Credit Card Fraud Analytics

Shreyash Kondakindi

May 1, 2025

1 Introduction

This report documents a series of stepwise feature selection experiments with the goal of improving fraud detection performance (FDR metric). Each trial refines the filter and wrapper sizes and the model hyper-parameters. After each experiment, the performance graph and the selected features are attached for reference, which then outline the rationale guiding the next trial.

2 Baseline Experiment

- **Filter:** 200 features
- **Wrapper:** Forward selection, 20 features
- **Model:** LGBMClassifier(n_estimators=20, num_leaves=4)
- **CV:** 2-fold
- **Performance:** Saturated at ~ 0.68

Note: Baseline was discussed in the slides which I have adopted and I hope too improve on the same.

3 Experiments

3.1 Trial 1: LGBM (30 trees, 6 leaves, max_depth=4)

- **Filter:** 200 **Wrapper:** 20
- **Model:** LGBMClassifier(n_estimators=30, num_leaves=6, max_depth=4)
  More trees to capture more signal; num_leaves increased to add complexity to the model but it is still efficient; max_depth parameter restricts overfitting and maintains speed.
- **CV:** 3-fold (For better generalization)
- **Duration:** 00:09:30.78
- **Result:** Performance remained ~ 0.68; saturation before 15 features.

Performance did not improve, so expanded the filter pool to 300, expecting to get a richer feature pool for the wrapper to choose from and switched to Random Forest to explore different inductive biases.

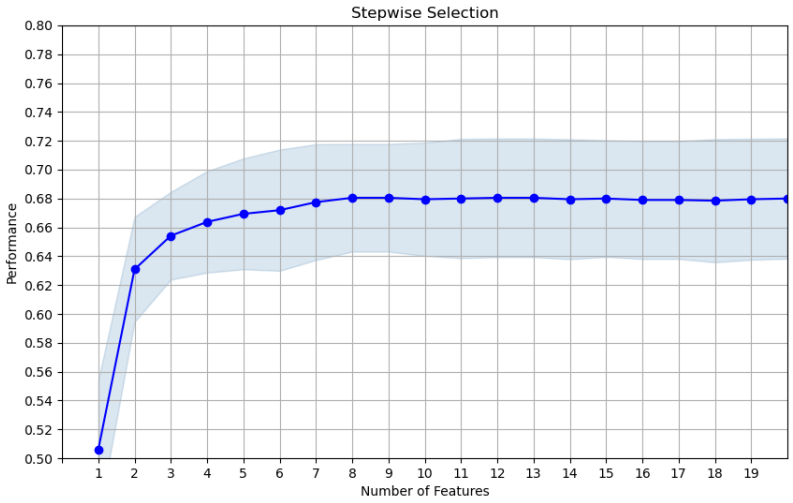

Figure 1: Performance curve for Trial 1.

wrapper order	variable	filter score
1	Cardnum_unique_count_for_card_state_1	0.5186059951957085
2	Cardnum_total_1	0.5911137143420107
3	Card_Merchnum_Zip_total_amount_1_by_60	0.3058974499494893
4	Cardnum_total_amount_0_by_60	0.37029630054350016
5	Cardnum_unique_count_for_card_zip_1	0.5172084487865884
6	card_zip_total_amount_1_by_30	0.281011201038026
7	Card_dow_unique_count_for_merch_state_1	0.49093766046906934
8	Card_dow_count_30	0.37512322282238025
9	Card_dow_unique_count_for_merch_zip_1	0.49073745525872875
10	Cardnum_total_3	0.569584754753369
11	card_state_count_1_by_30_sq	0.30483642569076347
12	Cardnum_actual/avg_0	0.3142753854346274
13	card_state_count_1_by_30	0.30483642569076347
14	Card_dow_unique_count_for_merch_zip_7	0.4468895047803746
15	Cardnum_variability_avg_0	0.34581915450678724
16	Card_dow_unique_count_for_merch_state_14	0.41787101399187326
17	Card_dow_unique_count_for_merch_zip_14	0.41773337290976414
18	card_state_total_amount_1_by_60	0.348890891644948 7
19	card_state_total_amount_1_by_30	0.29984290860156976
20	card_merch_total_amount_1_by_60	0.3055671810441145

Figure 2: Top 20 features selected in Trial 1.

