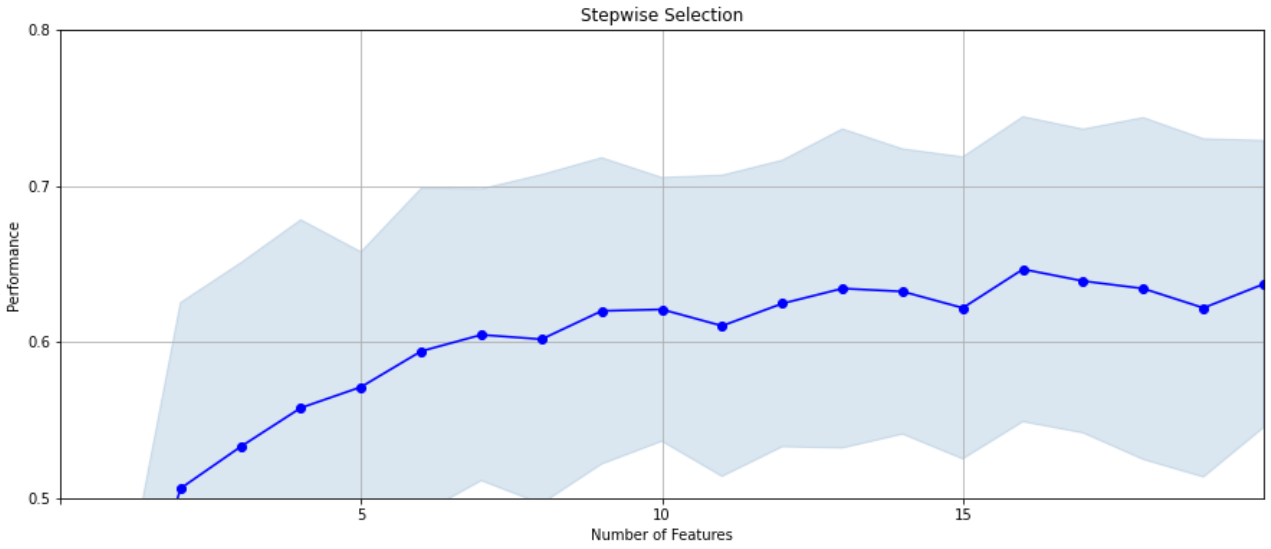
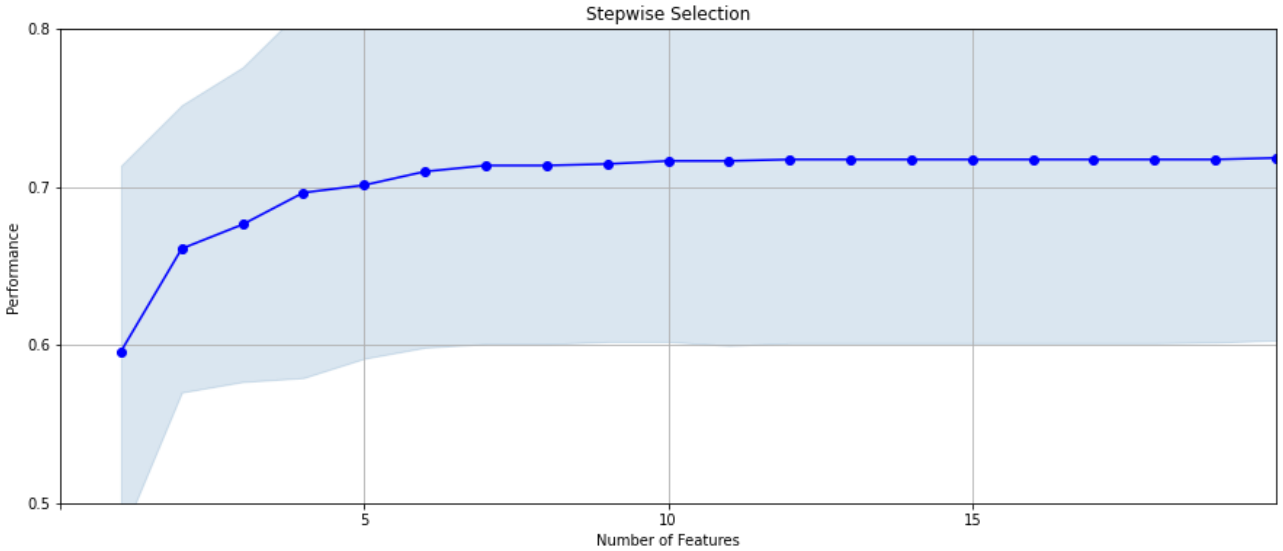
Homework 3

