

Report: Predict Bike Sharing Demand with AutoGluon Solution

Vinlaw Mudehwe

Initial Training

What did you realize when you tried to submit your predictions? What changes were needed to the output of the predictor to submit your results?

Some of the columns in the train set were not available in the test set e.g casual and registered and therefore had to be dropped from the train set for more accurate model inference using the test set. Some of the columns were initially set as numerical data types but had to be changed to categorical data types in order for the algorithm to understand that their respective values should not be computed as raw numerical inputs but as representations of distinct categorical values.

What was the top ranked model that performed?

Of all the models, the one that ranked highest was WeightedEnsemble_L3

Exploratory data analysis and feature creation

What did the exploratory analysis find and how did you add additional features?

The EDA found that some columns (weather and season) were treated as numerical data types when instead, they had to be treated as categorical values. As such, these columns were changed to a categorical data type. Furthermore, the datetime column had to be formatted to datetime value and dummy columns were created to parse days, months, years and hours as separate data columns.

How much better did your model preform after adding additional features and why do you think that is?

There was no change to the kaggle score which remained at 1.79637. However, there was a significant change to the RMSE value of the highest performing model (WeughtedEnsemble_L3) which changed from 53.025270 to 30.090282. I assume this was likely because with the new created features and for the same amount of time_limit assigned (600s), the model performed better on the train_set and likely began to overfit hence the change in the RMSE without any changes to the kaggle score.

Hyper parameter tuning

How much better did your model preform after trying different hyper parameters?

The model performed better with regards to the kaggle score which reached a low score of 0.63238. For this, more time was given for the model to train (900s) compared to the previous 600s.

If you were given more time with this dataset, where do you think you would spend more time?

I would spend more time of trying different combinations of hyperparameters in order to improve model performance. I believe that there is a limit to the amount of EDA one can make on a dataset, but tuning hyperparameters especifiially for a mechanism like autogluon which trains multiple models at once is something that is very crucial in improving model performance and accuracy.

Create a table with the models you ran, the hyperparameters modified, and the kaggle score.

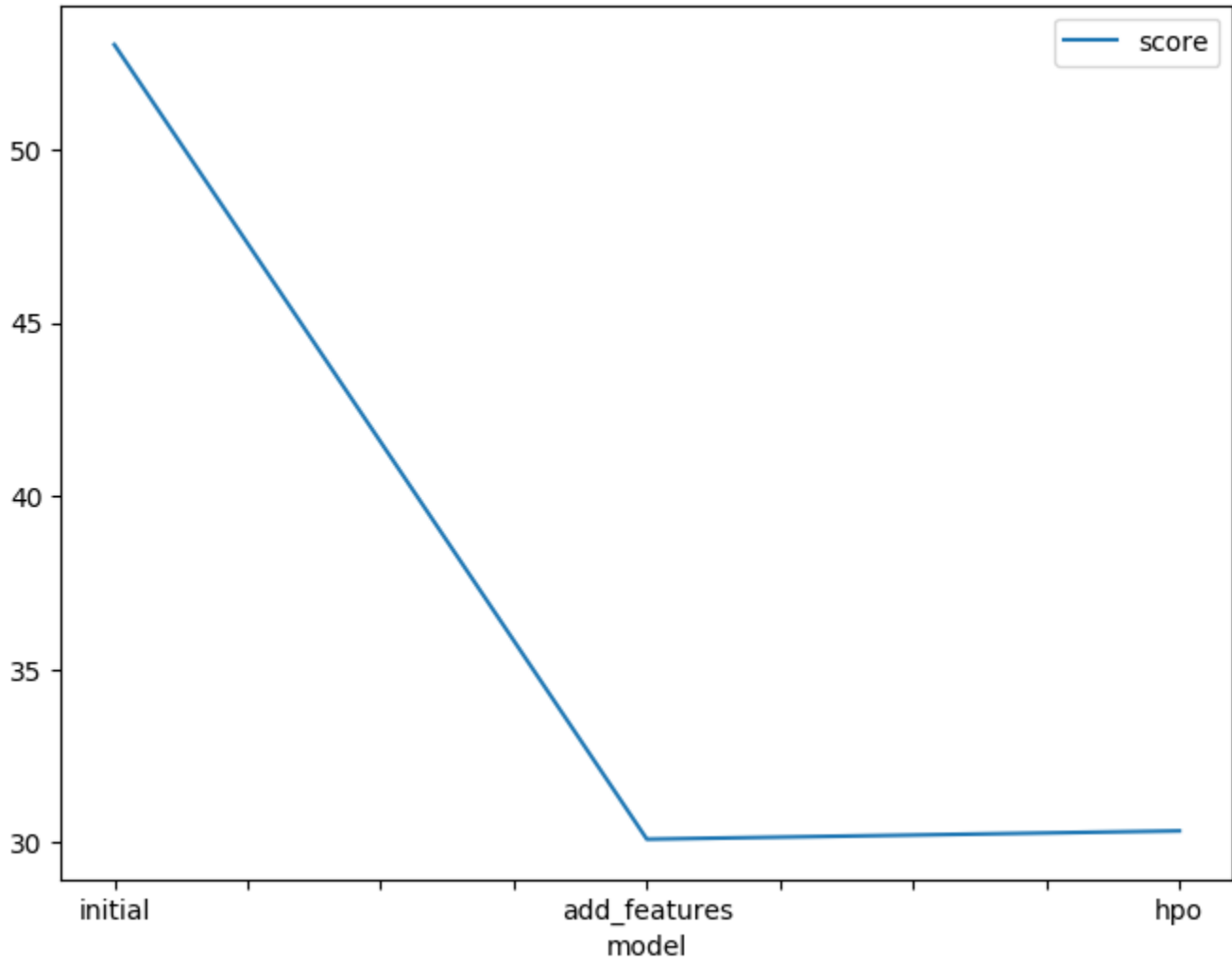
```
In [11]: pd.DataFrame({
    "model": ["initial", "add_features", "hpo"],
    "time_limit": [600, 600, 900],
    "presets": ["best_quality", "best_quality", "best_quality"],
    "hyperparameter_tune_kwargs": ["default", "default", "random"],
    "score": [1.79637, 1.79637, 0.63238]
})
```