3.2 Trial 2: Random Forest (7 trees, max_leaf_nodes=10)

- **Filter:** 300 **Wrapper:** 20
- **Model:** RandomForestClassifier(n_estimators=7, max_leaf_nodes=10)
- **CV:** 2-fold (Reduced in the interest of time but still offers decent generalization)

- **Duration:** 00:13:00.80
- **Result:** Performance 0.68 ± 0.05

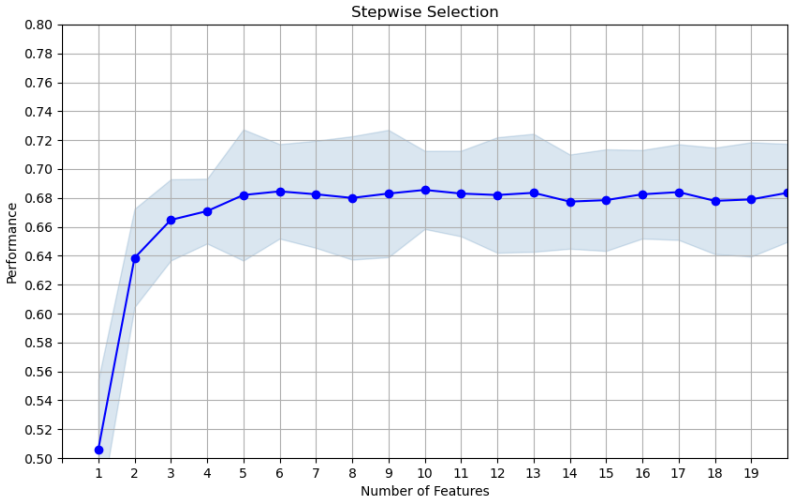

Figure 3: Performance curve for Trial 2.

wrapper order	variable	filter score
1	Cardnum_unique_count_for_card_state_1	0.5186059951957085
2	Card_Merchnum_State_total_7	0.2598647537776855
3	card_zip_vdratio_1by60	0.2762725401131096
4	Card_Merchdesc_Zip_vdratio_0by60	0.2777984459387328
5	Card_dow_count_0_by_14	0.25581156128069243
6	Card_Merchdesc_total_30	0.2552448791657018
7	card_zip_total_3	0.2602579352697274
8	Card_Merchnum_Zip_total_amount_1_by_60	0.3058974499494893
9	card_merch_total_amount_1_by_60	0.3055671810441145
10	Cardnum_count_0	0.5290724258187427
11	Card_Merchnum_Zip_total_30	0.25202678948453816
12	Card_Merchnum_State_total_30	0.25398954181200284
13	Cardnum_count_1_by_7	0.31810073679952333
14	Card_dow_total_amount_0_by_30	0.2546757578416611
15	Card_Merchdesc_State_total_3	0.2599209696267789
16	card_zip_total_amount_1_by_60	0.34588508284282526
17	Cardnum_total_amount_0_by_30	0.33047395732524576
18	card_state_count_1_by_60_sq	0.34439841977267205
19	Card_Merchnum_desc_vdratio_0by30	0.27011254257132544
20	card_state_vdratio_1by60	0.29620295613130043

Figure 4: Top 20 features selected in Trial 2.

With RF not yielding improvement and there's no saturation till 20 features, I tested deeper LGBM trees (25, 8 leaves, depth=5) and expanded wrapper to 25 features.

3.3 Trial 3: LGBM (25 trees, 8 leaves, max_depth=5)

- **Filter:** 250 **Wrapper:** 25
- **Model:** LGBMClassifier(n_estimators=25, num_leaves=8, max_depth=5)
  More leaves (+ depth) increases capacity. Pushing wrapper to 25 lets us test if a few extra features boosts performance from 0.68.
- **CV:** 2-fold
- **Duration:** 00:19:16.02
- **Result:** Performance ∼0.705