1. Ran Feature Selection for filter number of variables (300) and wrapper as (20) for 2 different wrapper algorithms, Random Forest and LightGBM

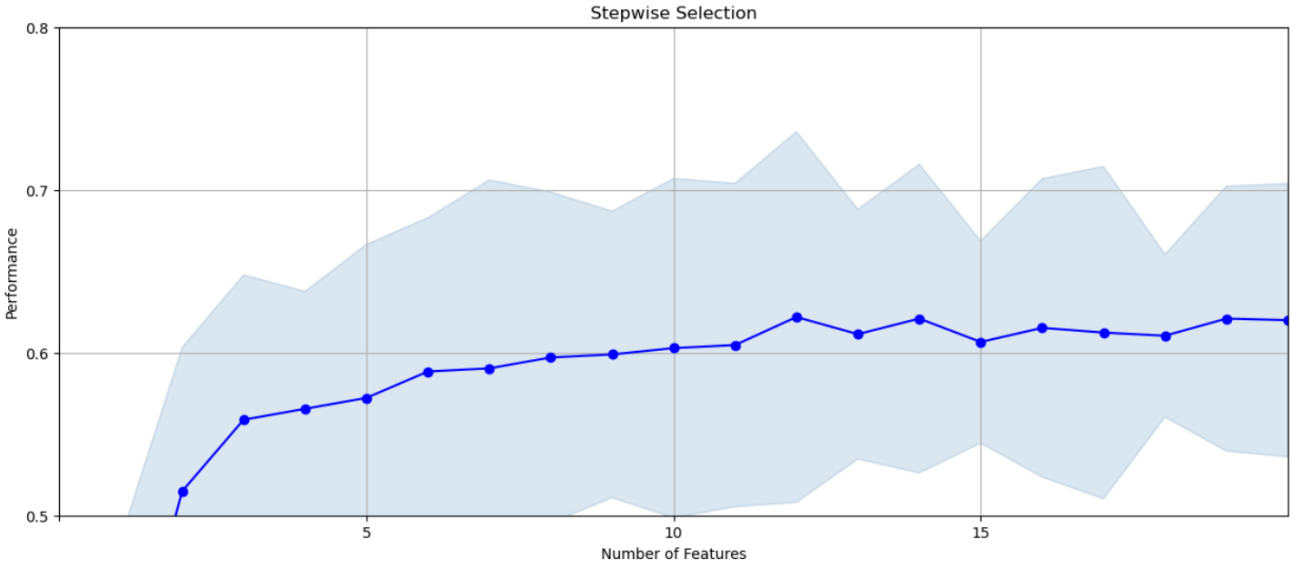
a. Random Forest and Forward Selection:



b. LightGBM and Forward Selection:



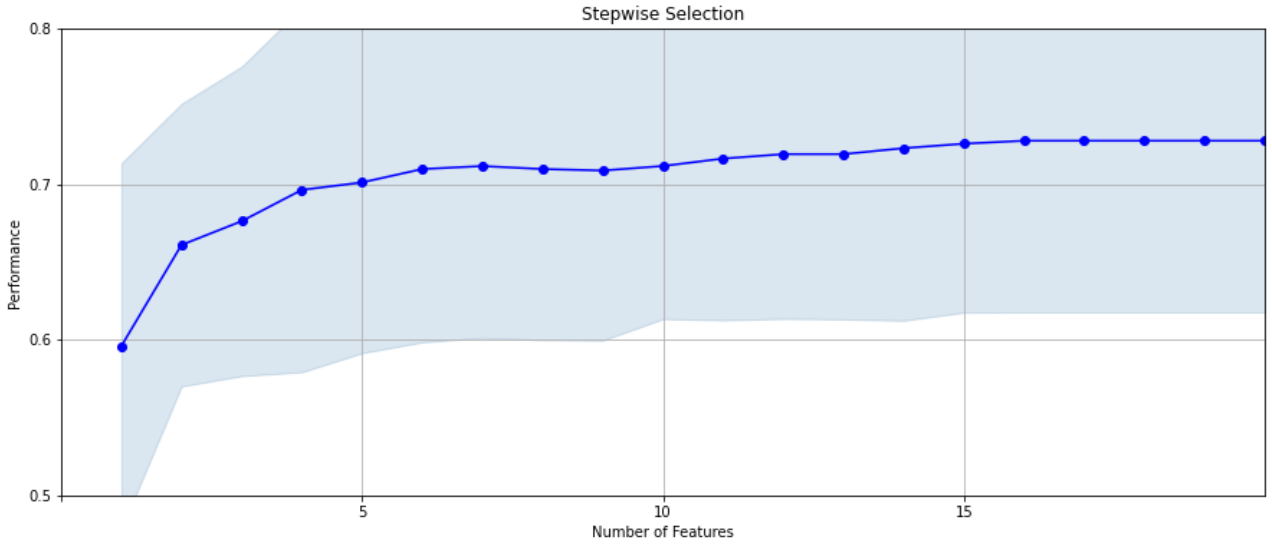
c. LightGBM and Backward Selection:



