Advertisement CTR Prediction

Fine-tune the System

DS5220 / Fall 2023 Semester

Team Members: Liyang Song, Qian Yin

Dec 10, 2023

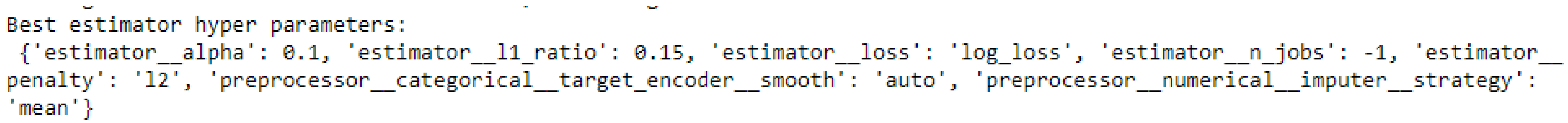
Introduction

In this part, we tune hyperparameters and classification threshold to improve model performances.

- Goal:
  - Minimize Type I error (Increase precision score)
- Five models used:
  - SGD classifier
  - Random forest classifier
  - Decision tree classifier
  - Adaboost classifier
  - Gradient boosting classifier
- Steps for each model:
  - Tune hyperparameters of composite estimators
  - Evaluate tuned composite estimators on train set and validation set
  - Check for false discoveries
  - Calculate permutation feature importance
  - Tune classification threshold
  - Find the best model and evaluate model on the test set

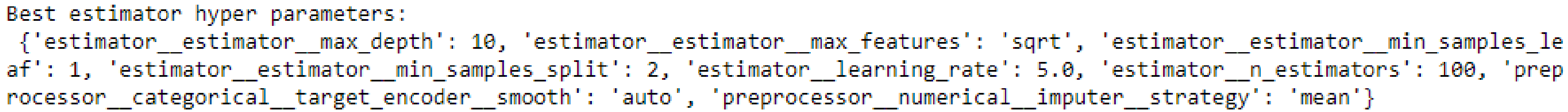
Best Estimator Hyperparameters

- SGD Classifier



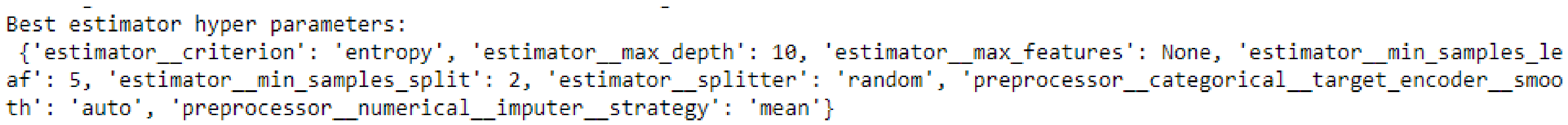
```
Best estimator hyper parameters:
 {'estimator__alpha': 0.1, 'estimator__l1_ratio': 0.15, 'estimator__loss': 'log_loss', 'estimator__n_jobs': -1, 'estimator__
penalty': 'l2', 'preprocessor__categorical__target_encoder__smooth': 'auto', 'preprocessor__numerical__imputer__strategy':
'mean'}
```

- Ada Boost Classifier



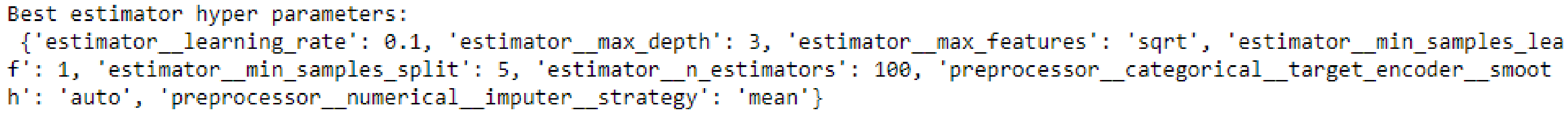
```
Best estimator hyper parameters:
 {'estimator__estimator__max_depth': 10, 'estimator__estimator__max_features': 'sqrt', 'estimator__estimator__min_samples_le
af': 1, 'estimator__estimator__min_samples_split': 2, 'estimator__learning_rate': 5.0, 'estimator__n_estimators': 100, 'prep
rocessor__categorical__target_encoder__smooth': 'auto', 'preprocessor__numerical__imputer__strategy': 'mean'}
```

- Decision Tree Classifier



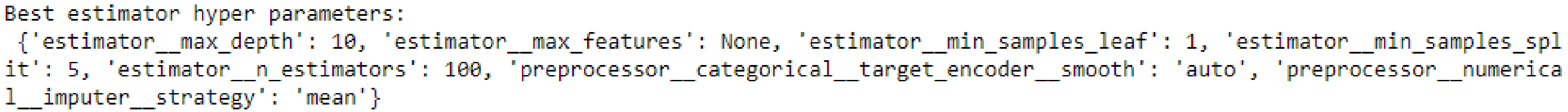
```
Best estimator hyper parameters:
 {'estimator__criterion': 'entropy', 'estimator__max_depth': 10, 'estimator__max_features': None, 'estimator__min_samples_le
af': 5, 'estimator__min_samples_split': 2, 'estimator__splitter': 'random', 'preprocessor__categorical__target_encoder__smoo
th': 'auto', 'preprocessor__numerical__imputer__strategy': 'mean'}
```

- Gradient Boosting Classifier



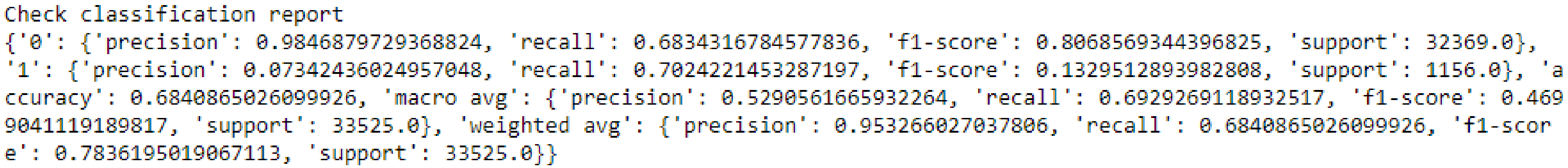
```
Best estimator hyper parameters:
 {'estimator__learning_rate': 0.1, 'estimator__max_depth': 3, 'estimator__max_features': 'sqrt', 'estimator__min_samples_lea
f': 1, 'estimator__min_samples_split': 5, 'estimator__n_estimators': 100, 'preprocessor__categorical__target_encoder__smoot
h': 'auto', 'preprocessor__numerical__imputer__strategy': 'mean'}
```

- Random forest classifier

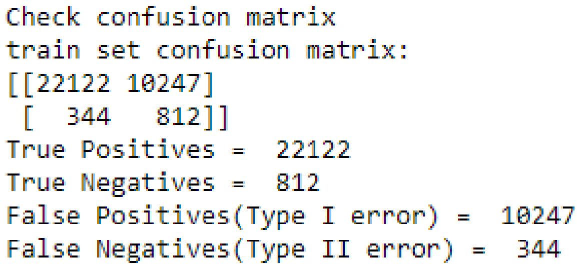


```
Best estimator hyper parameters:
 {'estimator__max_depth': 10, 'estimator__max_features': None, 'estimator__min_samples_leaf': 1, 'estimator__min_samples_spl
it': 5, 'estimator__n_estimators': 100, 'preprocessor__categorical__target_encoder__smooth': 'auto', 'preprocessor__numerica
l__imputer__strategy': 'mean'}
```