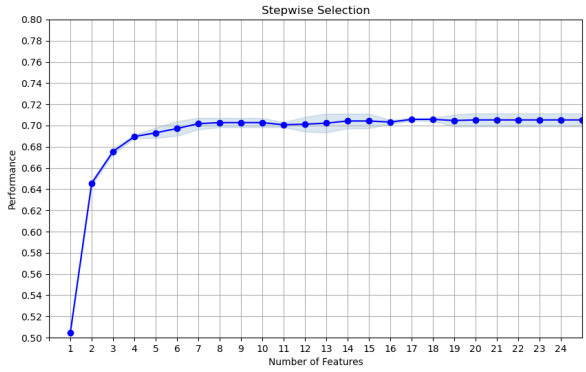

Figure 5: Performance curve for Trial 3.

wrapper order	variable	filter score
1	Cardnum_unique_count_for_card_state_1	0.5186059951957085
2	card_zip_total_7	0.263851061261844
3	card_zip_total_amount_1_by_60	0.34588508284282526
4	Cardnum_max_1	0.5048046652882496
5	Cardnum_vdratio_0by14	0.4017270867195838
6	Card_Merchnum_Zip_total_amount_1_by_60	0.3058974499494893
7	Card_dow_unique_count_for_merch_zip_14	0.41773337290976414
8	card_merch_total_amount_1_by_60	0.3055671810441145
9	Card_dow_unique_count_for_merch_state_14	0.41787101399187326
10	Card_Merchnum_State_total_amount_1_by_60	0.3053920014850665
11	Card_dow_count_60	0.2983163058590752
12	Card_dow_total_60	0.38078676212738305
13	Card_dow_total_7	0.5087544606639527
14	Card_dow_unique_count_for_Card_Merchdesc_14	0.3897333754440073
15	Card_dow_unique_count_for_state_des_14	0.3897333754440073
16	Card_Merchdesc_State_total_14	0.2673557168249354
17	Cardnum_variability_med_0	0.26918950785314444
18	Cardnum_variability_avg_1	0.2881269377060854
19	Cardnum_count_3	0.5721987062055063
20	Cardnum_actual/toal_0	0.48947656157688213
21	Card_dow_variability_max_7	0.411571341637837
22	Card_dow_count_30	0.37512322282238025
23	Cardnum_variability_avg_0	0.34581915450678724
24	Card_dow_actual/toal_14	0.3370331741934217
25	Card_dow_actual/toal_7	0.3941931555794056

Figure 6: Top 25 features selected in Trial 3.

As expected with more variables out of the wrapper, performance is better and saturation also occurs around the 15 variable. TIme taken is slightly higher but performance increased to $\sim$ 0.705. Next, switch to CatBoost as it excels on high-cardinality, categorical interactions (merchant/ZIP/day-of-week). This might increase the performance. Let's make some optimizations with the hyper-parameters of the model so that the time for implementation is also not significantly increased. Let's try CatBoost (30, depth=6, LR=0.1) with early stopping and L2 regularization.

3.4 Trial 4: CatBoost (30 iterations, depth=6, LR=0.1)

- **Filter:** 250 **Wrapper:** 25
- **Model:** CatBoostClassifier(iterations=30, depth=6, learning_rate=0.1, l2_leaf_reg=3)
- **CV:** 2-fold
- **Duration:** 00:40:09.04
- **Result:** Performance ~ 0.705

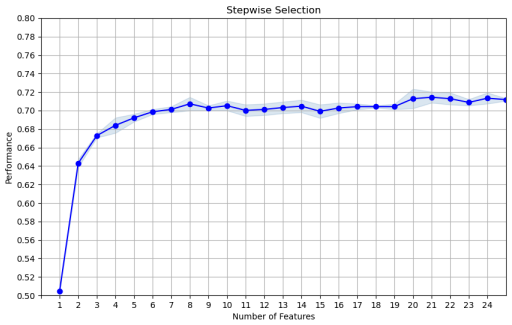

Figure 7: Performance curve for Trial 4.