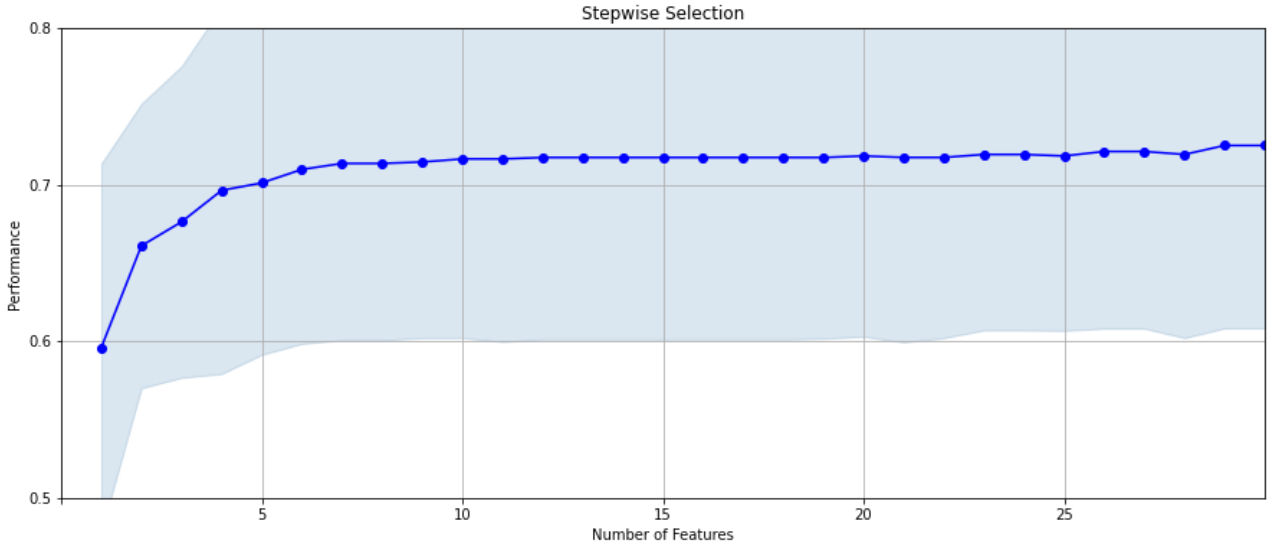
Based on the above graphs, we see that LightGBM provides better performance than Random Forest and hence I will be using that for my next round of trials with different number of features selected in filter and wrapper. I also prefer LightGBM because it is not stochastic in nature and always outputs the same result unlike random forest where we will need multiple runs to see which variables to keep finally. We also see that Forward Selection performs better than Backward Selection. I tried running the RF model with Backward selection too but even after 4 hours it did not output any result and the kernel died. However, our professor had mentioned that Forward selection gives better results in this case so I'm keeping that as my final algorithm for wrapper feature selection.

2. Running Feature selection for different values of number of features selected after filtering and wrapping: (**The algorithms used for wrapper are LGBM and forward selection**)