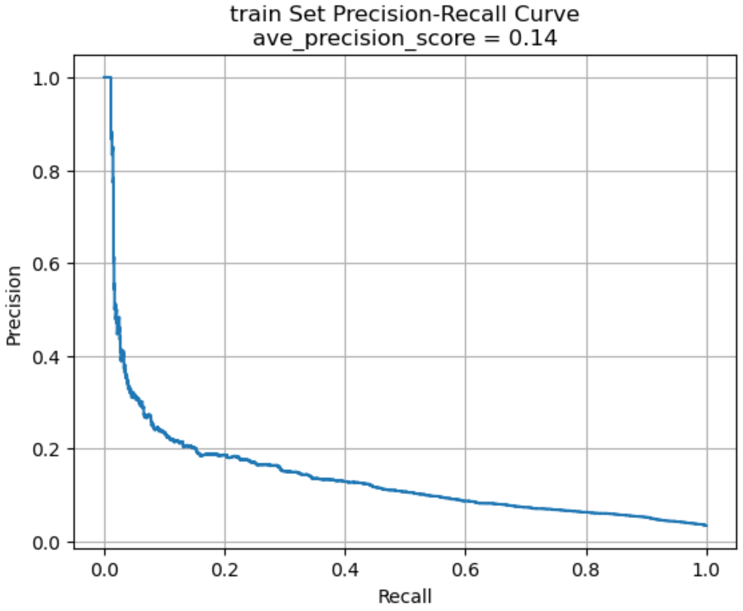
SGD classifier – Performance on train

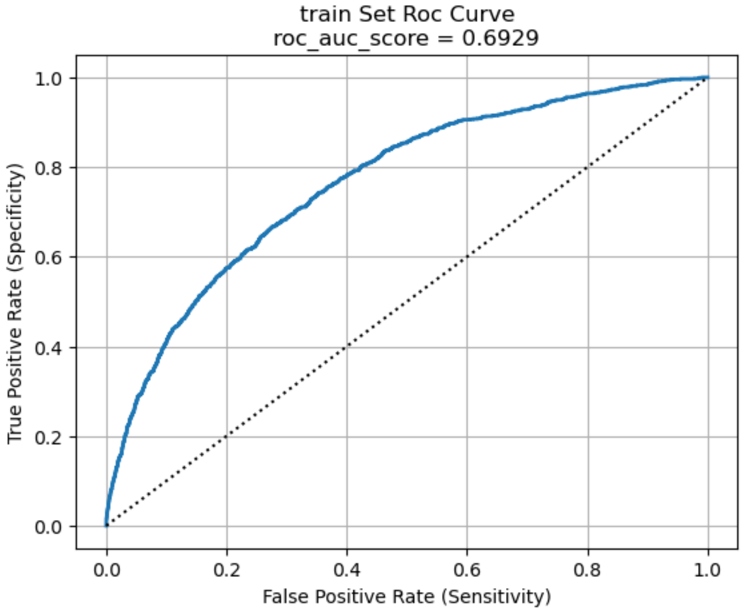


```
Check classification report
{'0': {'precision': 0.9846879729368824, 'recall': 0.6834316784577836, 'f1-score': 0.8068569344396825, 'support': 32369.0},
'1': {'precision': 0.07342436024957048, 'recall': 0.7024221453287197, 'f1-score': 0.1329512893982808, 'support': 1156.0}, 'a
ccuracy': 0.6840865026099926, 'macro avg': {'precision': 0.5290561665932264, 'recall': 0.6929269118932517, 'f1-score': 0.469
9041119189817, 'support': 33525.0}, 'weighted avg': {'precision': 0.953266027037806, 'recall': 0.6840865026099926, 'f1-scor
e': 0.7836195019067113, 'support': 33525.0}}
```



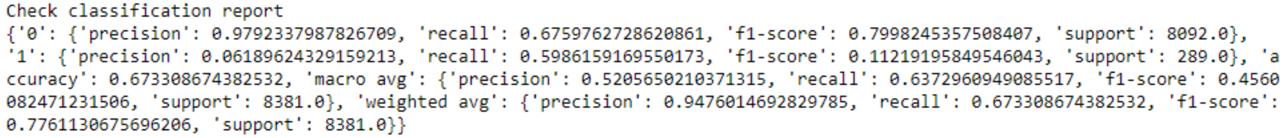
```
Check confusion matrix
train set confusion matrix:
[[22122 10247]
 [  344   812]]
True Positives =  22122
True Negatives =  812
False Positives(Type I error) =  10247
False Negatives(Type II error) =  344
```



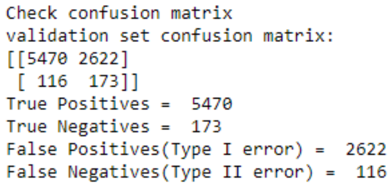


- Compared with the default model, **precision** and **ave_precision_score** increase. The number of **false positives** increases slight.

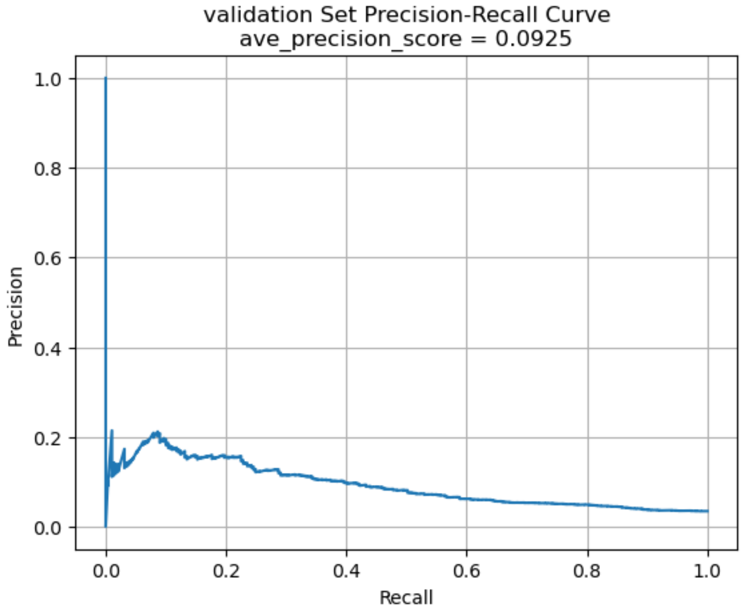
SGD classifier – Performance on validation

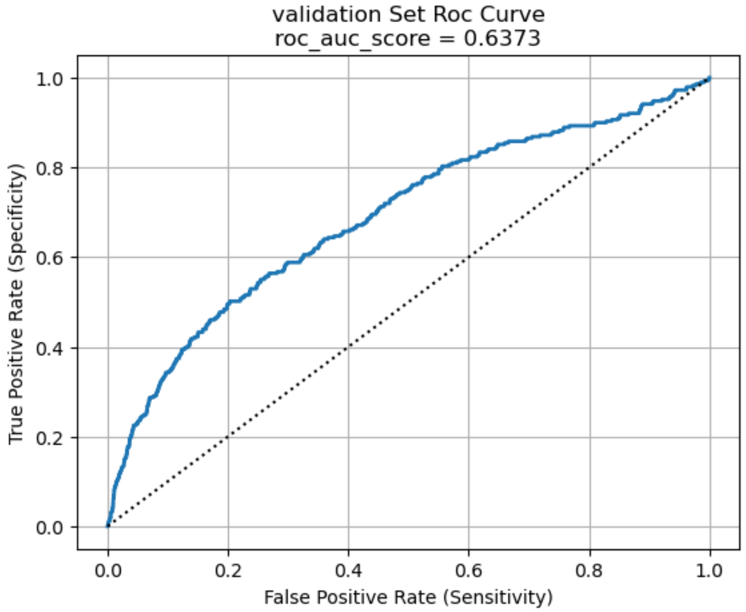


```
Check classification report
{'0': {'precision': 0.9792337987826709, 'recall': 0.6759762728620861, 'f1-score': 0.7998245357508407, 'support': 8092.0},
'1': {'precision': 0.06189624329159213, 'recall': 0.5986159169550173, 'f1-score': 0.11219195849546043, 'support': 289.0}, 'a
ccuracy': 0.673308674382532, 'macro avg': {'precision': 0.5205650210371315, 'recall': 0.6372960949085517, 'f1-score': 0.4560
082471231506, 'support': 8381.0}, 'weighted avg': {'precision': 0.9476014692829785, 'recall': 0.673308674382532, 'f1-score':
0.7761130675696206, 'support': 8381.0}}
```



```
Check confusion matrix
validation set confusion matrix:
[[5470 2622]
 [ 116  173]]
True Positives =  5470
True Negatives =  173
False Positives(Type I error) =  2622
False Negatives(Type II error) =  116
```