wrapper order	variable	filter score
1	Cardnum_unique_count_for_card_state_1	0.5186059951957085
2	card_zip_total_7	0.263851061261844
3	Card_dow_total_amount_0_by_60	0.32686028495663755
4	Cardnum_max_1	0.5048046652882496
5	card_zip_total_amount_1_by_60	0.34588508284282526
6	Card_dow_vdratio_0by14	0.5026571349557347
7	Cardnum_count_7	0.5163992459359451
8	Card_Merchdesc_State_total_14	0.2673557168249354
9	Cardnum_avg_1	0.3290144740130786
10	card_state_max_1	0.26648415258551406
11	Cardnum_actual/max_3	0.3470972303573188
12	Cardnum_count_3	0.5721987062055063
13	card_zip_total_amount_1_by_30	0.281011201038026
14	Cardnum_vdratio_0by60	0.44692144891551816
15	Card_dow_unique_count_for_merch_state_1	0.49093766046906934
16	Card_dow_day_since	0.45108180188490676
17	card_state_max_3	0.26883427031694823
18	Card_Merchnum_Zip_total_14	0.2645889695165392
19	Cardnum_max_14	0.2923715986100943
20	Cardnum_total_30	0.36910954614555874
21	Card_dow_unique_count_for_Card_Merchdesc_14	0.3897333754440073
22	Cardnum_variability_avg_0	0.34581915450678724
23	Card_dow_unique_count_for_merch_zip_60	0.31927203864974263
24	Cardnum_unique_count_for_Merchnum_60	0.283133002466769
25	Cardnum_unique_count_for_card_zip_3	0.5068666712494231

Figure 8: Top 25 features selected in Trial 4.

Did not see much performance improvement and the performance curve has not saturated even after 25 variables. The duration is also 2x the iteration with LGBM which also achieved similar performance.
To regain speed, I returned to baseline LGBM (20,4) which seemed to work decently well but retained the larger filter pool (250).

3.5 Trial 5: Baseline LGBM + Expanded Filter

- **Filter:** 250
  More raw candidates gives the wrapper extra diversity—some of the weaker but complementary signals can now get in.

- **Wrapper:** 20
  Let's keep it at 20 but give more variables out of the filter to work with. Maybe some variables with faint signal might combine to give a better estimate.

- **Model:** LGBMClassifier(n_estimators=20, num_leaves=4, learning_rate=0.1)
  Keeps individual model fits ultra-fast, so adding more wrapper iterations (30 vs 20) only costs you $\sim 50\%$ more time.
  Learning rate 0.1 ensures those 20 trees still capture the bulk of signal quickly, so wrapper loops remain fast. Maybe a slower learning rate can capture more refined details but would increase the time of training too much.

- **CV:** 2-fold

- **Duration:** 00:21:03.89

- **Result:** Performance ~ 0.70 with much less time than previous trial.

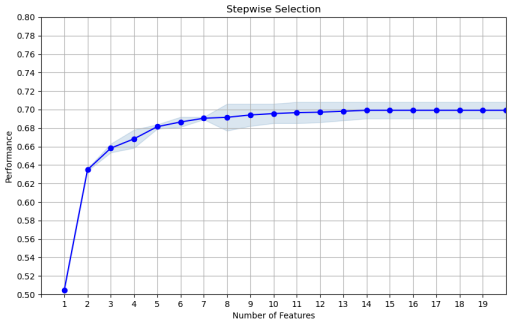

Figure 9: Performance curve for Trial 5.

wrapper order	variable	filter score
1	Cardnum_unique_count_for_card_state_1	0.5186059951957085
2	Card_Merchnum_desc_total_14	0.26558329883066845
3	card_state_total_amount_1_by_60	0.3488908916449487
4	card_state_max_14	0.2293557922186697
5	Card_dow_vdratio_0by30	0.5076936391446802
6	Cardnum_total_amount_1_by_60	0.4574137668325223
7	card_zip_count_0_by_60	0.2642663413543186
8	merch_zip_total_1	0.24215877603657676
9	Card_dow_unique_count_for_merch_state_14	0.41787101399187326
10	Cardnum_actual/toal_1	0.4755878953142161
11	Cardnum_variability_med_0	0.26918950785314444
12	Merchnum_desc_total_3	0.24915661971242042
13	card_state_total_1	0.2954647564386408
14	Cardnum_vdratio_0by14	0.4017270867195838
15	Card_dow_vdratio_0by7	0.490635686388675
16	Cardnum_unique_count_for_card_zip_7	0.47442357530392704
17	Cardnum_unique_count_for_Merchnum_7	0.4680367503273133
18	Cardnum_day_since	0.45108180188490676
19	Card_dow_day_since	0.45108180188490676
20	Cardnum_unique_count_for_card_zip_14	0.43560436789481366

Figure 10: Top 20 features selected in Trial 5.

