

# Report: Predict Bike Sharing Demand with AutoGluon Solution

Farid Rahimli

## 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?

*Initially the predictor did not accept the test set due to difference in columns train set had columns for casual and registered users however, test set was lacking these columns. The training would not have worked unless these columns were dropped or synthetically added to the test. Latter would be very inaccurate therefore my decision was to drop the columns from the training set. After conducting data visualizations specifically of the target variable distribution it was evident that the variable was highly skewed to the right. This meant that lower values were prevalent in the training set. To eliminate this data was transformed with log1. Result of this was bell shaped distribution which was much more applicable. During submission to kaggle some rmse values were negative which cause an error. This were eliminated by transforming the output to absolute values.*

What was the top ranked model that performed?

*Top ranked model for all three stages of training was WeightedEnsembleL3. The lowest error was on the tertiary training according to AutoGluon evaluation metrics but according to Kaggle the lowest error was on the secondary training*

## Exploratory data analysis and feature creation

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

### *EDA Findings*

*First and one of the most important findings during EDA was that the target variable was skewed and needed to be balanced. Secondary important finding was that the additional datetime features that were extracted and added to the training and test data had strong impact on rentals. Moreover, EDA showed that amount of rentals were peaking during the rush hour times. Lastly, temperature and weather had significant impact on rentals at all times.*

### *Features Added*

*Additional features such as: hour, day, month, weekday, year were extracted. 2 bool type features were added as well such as: weekday\_bool and rushhour\_bool. Polynomial features that multiples 2 existing features were added: interaction between humidity and temperature at the vector of temp\_humidity, interaction between windspeed and temperature and interaction between hour and workday at the vector of workday\_hour. Lastly to ease the process of training weather, year, weekday and month features were set as categorical values.*

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

*Initial Training score value according to AutoGluon evaluation metrics was ~ -1*

*Secondary Training score value according to AutoGluon evaluation metric after Feature engineering was ~ -0.5300*

*Tertiary Training score value according to AutoGluon evaluation metric after Feature engineering and Hyperparameter tuning was ~ -0.26923*

NOTE ALL THESE VALUES ARE AFTER LOG TRANSFORMATION TO BALANCE THE VALUES AND CREATE BETTER VISUALIZATION

*Initial Training score value according to AutoGluon evaluation metrics was ~ -138.3919*

*Secondary Training score value according to AutoGluon evaluation metric after Feature engineering was ~ -0.69917*

*Tertiary Training score value according to AutoGluon evaluation metric after Feature engineering and Hyperparameter tuning was ~ -0.313665*

NOTE ALL THESE VALUES ARE ORIGINAL VALUES BEFORE LOG TRANSFORMATION

## Hyper parameter tuning

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

*After tuning the score were improved drastically Kaggle score improved from ~ 1.8 to ~ 0.4 AutoGluon RMSE improved from ~ 138 (non log) to ~ 0.3 (non log)*

*Improvements were possible due to additional hyperparameters for each model type AutoGluon uses (added to GBM, XGBoost, Random Forest and Neural Network)*

*Added best+prese quality attribute to choose best model*

*Ensemble models prevailed in terms of accuracy*

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

*Additional feature engineering could improve the results such as extracting and constructing new feature for holidays as well as lag feature could have been implemented due to the timeseries aspect of the data*

*More hyperparameter tuning could improve results as well as implementing Bayesian hyperparameter optimization*

