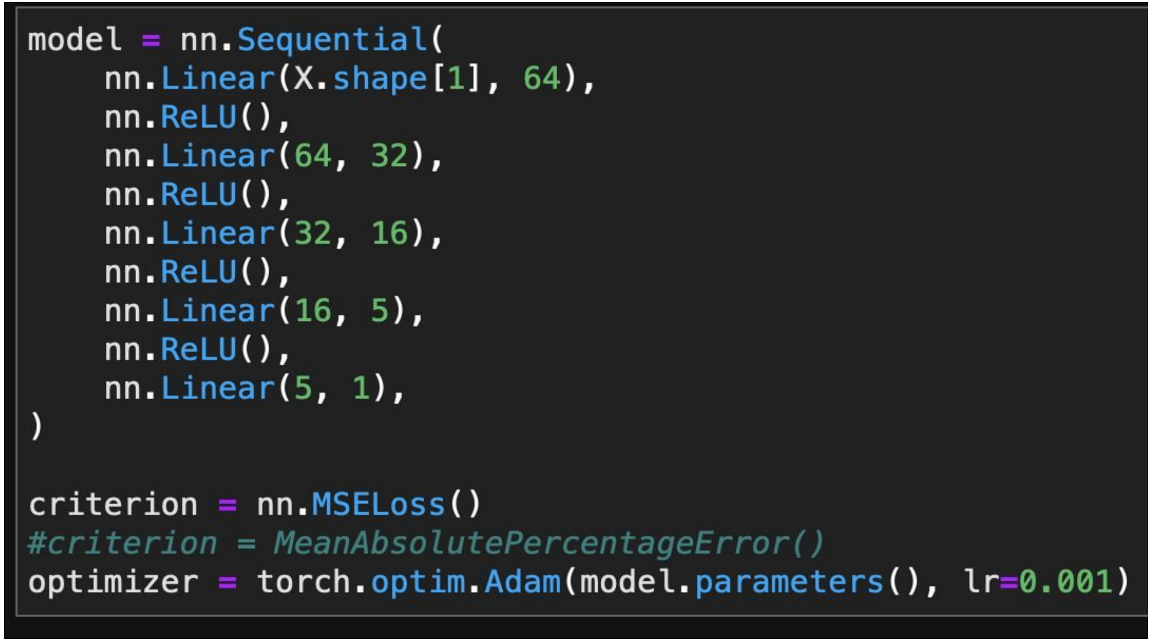
```
In [13]: numeric=['Год', 'Объем двигателя', 'Мощность', 'Пробег']

In [21]: model.feature_importances_

Out[21]: array([8.41520400e-02, 1.33851085e-04, 8.53458101e-01, 6.22560082e-02])
```

The “engine power” feature has the greatest impact on the predicted attribute “price”§

Neural network

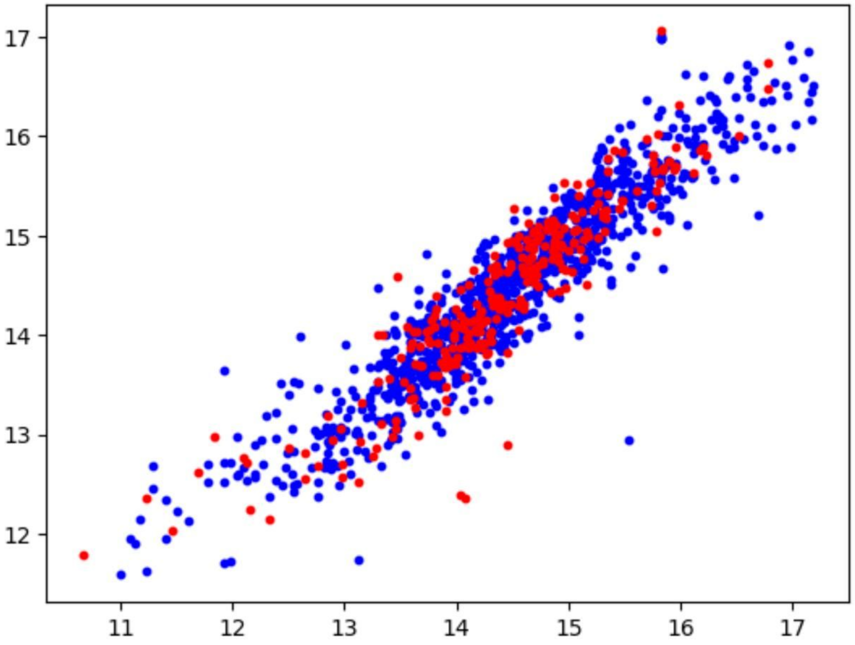
Neural network

- 5 fully connected layers & ReLU activation function
- Training without logarithmization is highly unstable
- All numerical variables converted to logarithms
- Not too much data, so no batching was used
- Overfitting was avoided with picking low learning rate and fine selection of training length