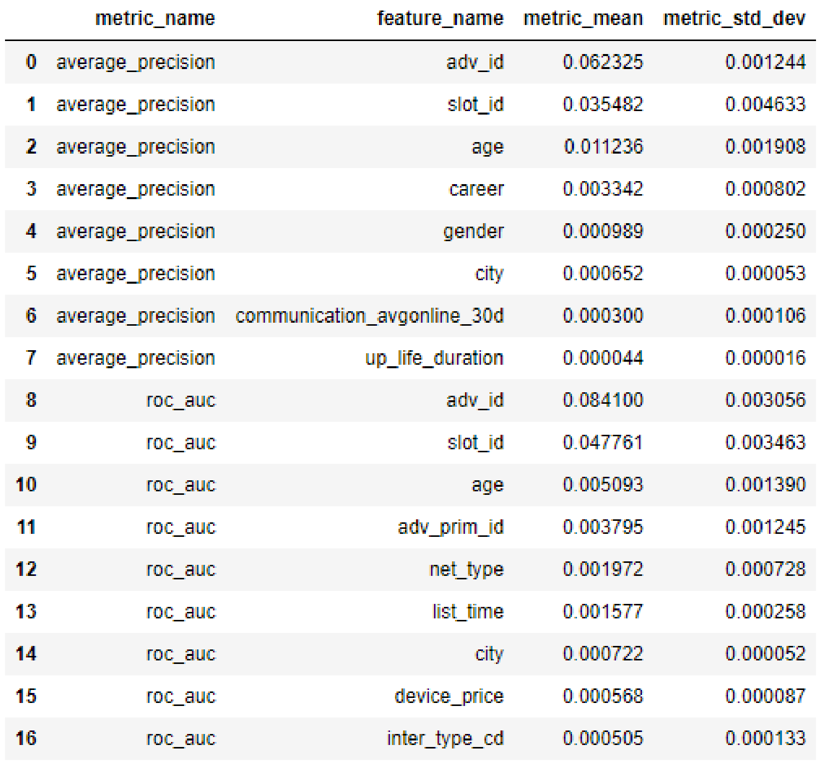




- Compared to the performance on the train set, the performance on validation set doesn't change much.

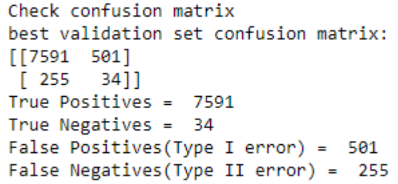
SGD classifier – Permutation Feature Importance

- This is a list of the most significant attributes.

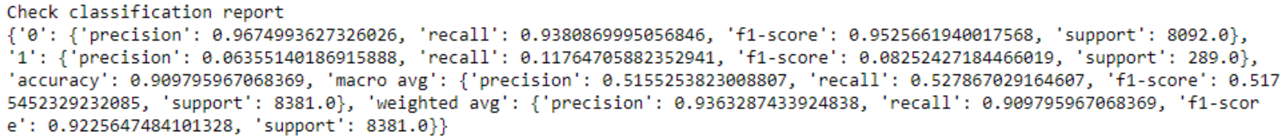


	metric_name	feature_name	metric_mean	metric_std_dev
0	average_precision	adv_id	0.062325	0.001244
1	average_precision	slot_id	0.035482	0.004633
2	average_precision	age	0.011236	0.001908
3	average_precision	career	0.003342	0.000802
4	average_precision	gender	0.000989	0.000250
5	average_precision	city	0.000652	0.000053
6	average_precision	communication_avgonline_30d	0.000300	0.000106
7	average_precision	up_life_duration	0.000044	0.000016
8	roc_auc	adv_id	0.084100	0.003056
9	roc_auc	slot_id	0.047761	0.003463
10	roc_auc	age	0.005093	0.001390
11	roc_auc	adv_prim_id	0.003795	0.001245
12	roc_auc	net_type	0.001972	0.000728
13	roc_auc	list_time	0.001577	0.000258
14	roc_auc	city	0.000722	0.000052
15	roc_auc	device_price	0.000568	0.000087
16	roc_auc	inter_type_cd	0.000505	0.000133

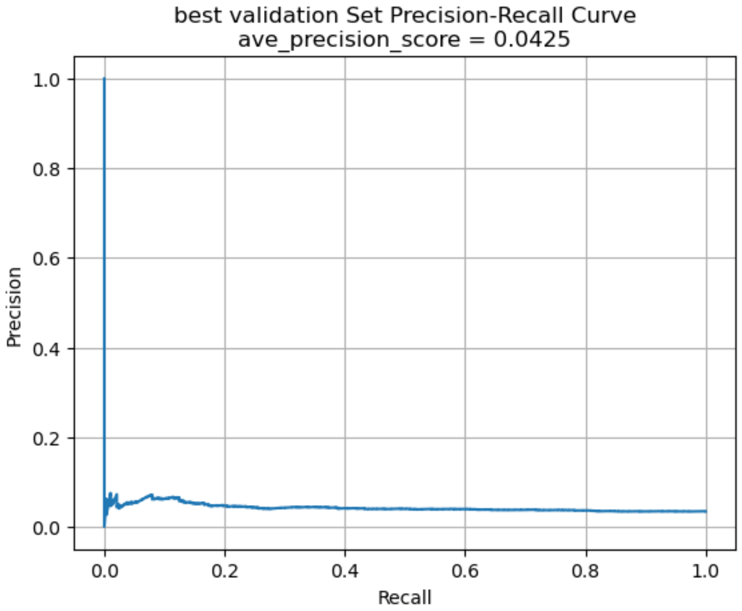
SGD classifier – Assess classification thresholds



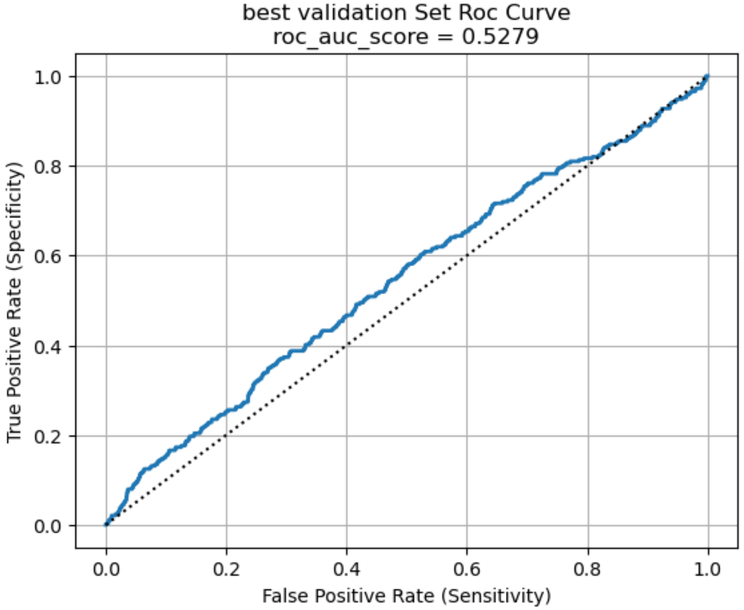
```
Check confusion matrix
best validation set confusion matrix:
[[7591  501]
 [ 255   34]]
True Positives =  7591
True Negatives =  34
False Positives(Type I error) =  501
False Negatives(Type II error) =  255
```