Out[11]:

	model	time_limit	presets	hyperparameter_tune_kwargs	score
0	initial	600	best_quality	default	1.79637
1	add_features	600	best_quality	default	1.79637
2	hpo	900	best_quality	random	0.63238

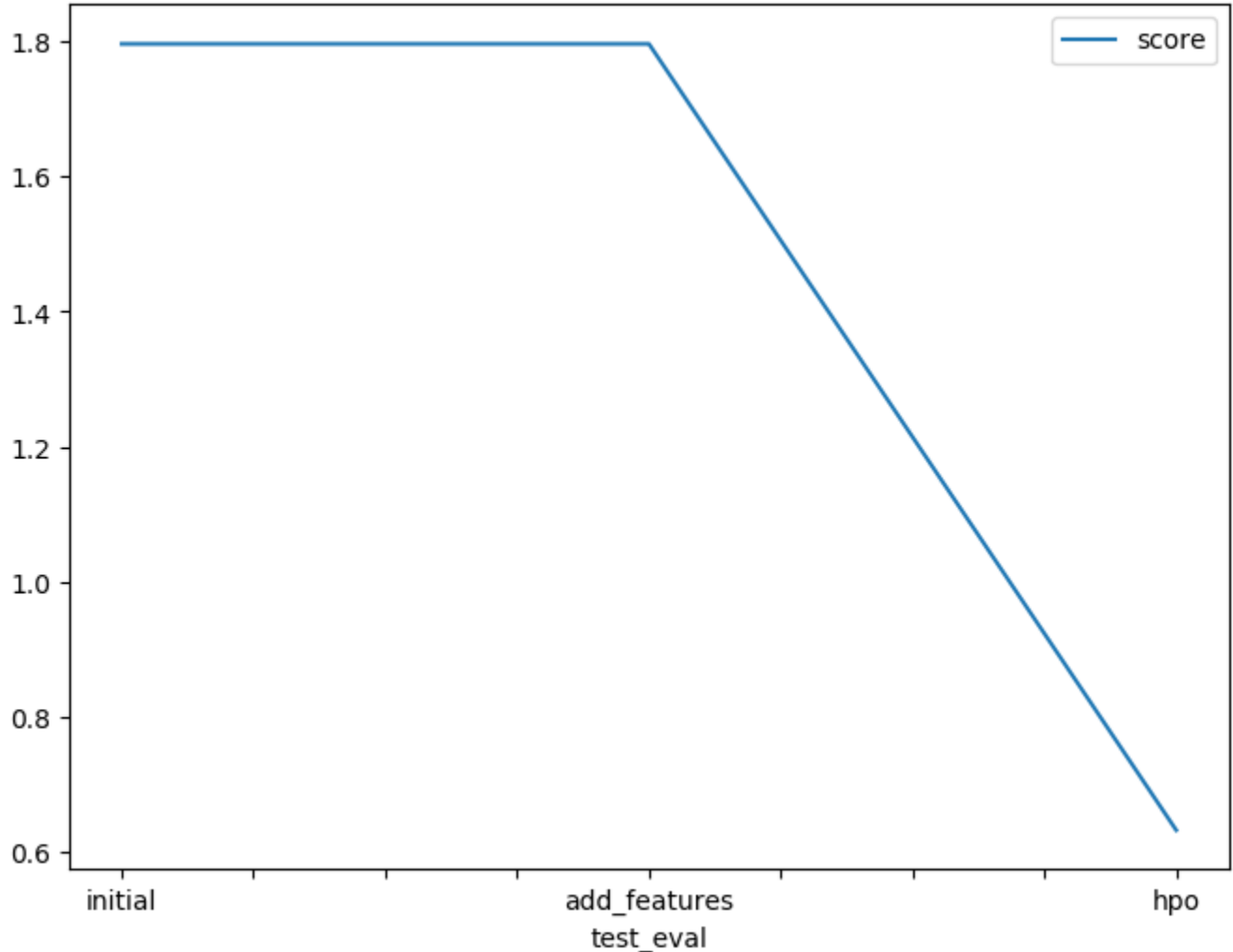
Create a line plot showing the top model score for the three (or more) training runs during the project.

```
In [10]: fig = pd.DataFrame(
    {
        "model": ["initial", "add_features", "hpo"],
        "score": [53.025270, 30.090282, 30.330782]
    }
).plot(x="model", y="score", figsize=(8, 6)).get_figure()
fig.savefig('model_train_score.png')
```



Create a line plot showing the top kaggle score for the three (or more) prediction submissions during the project.

```
In [12]: fig = pd.DataFrame(
    {
        "test_eval": ["initial", "add_features", "hpo"],
        "score": [1.79637, 1.79637, 0.63238]
    }
).plot(x="test_eval", y="score", figsize=(8, 6)).get_figure()
fig.savefig('model_test_score.png')
```



Summary

From the project completed it was noted that in order to improve model performance as far as accuracy of predictions is involved, a combination of exploratory data analysis and hyperparameter optimization is required. Overall, however, according to the results produced, EDA did not do much in improving the model performance. The greatest influence on performance and accuracy was HPO. As a result, the key takeaway, at least in the case of Autogluon is that better results are highly related to better HPO than EDA.