## LightGBM - Model performance summary

Before any tuning, using some baseline model parameters, the performance is shown below:
Baseline Performance with n_estimators=100:

Train RMSLE: 0.6513

Valid RMSLE: 0.6738

After Hyperparameter Tuning (with optimal parameters), best Train RMSLE was 0.3040, which was achieved with num_leaves=150. And the best Validation RMSLE achieved was 0.4622 for min_child_samples=5. The individual tuning of parameters are shown in the appendix.

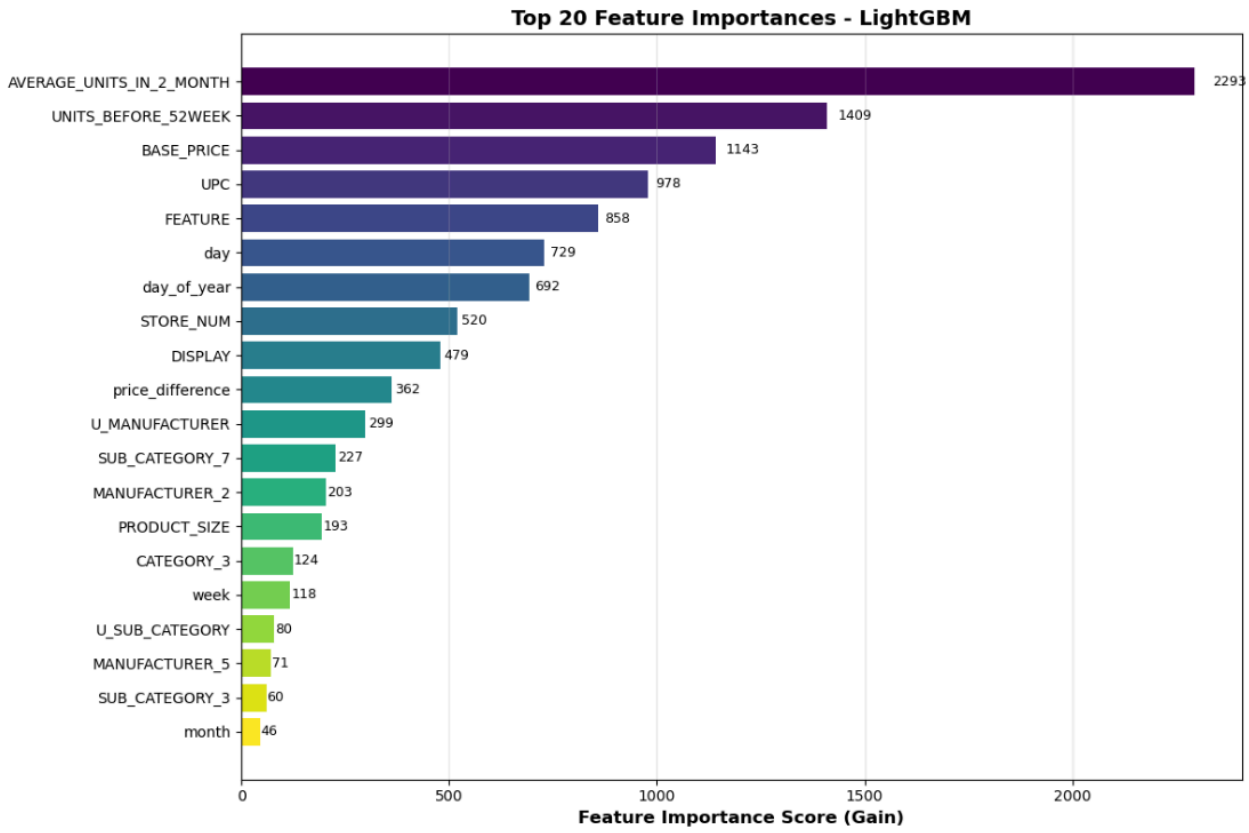
| **Parameter** | **Optimal value** | **Validation RMSLE** |
|---|---|---|
| n_estimators: | 800 | 0.4633 |
| num_leaves | 80 | 0.4629 |
| min_child_samples | 5 | 0.4622 |
| max_depth | 4 | 0.4641 |
| min_split_gain | 0 | 0.4634 |

The average performance on the train and the test sets are shown as below:

| | |
|---|---|
| Average Train RMSLE | 0.3946 |
| Average Train RMSLE | 0.4636 |
| Performance Gap | 0.069 |

Although the performance gap is slightly on the higher side, it wouldn't suggest extreme overfitting. Some regularization could have helped further.