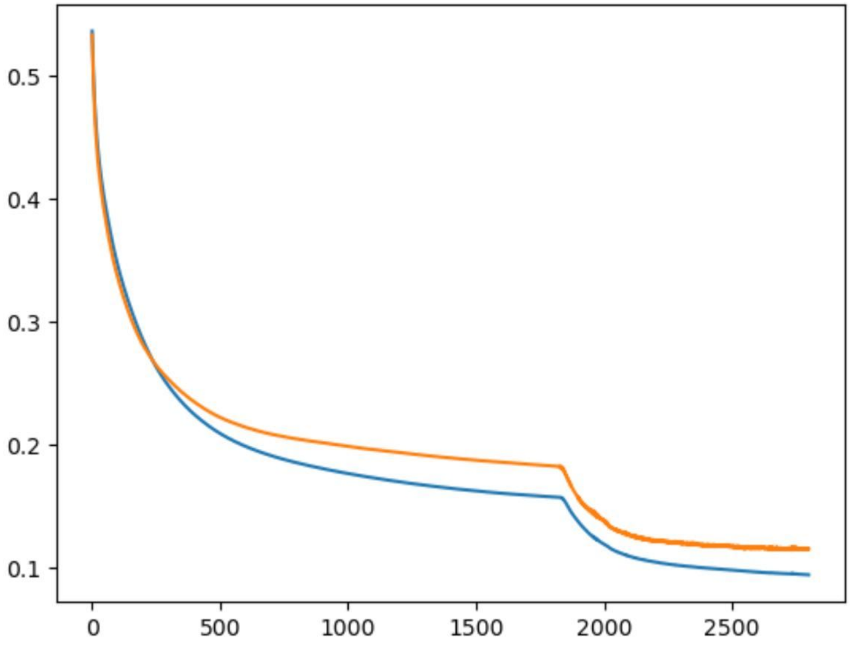


```
model = nn.Sequential(
    nn.Linear(X.shape[1], 64),
    nn.ReLU(),
    nn.Linear(64, 32),
    nn.ReLU(),
    nn.Linear(32, 16),
    nn.ReLU(),
    nn.Linear(16, 5),
    nn.ReLU(),
    nn.Linear(5, 1),
)

criterion = nn.MSELoss()
#criterion = MeanAbsolutePercentageError()
optimizer = torch.optim.Adam(model.parameters(), lr=0.001)
```