After seeing good performance for LGBM with more filter variables, I wanted to see how far the model can go when its feature pool is tighter — testing whether the stronger regularization and splitting constraints we bake into the next trial can compensate for less raw signal. I tightened the filter to 170 and applied split regularization (min_child_samples, min_split_gain, L1/L2) to eek out the last bits of performance from the algorithm.

3.6 Trial 6: Regularized LGBM + Tight Filter

- **Filter:** 170 **Wrapper:** 20
- **Model**: LGBMClassifier(n_estimators=40, num_leaves=8, max_depth=6, learning_rate=0.05, min_child_samples=50, min_split_gain=0.01, reg_alpha=0.1, reg_lambda=0.2)
- **CV:** 2-fold
- **Duration**: 00:13:11.10
- **Result**: Performance ∼0.705; saturation appx. after 12 features.

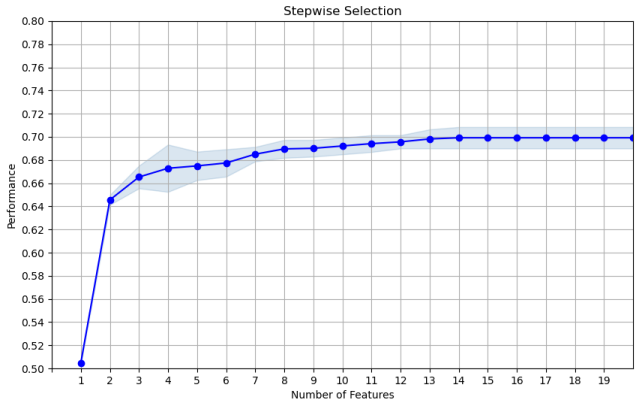

Figure 11: Performance curve for Trial 6.

wrapper order	variable	filter score
1	Cardnum_unique_count_for_card_state_1	0.5186059951957085
2	Cardnum_total_1	0.5911137143420107
3	card_zip_total_amount_1_by_60	0.34588508284282526
4	Card_dow_count_0_by_60	0.37386348222117544
5	Card_dow_unique_count_for_merch_zip_1	0.49073745525872875
6	Card_dow_total_amount_0_by_60	0.32686028495663755
7	Cardnum_max_1	0.5048046652882496
8	Card_dow_actual/toal_14	0.3370331741934217
9	Card_dow_count_14	0.43157756954042453
10	Cardnum_total_amount_1_by_60	0.4574137668325223
11	Card_dow_vdratio_0by30	0.5076936391446802
12	card_merch_total_amount_1_by_60	0.3055671810441145
13	Cardnum_variability_max_1	0.49347540089351083
14	Card_dow_actual/max_7	0.3596461246195071
15	Card_dow_day_since	0.45108180188490676
16	Card_dow_actual/toal_7	0.3941931555794056
17	Card_dow_unique_count_for_merch_zip_7	0.4468895047803746
18	Card_dow_count_0_by_60_sq	0.37386348222117544
19	Cardnum_actual/avg_0	0.3142753854346274
20	Card_dow_actual/max_14	0.2986915829231291

Figure 12: Top 20 features selected in Trial 6.

Definitely an improvement and 0.70 is achieved with saturation beyond 12th feature. But I feel the mix of variables is not the best. There is card state, zip, day of week embedded but I see day of week too much. Since the number of variables out_of_filter is very less there seems to be some loss. As discussed in class, 200-250 seems to be the sweet spot for out_of_filter. Finally, let's perform a backward selection from 200 candidate variables to 20 to optimize feature mix with the updated LGBM wrapper.

3.7 Trial 7: Backward Selection

- **Filter:** 200 **Wrapper:** 20 (backward)
- **Model**: LGBMClassifier(n_estimators=25, num_leaves=6, max_depth=5, learning_rate=0.1)
- **CV:** 2-fold
- **Duration**: 05:06:01.41
- **Result**: Performance 0.705 with a diverse 20-feature set.