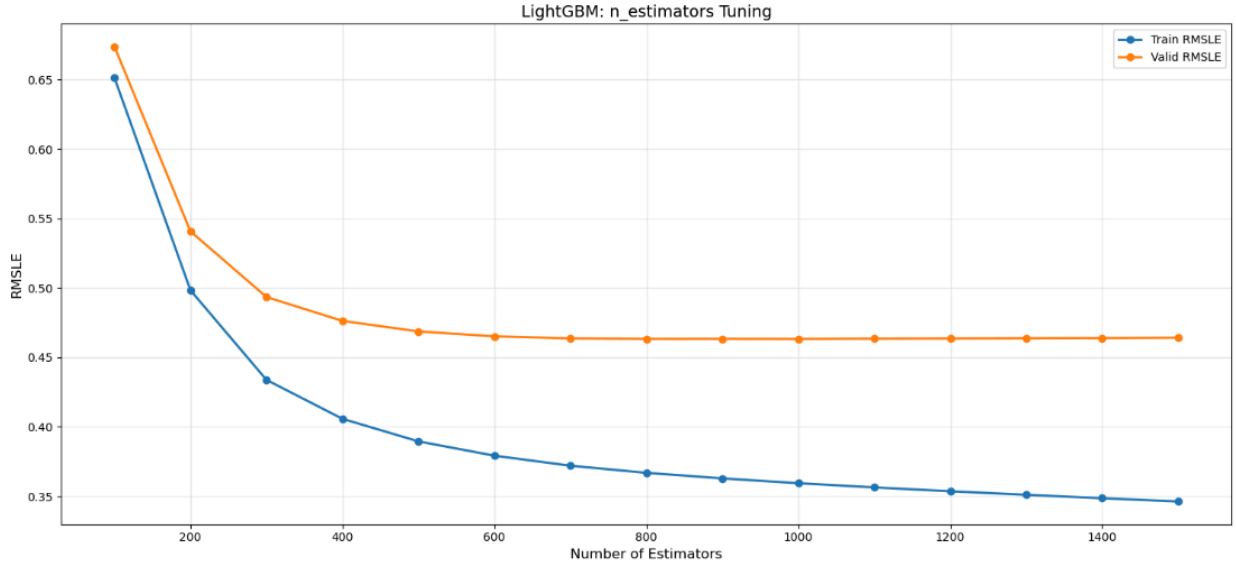
## Feature Importance



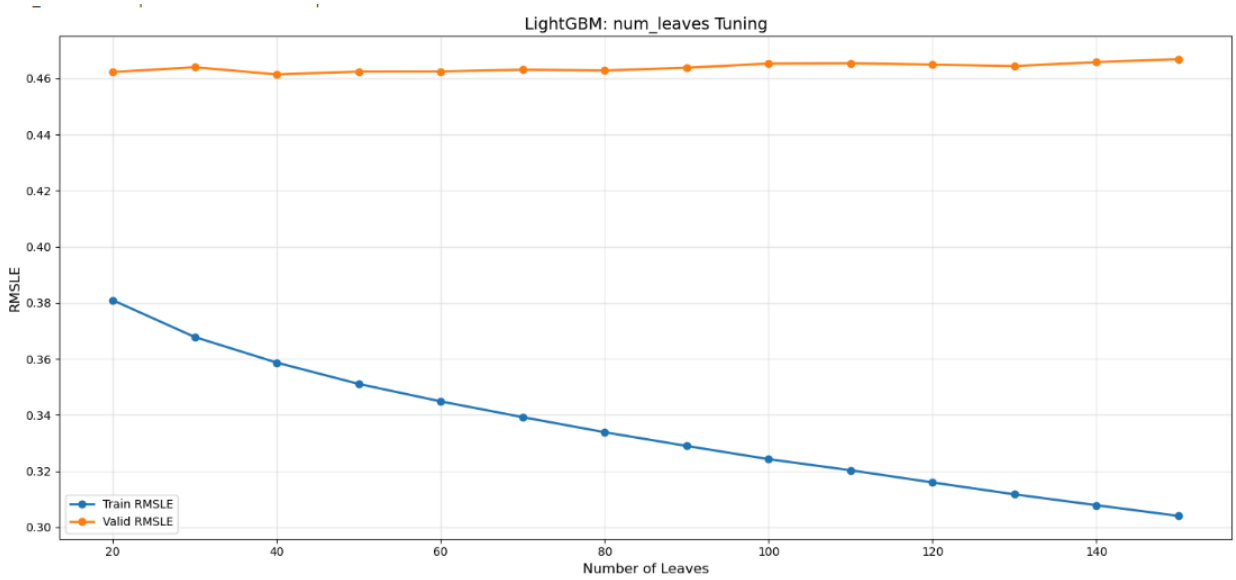

While considering the top 20 features by importance, it was seen that the top 5 features accounted for 61.4% of the total importance while the top 10 features accounted for ~87% of the total importance.