The learning process

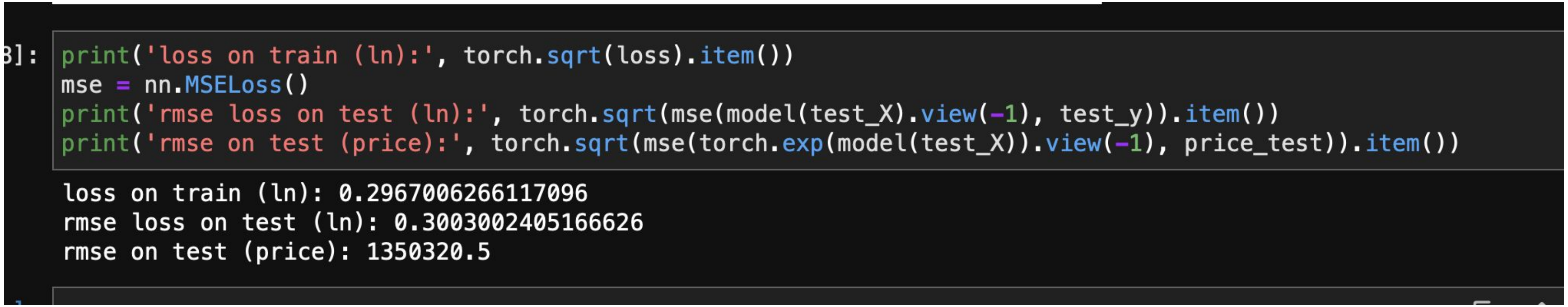


Graph: ln predictions x ln target
red — test



MSE Loss history. Blue — train, Yellow — test

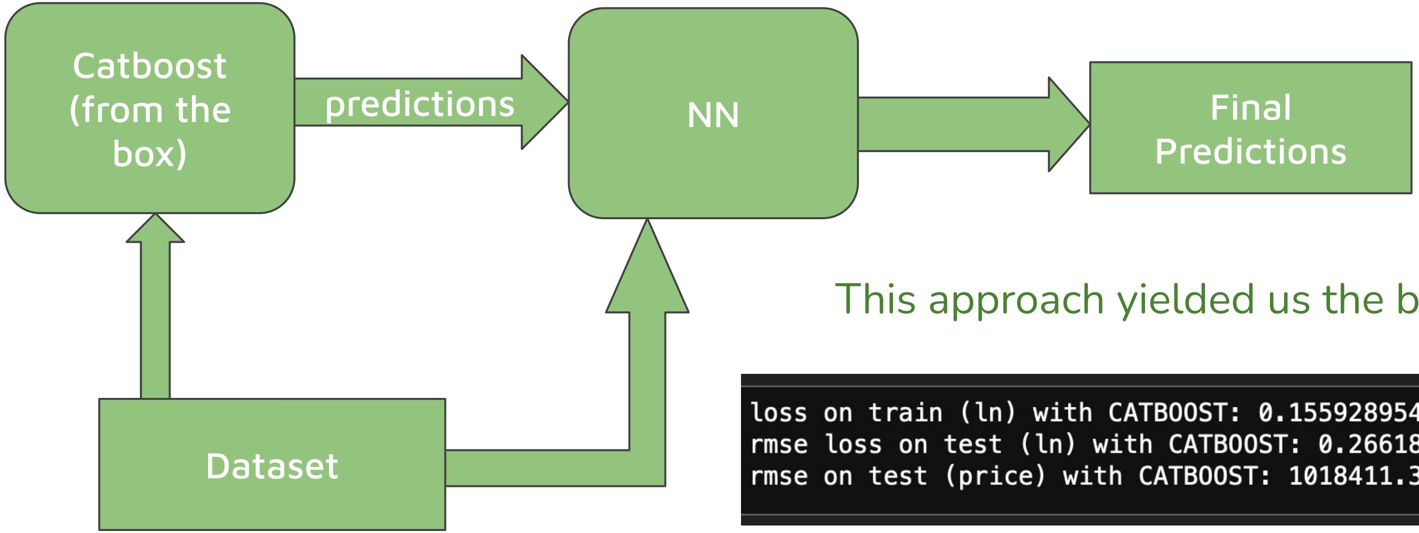
The results are the best so far



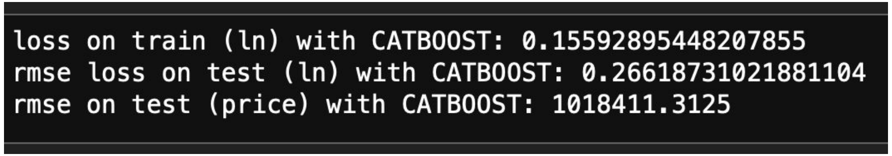
```
print('loss on train (ln):', torch.sqrt(loss).item())
mse = nn.MSELoss()
print('rmse loss on test (ln):', torch.sqrt(mse(model(test_X).view(-1), test_y)).item())
print('rmse on test (price):', torch.sqrt(mse(torch.exp(model(test_X)).view(-1), price_test)).item())
```

```
loss on train (ln): 0.2967006266117096
rmse loss on test (ln): 0.3003002405166626
rmse on test (price): 1350320.5
```

Using another model predictions as input to NN



This approach yielded us the best result



```
loss on train (ln) with CATBOOST: 0.15592895448207855
rmse loss on test (ln) with CATBOOST: 0.26618731021881104
rmse on test (price) with CATBOOST: 1018411.3125
```

MAE: 569,000
MAPE: 0.2