Why backward may help:

- **Removes Redundancy:** Forward saturation around 12–15 features suggests many additions beyond that were marginal. Backward will peel away the least-useful ones from a richer 200-feature set compared to the 150 in the last run.
- **Better Mix:** We'll likely end up with a more balanced spread (not so heavy on day-of-week) since it optimizes the full set instead of building up.
- Time may be higher but is acceptable for one final prune.

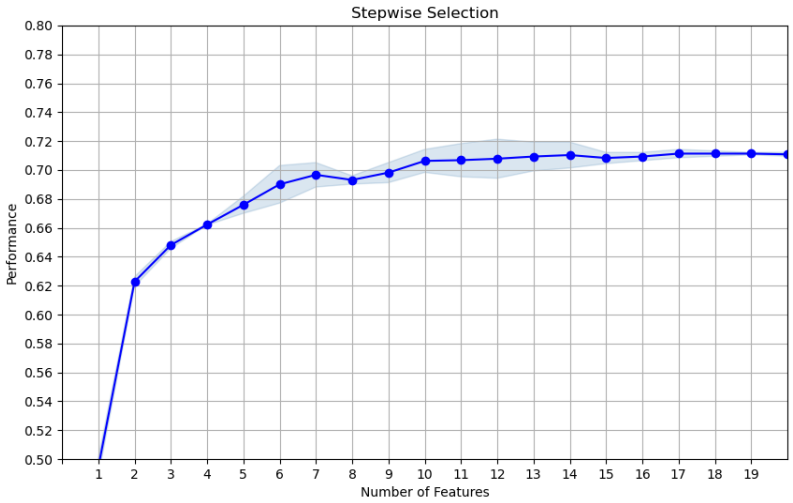

Figure 13: Performance curve for Trial 7.

The backward pass distilled the candidate pool to 20 features while maintaining ∼0.705 FDR.

- **Core Signals:** Card-level velocity and amount statistics dominate early selections.

wrapper order	variable	filter score
1	Cardnum_unique_count_for_card_state_1	0.5186059951957085
2	Cardnum_total_1	0.5911137143420107
3	Cardnum_total_0	0.5550795831550124
4	Cardnum_unique_count_for_card_state_30	0.38605957815621766
5	Cardnum_avg_3	0.30377151770438116
6	Cardnum_variability_max_1	0.49347540089351083
7	Cardnum_vdratio_0by14	0.4017270867195838
8	Cardnum_vdratio_0by30	0.4270745298741337
9	card_merch_vdratio_1by30	0.2733331094751868
10	card_merch_vdratio_0by30	0.2733331094751868
11	card_merch_day_since	0.27337064795212557
12	card_merch_vdratio_0by7	0.2733706479521257
13	card_merch_vdratio_1by14	0.2733706479521257
14	card_merch_vdratio_1by7	0.2733706479521257
15	card_zip_count_1_by_30_sq	0.27378390698567295
16	Card_dow_unique_count_for_state_des_60	0.2747260118836998
17	Card_dow_unique_count_for_Card_Merchdesc_60	0.274776063186285
18	card_zip_vdratio_0by60	0.2762725401131096
19	card_zip_vdratio_1by60	0.2762725401131096
20	card_zip_vdratio_0by30	0.27643520684651135

Figure 14: Features selected by backward selection.

- **Interactions:** Card–merchant spend ratios and recency features add complementary power.
- **Secondary Signals:** Location (ZIP) and weekday patterns appear minimally, indicating weakened marginal gain.
- **Saturation:** Performance plateaued after 10–12 features; additional features gave <0.005 lift each.

I'd choose Trial 7 (Backward Selection, LGBM 25×6@depth=5) as the standout.

Why: It delivers the highest observed FDR (0.705) while pruning away redundant signals, yielding a compact, diverse 20-feature set. By starting from the top-200 pool and removing the weakest features, it struck the best balance between predictive power and feature richness — and it corrected the over-emphasis on weekday variables seen in earlier forward runs. I feel the time it took to get there was a bit excessive but it dealt with some limitations of the forward runs.