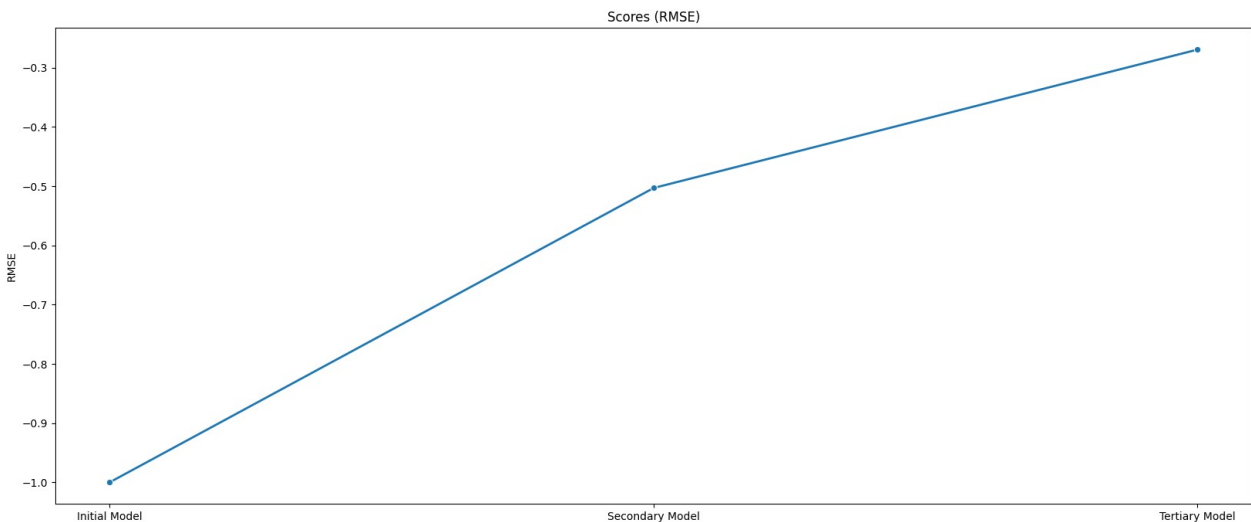
*Using different deep learning models to train*

Create a table with the models you ran, the hyperparameters modified,

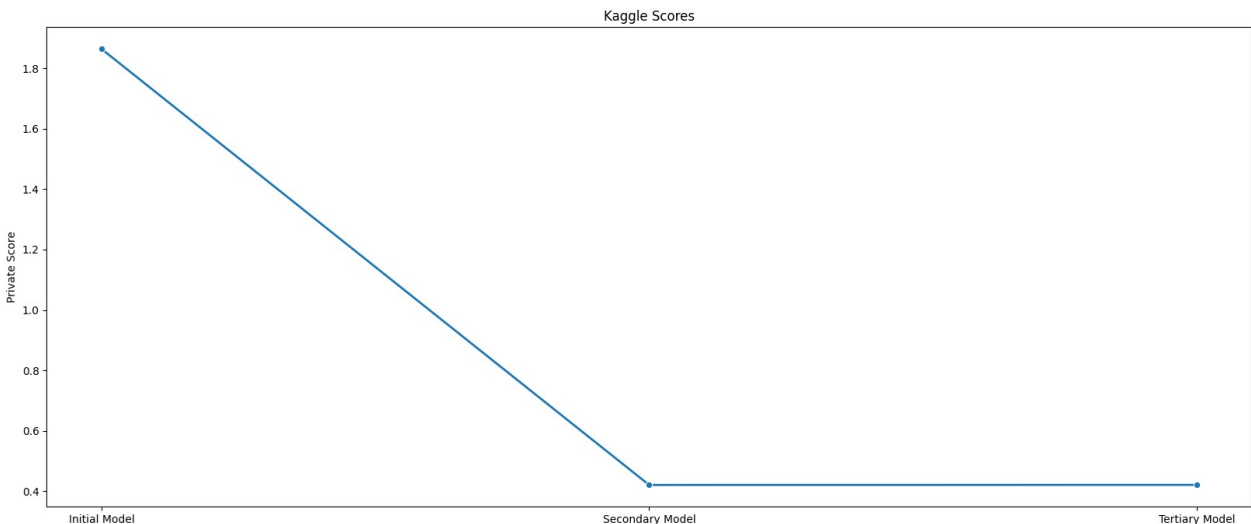
and the kaggle score.

Model Stage	Score (RMSE)	Score (Kaggle)	Changes
Initial Model	-1	1.86413	Default
Secondary Model	-0.503007	0.42063	Feature Engineered \ Cleaned
Tertiary Model	-0.269237	0.42091	Tuned, Additional Features

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



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



## Summary

*The best model was Weighted EnsembleL2 after hyperparameter tuning and Feature Engineering*

*RMSE scores improved substantially after implementing feature extractions and adding new features*

*Specifically Feature Extraction decreased the score from -189 to 0.5 (approximately)*

*Hyperparameter tuning and additional extraction / creation of polynomial features improved the score further from -0.5 to -0.2 (approximately)*

*Implementation of additional features can improve scores extracting holidays, additional weather information and lag*

*Moreover, bayesian hyperparameter tuning can show improvements in accuracy to some degree*

*More complex deep learning models can prove to be more accurate*