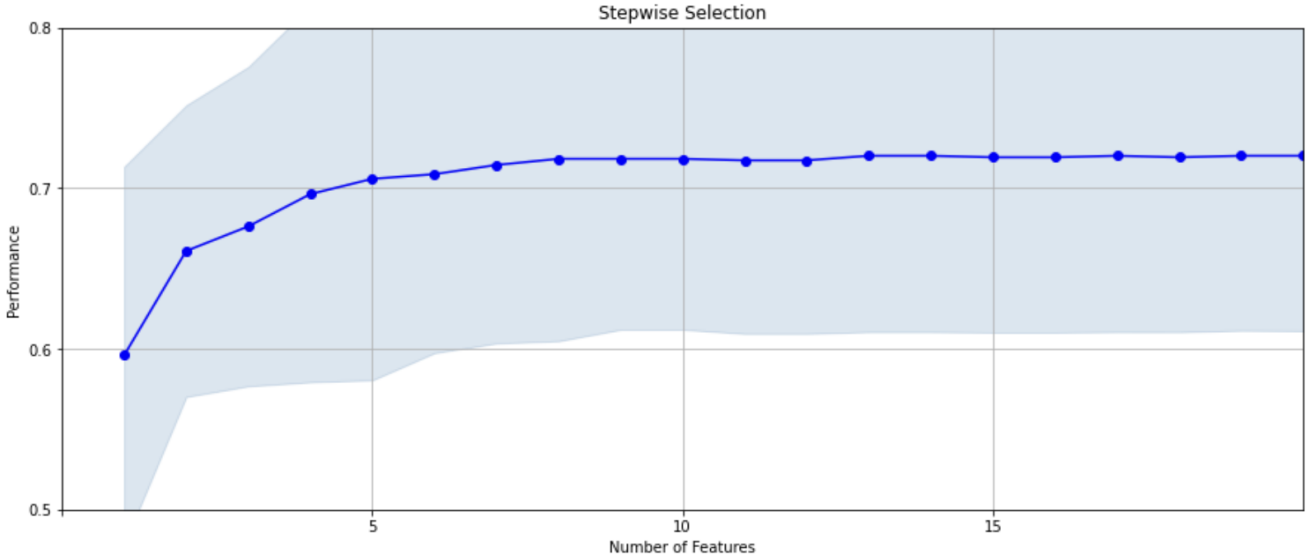
a. First model - Number of features - 200 (7% of the total number of variables), number of variables after wrapping - 20



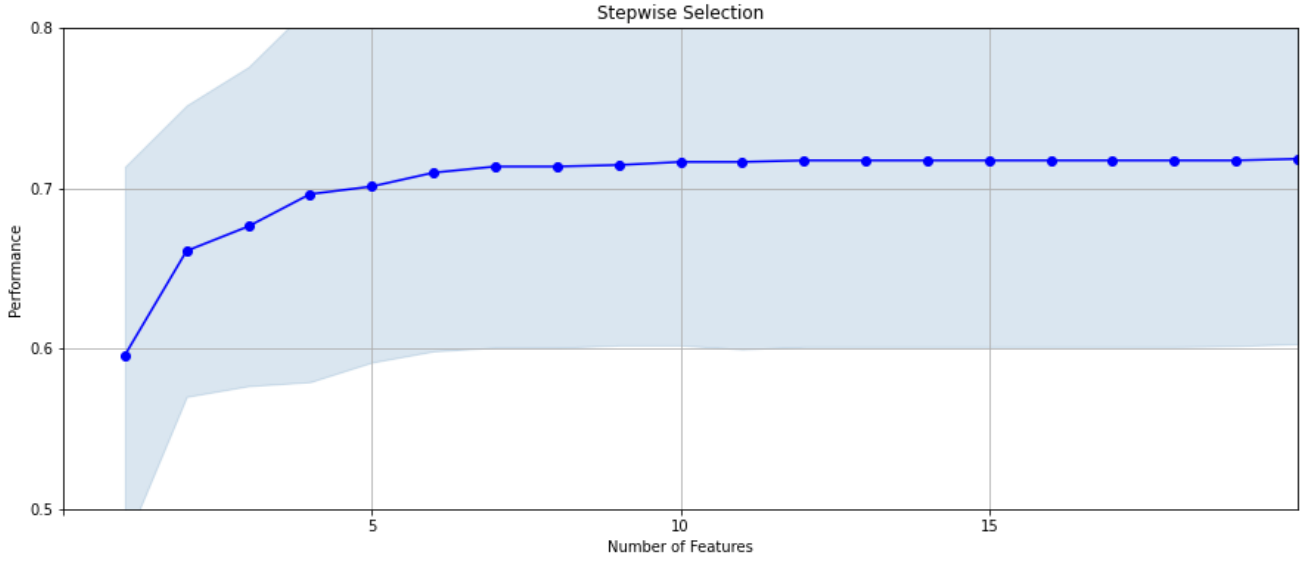
b. Second model - Number of features - 300 (10% of the total number of variables), number of variables after wrapping - 30