## Appendix

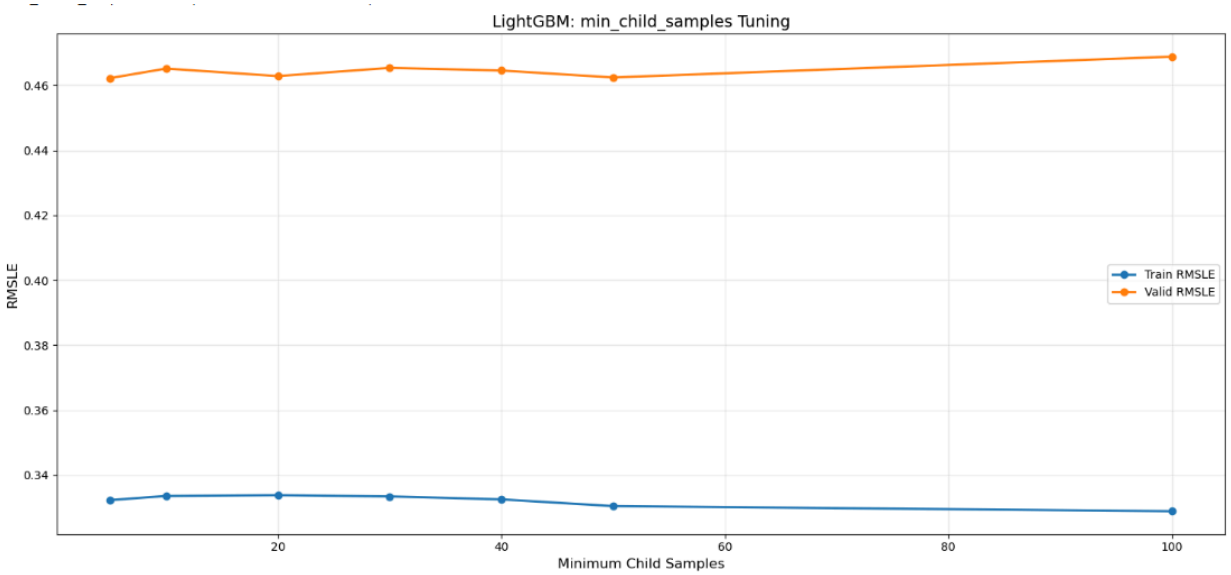
```
Tuning n_estimators:   7%|▋         | 1/15 [00:03<00:53,  3.81s/it]
n_estimators:  100 | Train RMSLE: 0.6513 | Valid RMSLE: 0.6738
Tuning n_estimators:  13%|█▎        | 2/15 [00:13<01:37,  7.47s/it]
n_estimators:  200 | Train RMSLE: 0.4983 | Valid RMSLE: 0.5407
Tuning n_estimators:  20%|██        | 3/15 [00:27<02:05, 10.46s/it]
n_estimators:  300 | Train RMSLE: 0.4338 | Valid RMSLE: 0.4935
Tuning n_estimators:  27%|██▋       | 4/15 [00:42<02:14, 12.23s/it]
n_estimators:  400 | Train RMSLE: 0.4058 | Valid RMSLE: 0.4763
Tuning n_estimators:  33%|███▎      | 5/15 [00:54<02:00, 12.07s/it]
n_estimators:  500 | Train RMSLE: 0.3895 | Valid RMSLE: 0.4687
Tuning n_estimators:  40%|████      | 6/15 [01:08<01:53, 12.61s/it]
n_estimators:  600 | Train RMSLE: 0.3792 | Valid RMSLE: 0.4652
Tuning n_estimators:  47%|████▋     | 7/15 [01:25<01:53, 14.23s/it]
n_estimators:  700 | Train RMSLE: 0.3720 | Valid RMSLE: 0.4637
Tuning n_estimators:  53%|█████▎    | 8/15 [01:43<01:47, 15.42s/it]