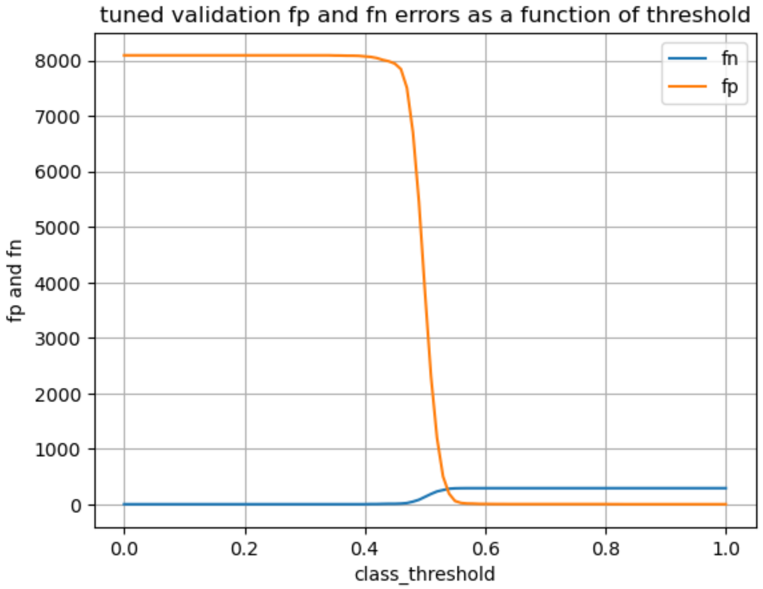
```
Check classification report
{'0': {'precision': 0.9674993627326026, 'recall': 0.9380869995056846, 'f1-score': 0.9525661940017568, 'support': 8092.0},
'1': {'precision': 0.06355140186915888, 'recall': 0.11764705882352941, 'f1-score': 0.08252427184466019, 'support': 289.0},
'accuracy': 0.909795967068369, 'macro avg': {'precision': 0.5155253823008807, 'recall': 0.527867029164607, 'f1-score': 0.517
5452329232085, 'support': 8381.0}, 'weighted avg': {'precision': 0.9363287433924838, 'recall': 0.909795967068369, 'f1-scor
e': 0.9225647484101328, 'support': 8381.0}}
```



best validation Set Precision-Recall Curve
ave_precision_score = 0.0425



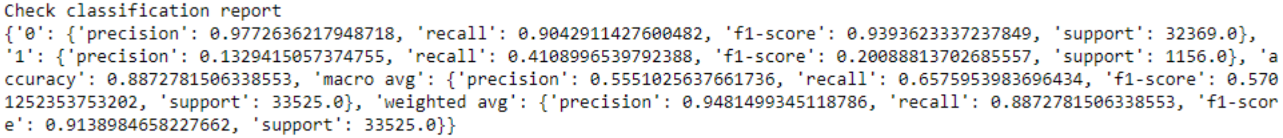
best validation Set Roc Curve
roc_auc_score = 0.5279



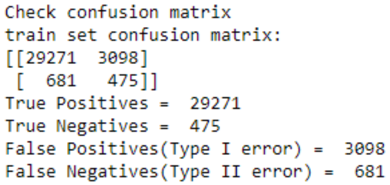
tuned validation fp and fn errors as a function of threshold

- Best threshold is 0.53. After adjusting the classification threshold, the **false positives** is much lower.

Random forest classifier – Performance on train

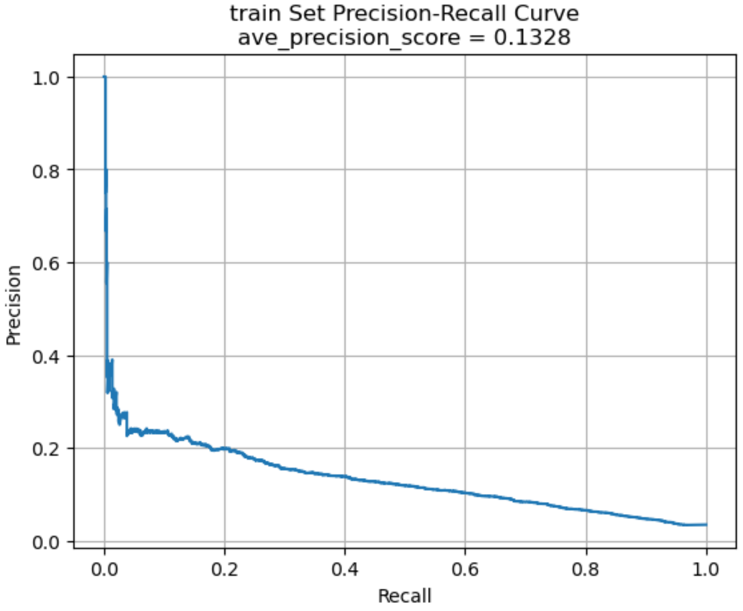


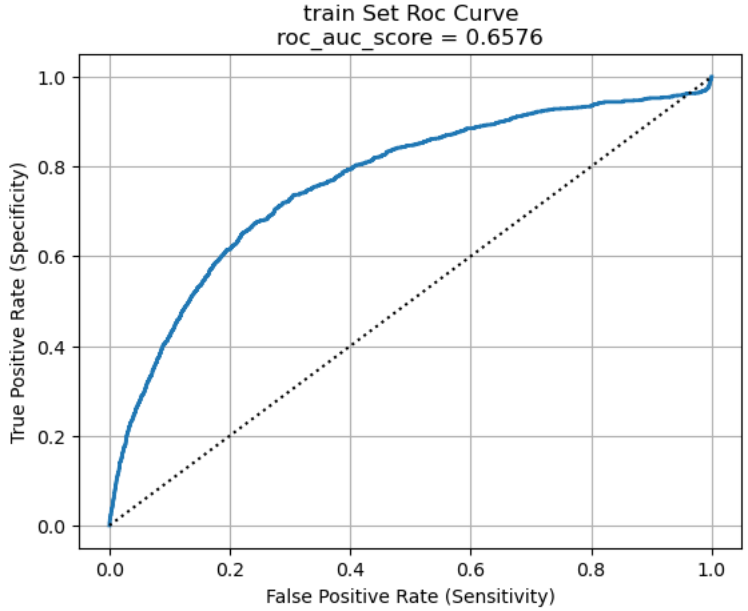
```
Check classification report
{'0': {'precision': 0.9772636217948718, 'recall': 0.9042911427600482, 'f1-score': 0.9393623337237849, 'support': 32369.0},
'1': {'precision': 0.1329415057374755, 'recall': 0.4108996539792388, 'f1-score': 0.20088813702685557, 'support': 1156.0}, 'a
ccuracy': 0.8872781506338553, 'macro avg': {'precision': 0.5551025637661736, 'recall': 0.6575953983696434, 'f1-score': 0.570
1252353753202, 'support': 33525.0}, 'weighted avg': {'precision': 0.9481499345118786, 'recall': 0.8872781506338553, 'f1-scor
e': 0.9138984658227662, 'support': 33525.0}}
```



```
Check confusion matrix
train set confusion matrix:
[[29271  3098]
 [  681   475]]
True Positives =  29271
True Negatives =  475
False Positives(Type I error) =  3098
False Negatives(Type II error) =  681
```

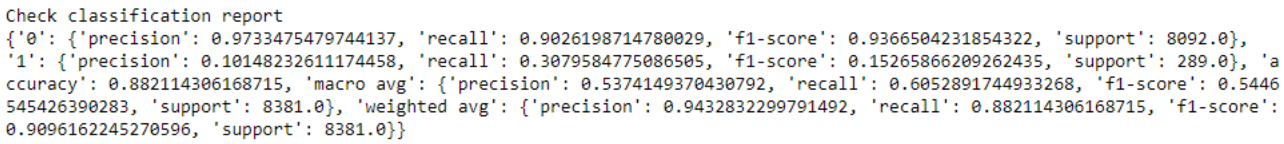
Check Precision-Recall Curve and Average Precision Score



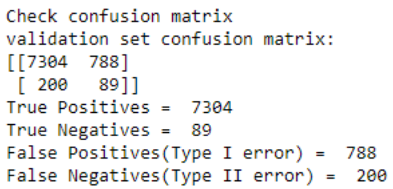


- Compared with the default model, **false positives** increase, and the **precision score** is much lower.

