Report: Predict Bike Sharing Demand with AutoGluon Solution
Kali Johari

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?
Trim negative counting numbers to zero.
Lower the values in the count column to zero.

What was the top ranked model that performed?
WeightedEnsemble_L3

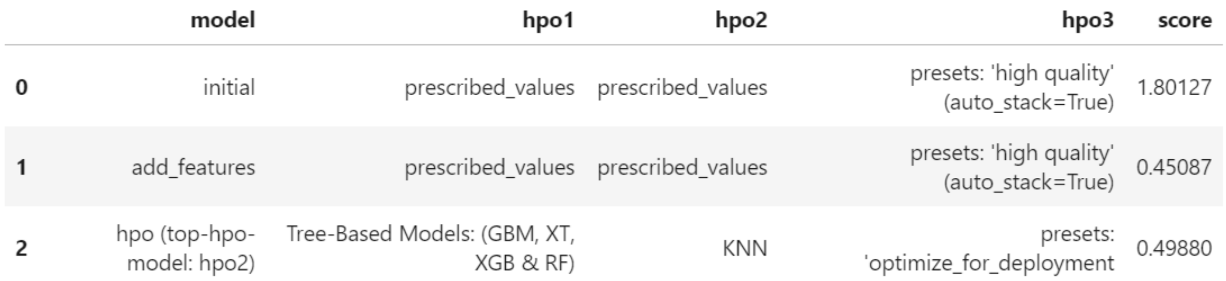
Exploratory data analysis and feature creation
What did the exploratory analysis find and how did you add additional features?
Initially, the datetime feature was converted to extract hour information from the timestamp. The season and weather features, initially stored as integers, were converted to categorical data types. Year, month, day (dayofweek), and hour were extracted as separate features from datetime, which was subsequently removed. The casual and registered features showed significant improvement in RMSE scores during cross-validation but were excluded from model training due to their absence in the test dataset. A new categorical feature, day_type, was created based on holiday and workingday to distinguish between weekdays, weekends, and holidays. Additionally, due to their high correlation, the atemp feature was removed to reduce multicollinearity with temp. Finally, data visualization was used to gain insights from the features.

How much better did your model preform after adding additional features and why do you think that is?
The model score improved by around 30% after adding the extra feature. Including important characteristics based on domain expertise will result in more precise forecasts.

Hyper parameter tuning
How much better did your model preform after trying different hyper parameters?
The kaggle score improved somewhat after altering the hyper settings.

If you were given more time with this dataset, where do you think you would spend more time?
Shall devote more work to future engineering and hyper parameter tweaking, as these are critical procedures for developing an efficient model.

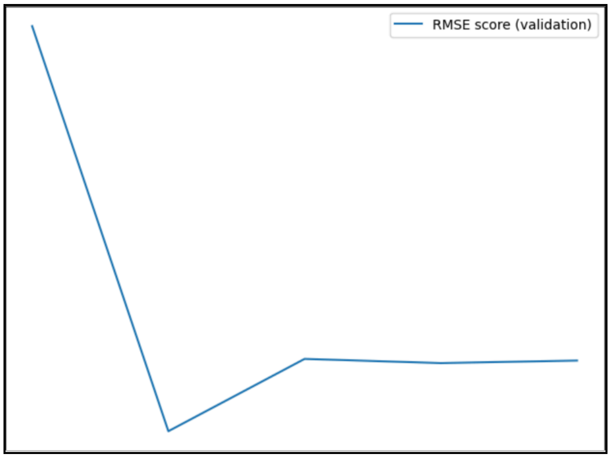
Create a table with the models you ran, the hyperparameters modified, and the kaggle score.



	model	hpo1	hpo2	hpo3	score
0	initial	prescribed_values	prescribed_values	presets: 'high quality' (auto_stack=True)	1.80127
1	add_features	prescribed_values	prescribed_values	presets: 'high quality' (auto_stack=True)	0.45087
2	hpo (top-hpo-model: hpo2)	Tree-Based Models: (GBM, XT, XGB & RF)	KNN	presets: 'optimize_for_deployment	0.49880

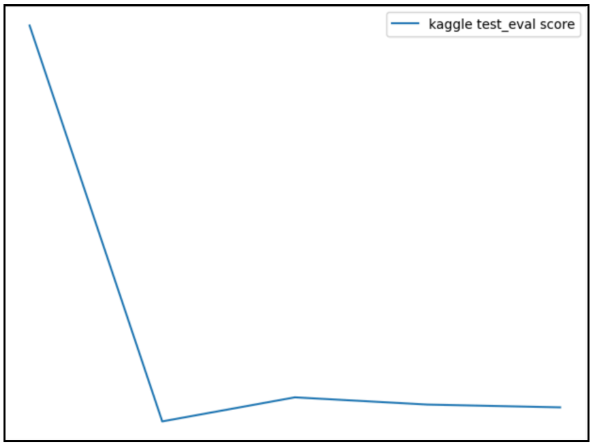
Create a line plot showing the top model score for the three (or more) training runs during the project.

TODO: Replace the image below with your own.



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

TODO: Replace the image below with your own.



Summary

The project focused on leveraging the AutoGluon AutoML framework for Tabular Data in a bike sharing demand prediction scenario. Utilizing AutoGluon, a combination of stack ensembled and individually configured regression models were developed, rapidly prototyping a base-line model. Subsequent enhancements were made by incorporating data from extensive exploratory data analysis (EDA) and feature engineering, resulting in significant improvements in model performance. While automatic hyperparameter tuning and model selection provided further enhancements, it was observed that hyperparameter tuning using AutoGluon without default parameters or random configuration can be cumbersome and highly dependent on factors like time limits, presets, and the range of parameters. Throughout the ML lifecycle, from defining the problem and business objectives to obtaining and analyzing data, and finally building and testing models, the project aimed to optimize bike sharing demand predictions using data-driven approaches.