n_estimators:  800 | Train RMSLE: 0.3669 | Valid RMSLE: 0.4633
Tuning n_estimators:  60%|██████    | 9/15 [02:03<01:41, 16.90s/it]
n_estimators:  900 | Train RMSLE: 0.3629 | Valid RMSLE: 0.4634
Tuning n_estimators:  67%|██████▋   | 10/15 [02:28<01:36, 19.26s/it]
n_estimators: 1000 | Train RMSLE: 0.3594 | Valid RMSLE: 0.4633
Tuning n_estimators:  73%|███████▎  | 11/15 [02:53<01:23, 20.98s/it]
n_estimators: 1100 | Train RMSLE: 0.3564 | Valid RMSLE: 0.4635
Tuning n_estimators:  80%|████████  | 12/15 [03:22<01:10, 23.34s/it]
n_estimators: 1200 | Train RMSLE: 0.3536 | Valid RMSLE: 0.4636
Tuning n_estimators:  87%|████████▋ | 13/15 [04:03<00:57, 28.96s/it]
n_estimators: 1300 | Train RMSLE: 0.3510 | Valid RMSLE: 0.4638
Tuning n_estimators:  93%|█████████▎| 14/15 [04:47<00:33, 33.21s/it]
n_estimators: 1400 | Train RMSLE: 0.3486 | Valid RMSLE: 0.4640
Tuning n_estimators: 100%|██████████| 15/15 [05:20<00:00, 21.34s/it]
n_estimators: 1500 | Train RMSLE: 0.3463 | Valid RMSLE: 0.4642
```