Random forest classifier – Performance on validation

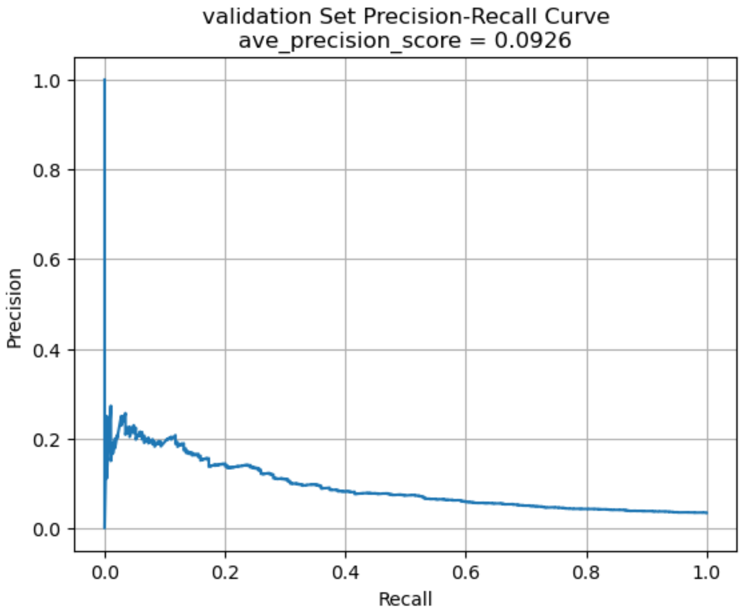


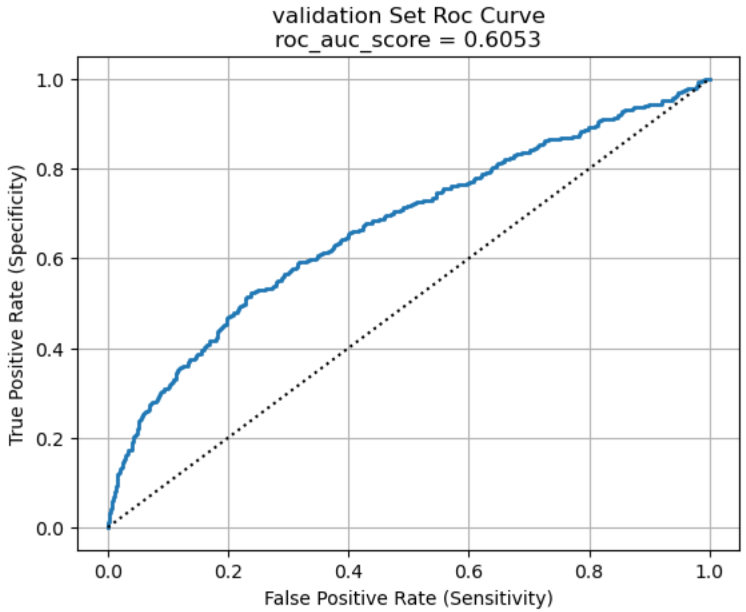
```
Check classification report
{'0': {'precision': 0.9733475479744137, 'recall': 0.9026198714780029, 'f1-score': 0.9366504231854322, 'support': 8092.0},
'1': {'precision': 0.10148232611174458, 'recall': 0.3079584775086505, 'f1-score': 0.15265866209262435, 'support': 289.0}, 'a
ccuracy': 0.882114306168715, 'macro avg': {'precision': 0.5374149370430792, 'recall': 0.6052891744933268, 'f1-score': 0.5446
545426390283, 'support': 8381.0}, 'weighted avg': {'precision': 0.9432832299791492, 'recall': 0.882114306168715, 'f1-score':
0.9096162245270596, 'support': 8381.0}}
```



```
Check confusion matrix
validation set confusion matrix:
[[7304  788]
 [ 200   89]]
True Positives =  7304
True Negatives =  89
False Positives(Type I error) =  788
False Negatives(Type II error) =  200
```

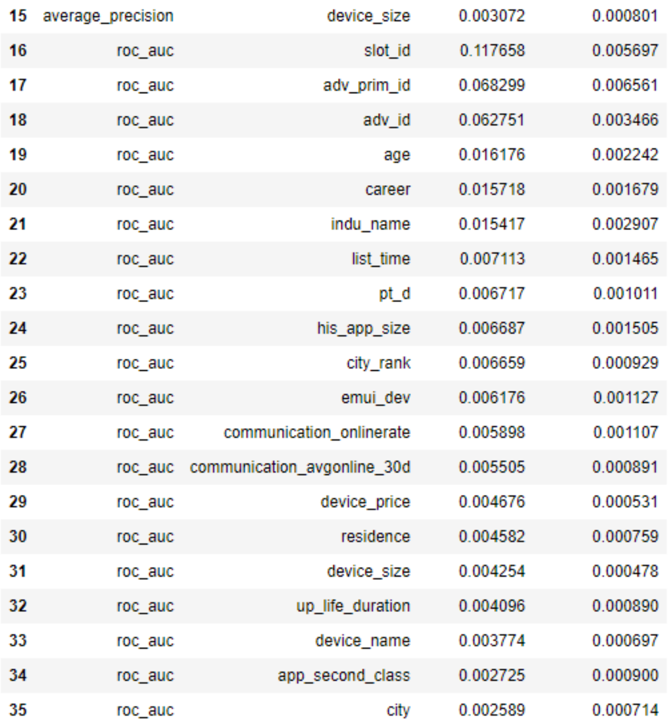
```
      Check Precision-Recall Curve and Average Precision Score
```


validation Set Precision-Recall Curve
ave_precision_score = 0.0926


validation Set Roc Curve
roc_auc_score = 0.6053

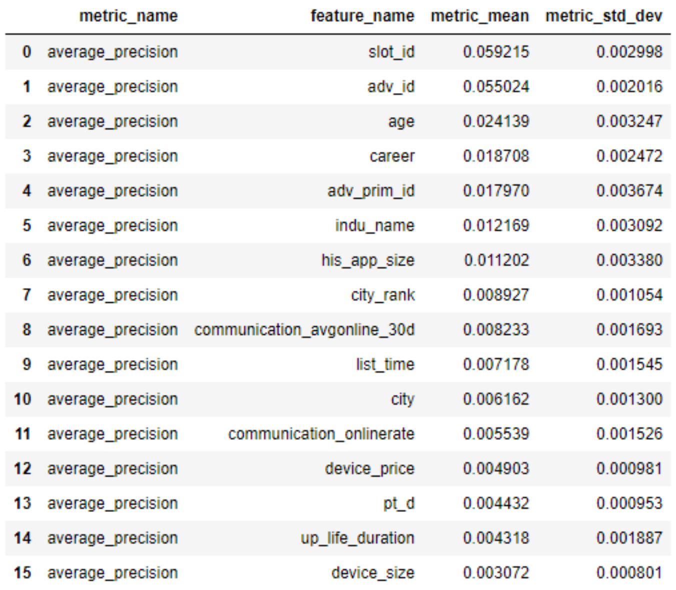
- The performance on the validation slightly decreases.

RF classifier – Permutation Feature Importance



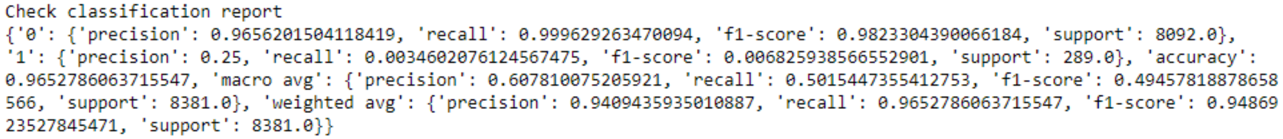
15	average_precision	device_size	0.003072	0.000801
16	roc_auc	slot_id	0.117658	0.005697
17	roc_auc	adv_prim_id	0.068299	0.006561
18	roc_auc	adv_id	0.062751	0.003466
19	roc_auc	age	0.016176	0.002242
20	roc_auc	career	0.015718	0.001679
21	roc_auc	indu_name	0.015417	0.002907
22	roc_auc	list_time	0.007113	0.001465
23	roc_auc	pt_d	0.006717	0.001011
24	roc_auc	his_app_size	0.006687	0.001505
25	roc_auc	city_rank	0.006659	0.000929
26	roc_auc	emui_dev	0.006176	0.001127
27	roc_auc	communication_onlinerate	0.005898	0.001107
28	roc_auc	communication_avgonline_30d	0.005505	0.000891
29	roc_auc	device_price	0.004676	0.000531
30	roc_auc	residence	0.004582	0.000759
31	roc_auc	device_size	0.004254	0.000478
32	roc_auc	up_life_duration	0.004096	0.000890
33	roc_auc	device_name	0.003774	0.000697
34	roc_auc	app_second_class	0.002725	0.000900
35	roc_auc	city	0.002589	0.000714

- This is a list of the most significant attributes.