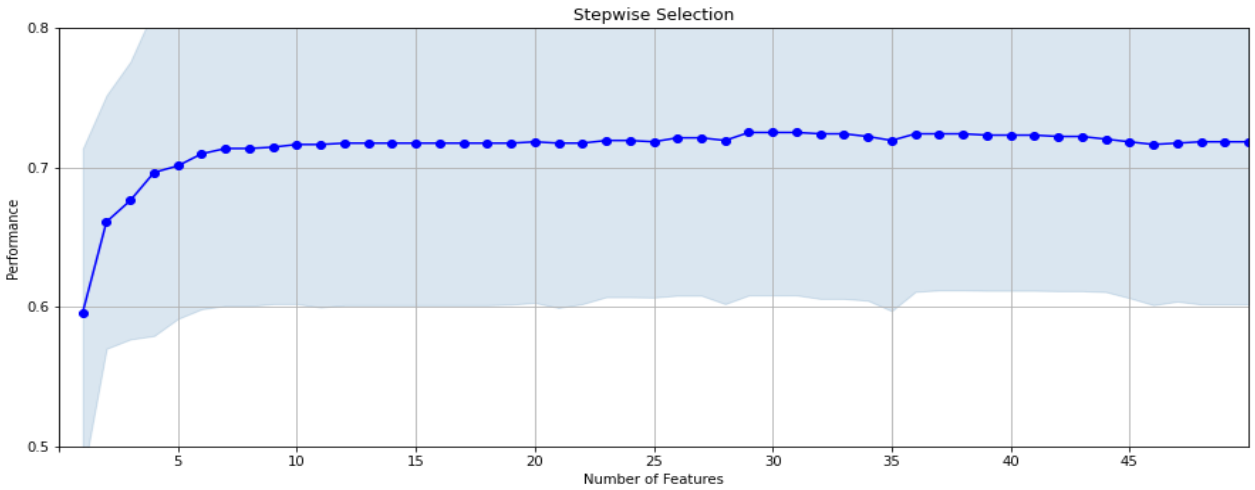
c. Third model - Number of features - 400 (14% of the total number of variables), number of variables after wrapping - 20



d. Fourth model - Number of features - 300 (10% of the total number of variables), number of variables after wrapping - 20



e. Fifth model - Number of features - 300 (10% of the total number of variables), number of variables after wrapping - 50



Based on the above graphs, I see that the model performs better and is more stable in terms of performance when I increase the filter number of variables from 200 to 300, while keeping wrapper variables at 20. Next, on increasing the number of filter variables to 400 we don't see any improvement in performance so we can keep the filter variables at 300. I then increase the number of wrapper variables to 30 and see that the performance remains the same. I also put wrapper variables as 50 to get better insight into where exactly saturation occurs. The saturation point comes somewhere around 10 variables and I would like to be a little conservative in my final selection to also account for the variability (that we get from cross-validation) so I will keep around 20 variables. **Hence my final model will have 300 variables after filtering and 20 after wrapping.**

3. List of Final Variables - These are the final 20 variables selected by the model (I ran it a couple of times and it gave the same result), the variables are sorted in the order selected by the wrapper and are printed along with their univariate KS score.

	wrapper order	variable	filter score
0	1	card_merch_total_14	0.630048
1	2	card_zip3_max_14	0.629515
2	3	card_merch_avg_14	0.518386
3	4	state_des_max_1	0.524301
4	5	card_zip_max_60	0.605193
5	6	Card_Merchnum_desc_avg_30	0.519394
6	7	Card_Merchnum_desc_avg_1	0.511187
7	8	card_merch_avg_1	0.515008
8	9	card_zip_avg_14	0.538336
9	10	Card_Merchdesc_Zip_avg_7	0.517607
10	11	Card_Merchdesc_Zip_avg_30	0.523477
11	12	Card_Merchnum_Zip_total_14	0.627421
12	13	Card_Merchnum_Zip_avg_14	0.518122
13	14	card_merch_avg_7	0.524281
14	15	Card_Merchnum_Zip_avg_7	0.523337
15	16	Card_Merchnum_desc_avg_7	0.519505
16	17	Card_Merchdesc_avg_7	0.516608
17	18	Card_Merchnum_Zip_avg_1	0.514052
18	19	Card_Merchdesc_Zip_max_14	0.603151
19	20	Card_Merchdesc_max_14	0.603965

4. Plot of Final Variables - This is the performance of the model selected by me (300 variables after the filter and 20 after wrapper function).