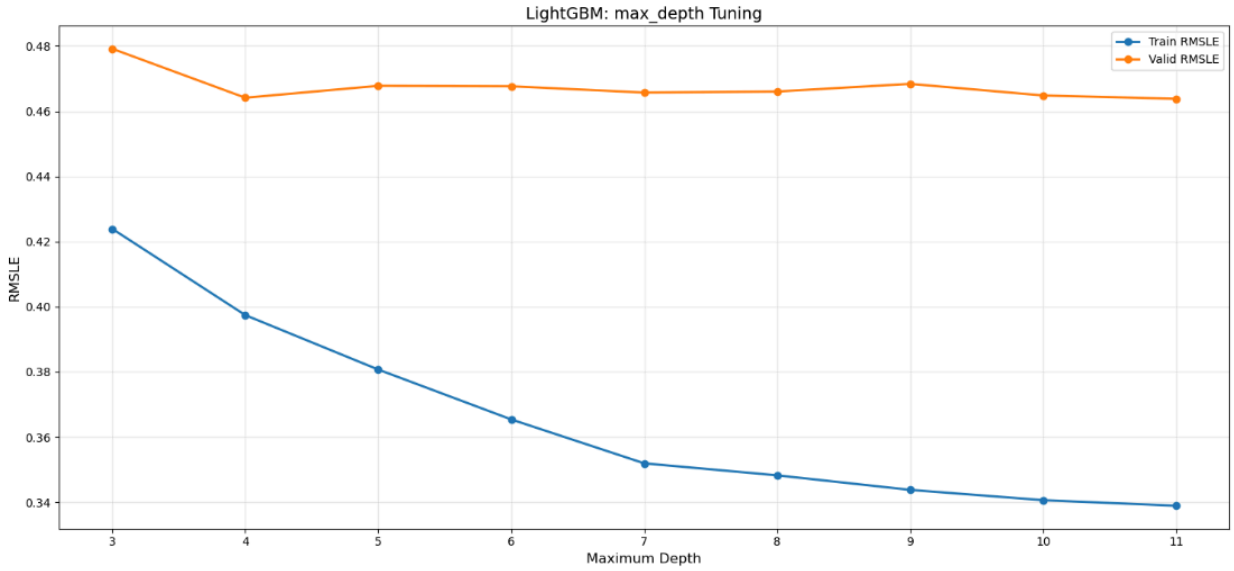



Best validation RMSLE is at n_estimators = 800 train RMSLE also continues to plateau at same level

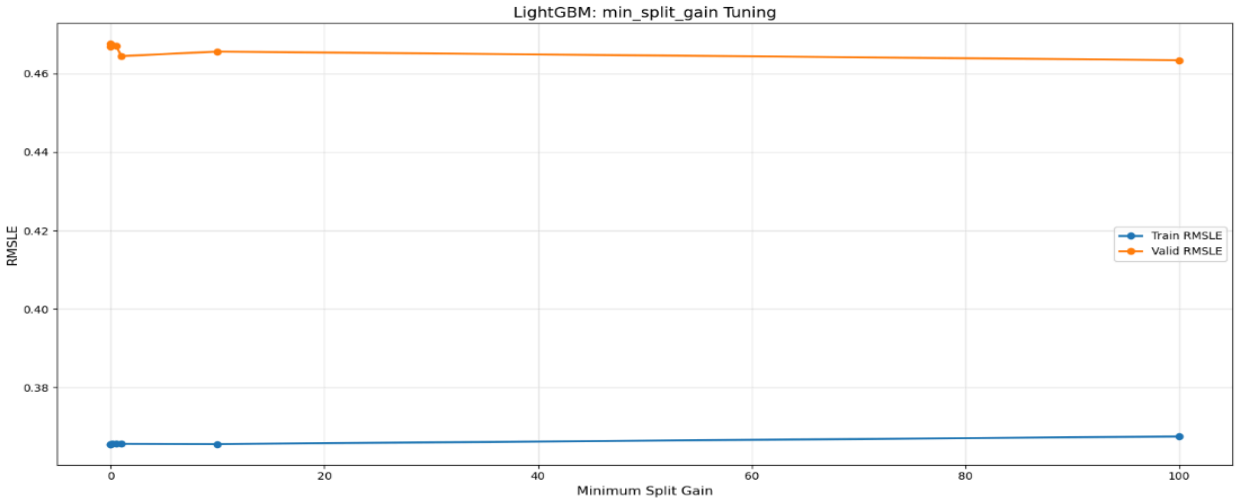



Validation RMSLE seems to be optimal at num_leaves = 80, where train RMSLE is also relatively lower






Best validation RMSLE is seen at max_depth =4 slthough train RMSLE continues to drop substantially until max_depth =7




There is hardly any gain and hence one can be more conservative and take min_split_gain at even 0 or 1 instead of 100