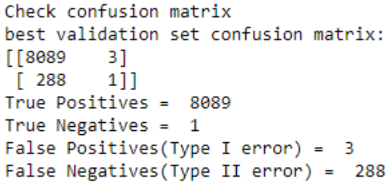


	metric_name	feature_name	metric_mean	metric_std_dev
0	average_precision	slot_id	0.059215	0.002998
1	average_precision	adv_id	0.055024	0.002016
2	average_precision	age	0.024139	0.003247
3	average_precision	career	0.018708	0.002472
4	average_precision	adv_prim_id	0.017970	0.003674
5	average_precision	indu_name	0.012169	0.003092
6	average_precision	his_app_size	0.011202	0.003380
7	average_precision	city_rank	0.008927	0.001054
8	average_precision	communication_avgonline_30d	0.008233	0.001693
9	average_precision	list_time	0.007178	0.001545
10	average_precision	city	0.006162	0.001300
11	average_precision	communication_onlinerate	0.005539	0.001526
12	average_precision	device_price	0.004903	0.000981
13	average_precision	pt_d	0.004432	0.000953
14	average_precision	up_life_duration	0.004318	0.001887
15	average_precision	device_size	0.003072	0.000801

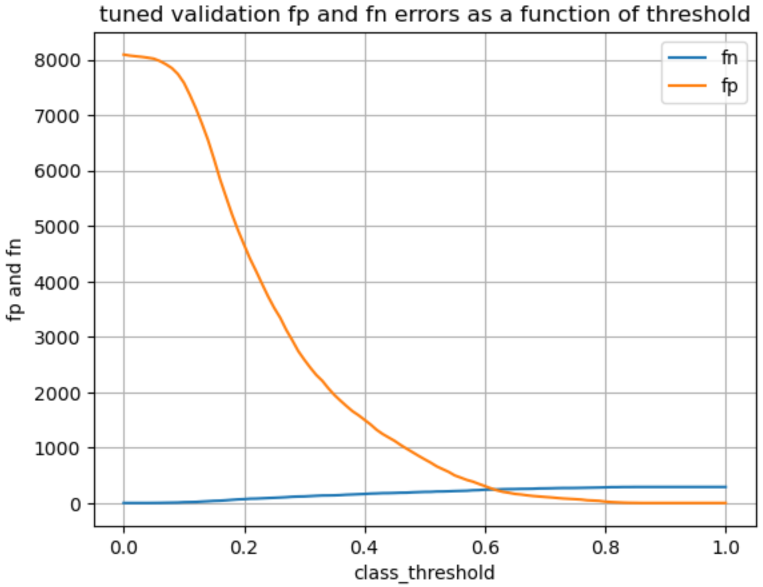
RF classifier – Assess classification thresholds



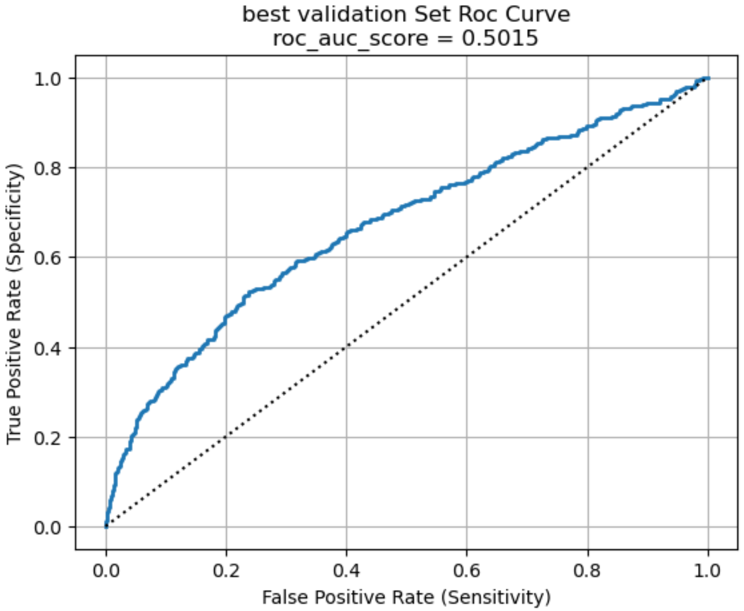
```
Check classification report
{'0': {'precision': 0.9656201504118419, 'recall': 0.999629263470094, 'f1-score': 0.9823304390066184, 'support': 8092.0},
'1': {'precision': 0.25, 'recall': 0.0034602076124567475, 'f1-score': 0.006825938566552901, 'support': 289.0}, 'accuracy':
0.9652786063715547, 'macro avg': {'precision': 0.607810075205921, 'recall': 0.5015447355412753, 'f1-score': 0.49457818878658
566, 'support': 8381.0}, 'weighted avg': {'precision': 0.9409435935010887, 'recall': 0.9652786063715547, 'f1-score': 0.94869
23527845471, 'support': 8381.0}}
```



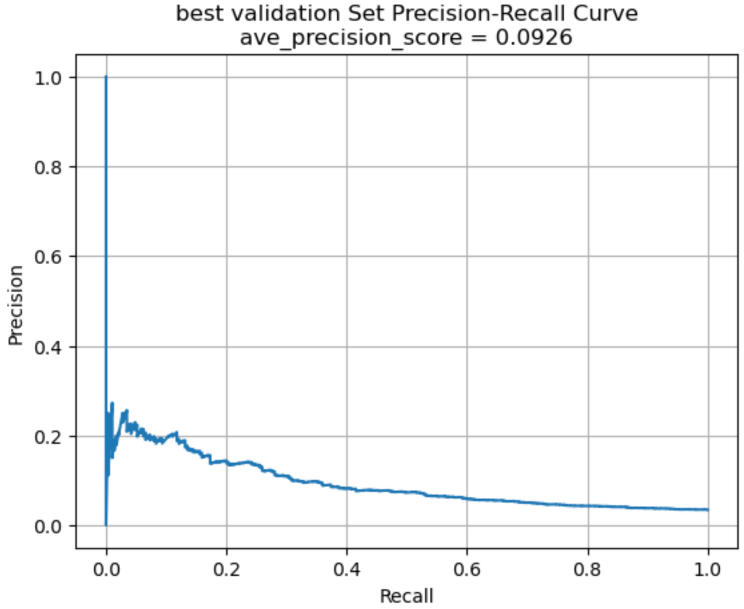
```
Check confusion matrix
best validation set confusion matrix:
[[8089    3]
 [ 288    1]]
True Positives =  8089
True Negatives =  1
False Positives(Type I error) =  3
False Negatives(Type II error) =  288
```



tuned validation fp and fn errors as a function of threshold



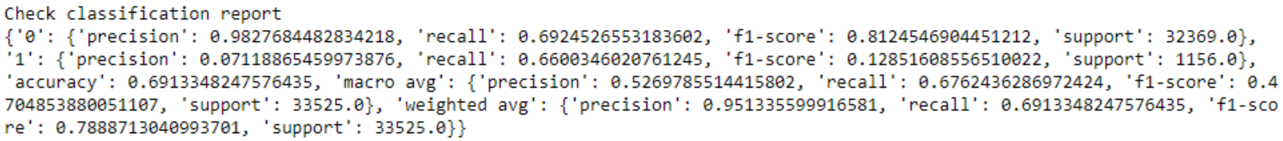
best validation Set Roc Curve
roc_auc_score = 0.5015



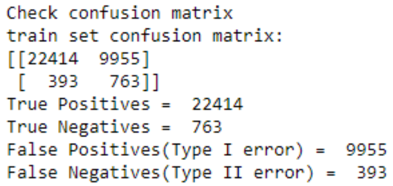
best validation Set Precision-Recall Curve
ave_precision_score = 0.0926

- **best_threshold** = 0.84. After using the best threshold, the **rate of Type I error** decreases.

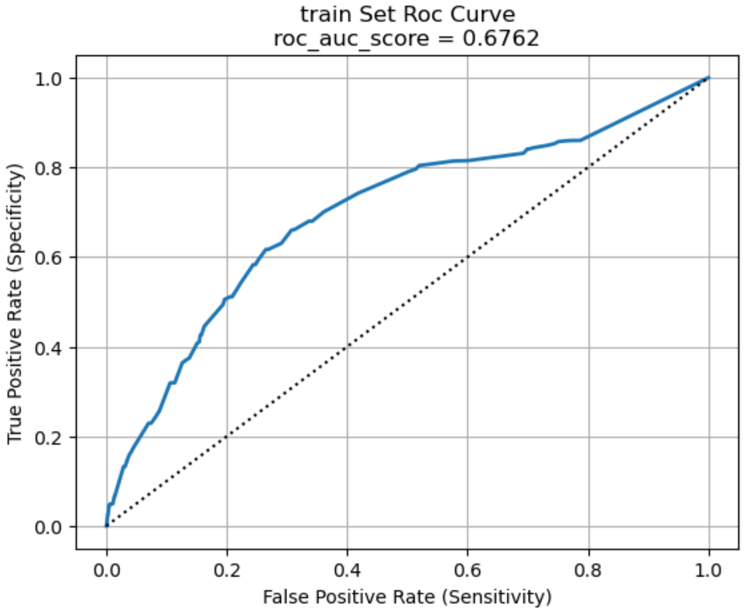
Decision Tree Classifier – Performance on train

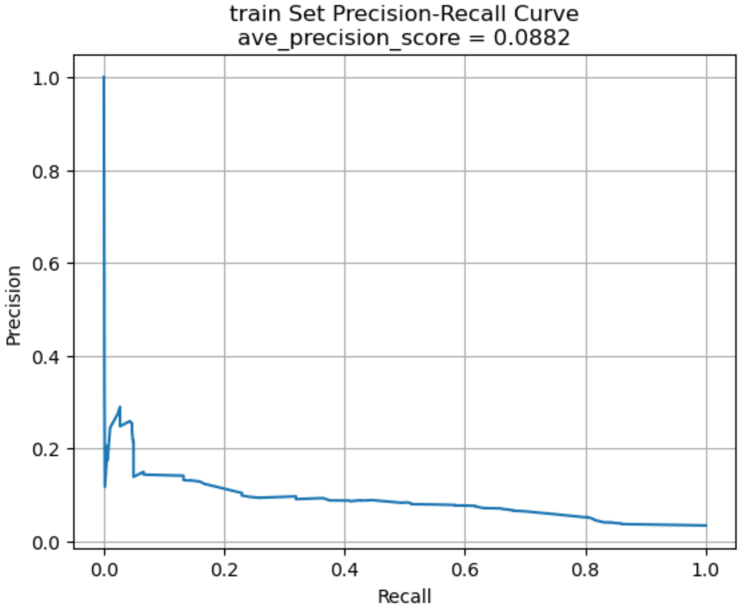


```
Check classification report
{'0': {'precision': 0.9827684482834218, 'recall': 0.6924526553183602, 'f1-score': 0.8124546904451212, 'support': 32369.0},
'1': {'precision': 0.07118865459973876, 'recall': 0.6600346020761245, 'f1-score': 0.12851608556510022, 'support': 1156.0},
'accuracy': 0.6913348247576435, 'macro avg': {'precision': 0.5269785514415802, 'recall': 0.6762436286972424, 'f1-score': 0.4
704853880051107, 'support': 33525.0}, 'weighted avg': {'precision': 0.951335599916581, 'recall': 0.6913348247576435, 'f1-sco
re': 0.7888713040993701, 'support': 33525.0}}
```



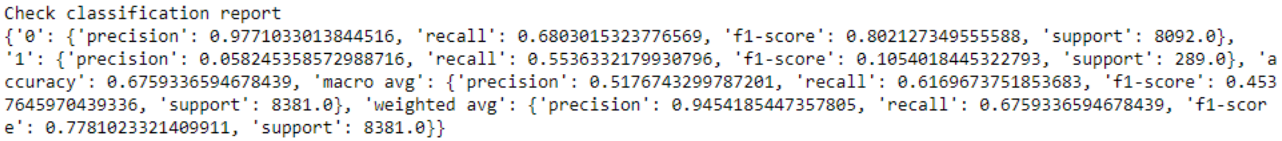
```
Check confusion matrix
train set confusion matrix:
[[22414  9955]
 [  393   763]]
True Positives =  22414
True Negatives =  763
False Positives(Type I error) =  9955
False Negatives(Type II error) =  393
```



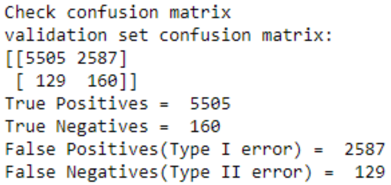


- Todo

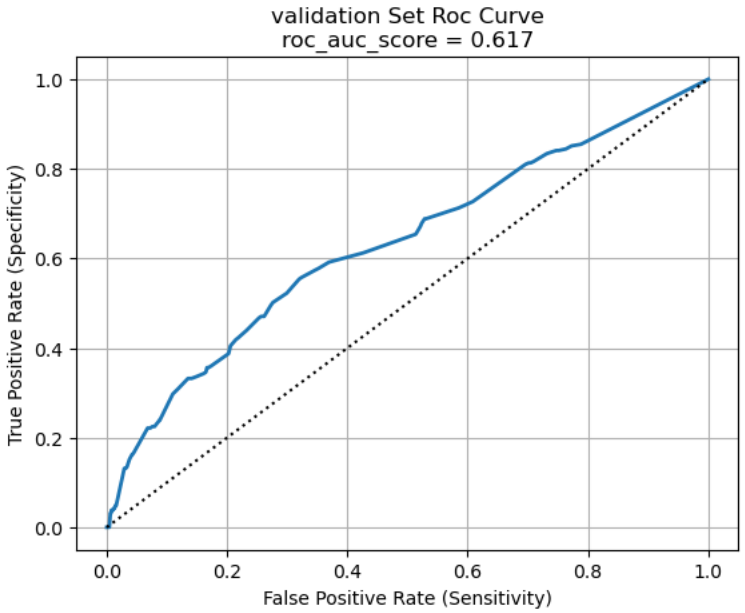
Decision Tree Classifier – Performance on validation

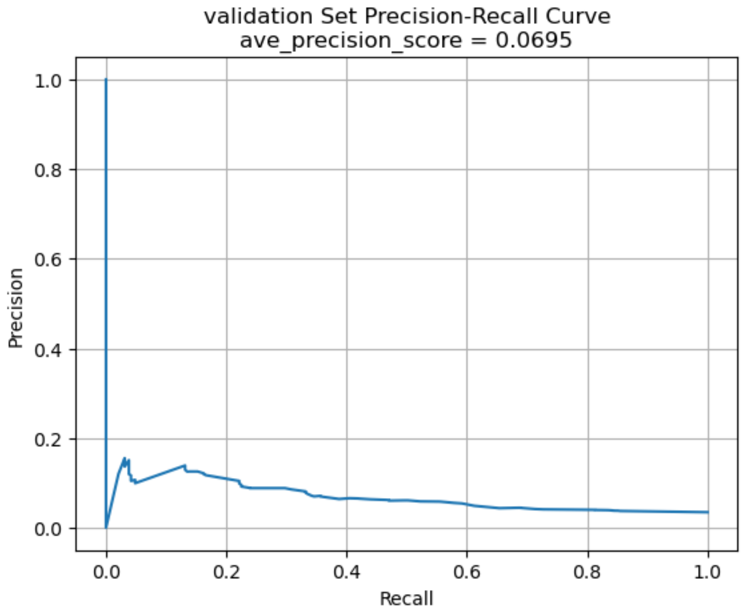


```
Check classification report
{'0': {'precision': 0.9771033013844516, 'recall': 0.6803015323776569, 'f1-score': 0.802127349555588, 'support': 8092.0},
'1': {'precision': 0.058245358572988716, 'recall': 0.5536332179930796, 'f1-score': 0.1054018445322793, 'support': 289.0}, 'a
ccuracy': 0.6759336594678439, 'macro avg': {'precision': 0.5176743299787201, 'recall': 0.6169673751853683, 'f1-score': 0.453
7645970439336, 'support': 8381.0}, 'weighted avg': {'precision': 0.9454185447357805, 'recall': 0.6759336594678439, 'f1-scor
e': 0.7781023321409911, 'support': 8381.0}}
```



```
Check confusion matrix
validation set confusion matrix:
[[5505 2587]
 [ 129  160]]
True Positives =  5505
True Negatives =  160
False Positives(Type I error) =  2587
False Negatives(Type II error) =  129
```

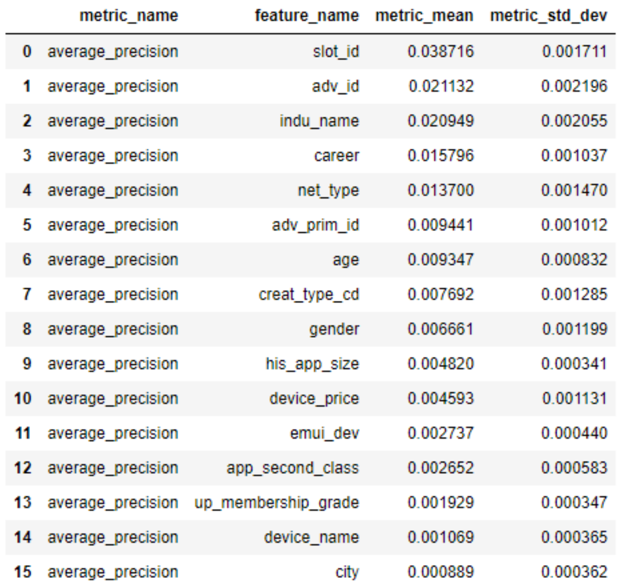


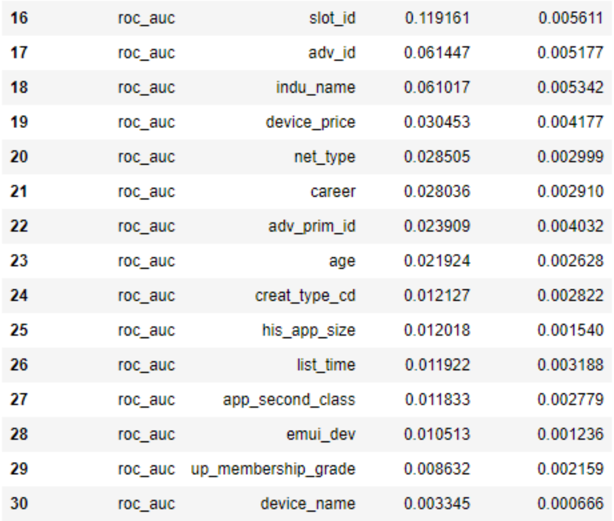


- The performances on the train set and validation set are similar.

DT Classifier– Permutation Feature Importance

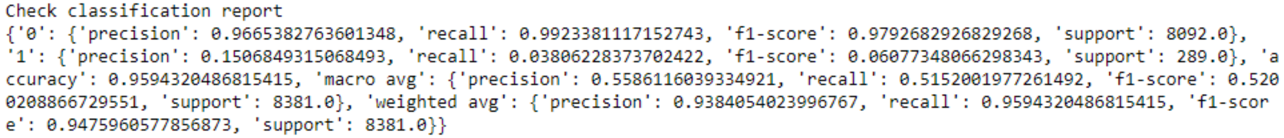
- This is a list of the most significant attributes.



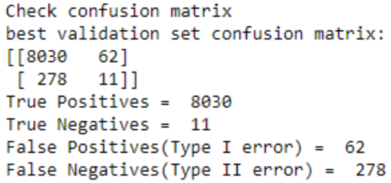


	metric_name	feature_name	metric_mean	metric_std_dev
0	average_precision	slot_id	0.038716	0.001711
1	average_precision	adv_id	0.021132	0.002196
2	average_precision	indu_name	0.020949	0.002055
3	average_precision	career	0.015796	0.001037
4	average_precision	net_type	0.013700	0.001470
5	average_precision	adv_prim_id	0.009441	0.001012
6	average_precision	age	0.009347	0.000832
7	average_precision	creat_type_cd	0.007692	0.001285
8	average_precision	gender	0.006661	0.001199
9	average_precision	his_app_size	0.004820	0.000341
10	average_precision	device_price	0.004593	0.001131
11	average_precision	emui_dev	0.002737	0.000440
12	average_precision	app_second_class	0.002652	0.000583
13	average_precision	up_membership_grade	0.001929	0.000347
14	average_precision	device_name	0.001069	0.000365
15	average_precision	city	0.000889	0.000362
16	roc_auc	slot_id	0.119161	0.005611
17	roc_auc	adv_id	0.061447	0.005177
18	roc_auc	indu_name	0.061017	0.005342
19	roc_auc	device_price	0.030453	0.004177
20	roc_auc	net_type	0.028505	0.002999
21	roc_auc	career	0.028036	0.002910
22	roc_auc	adv_prim_id	0.023909	0.004032
23	roc_auc	age	0.021924	0.002628
24	roc_auc	creat_type_cd	0.012127	0.002822
25	roc_auc	his_app_size	0.012018	0.001540
26	roc_auc	list_time	0.011922	0.003188
27	roc_auc	app_second_class	0.011833	0.002779
28	roc_auc	emui_dev	0.010513	0.001236
29	roc_auc	up_membership_grade	0.008632	0.002159
30	roc_auc	device_name	0.003345	0.000666

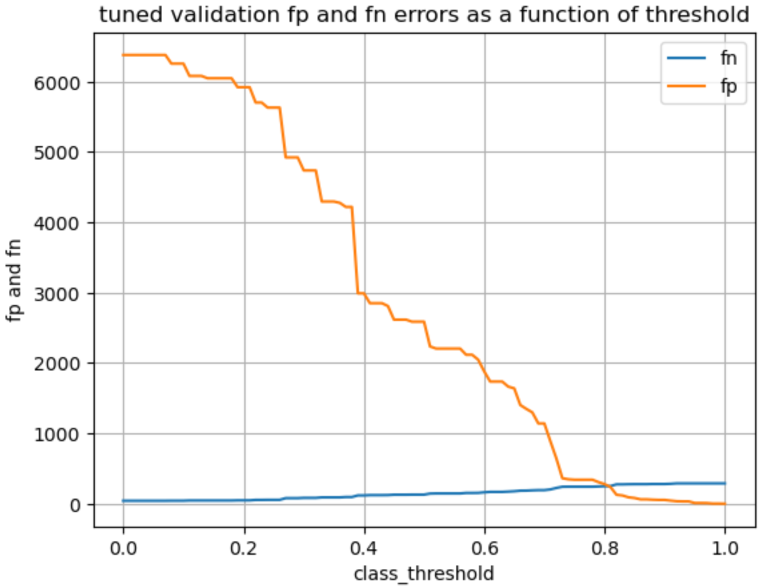
DT Classifier – Assess classification thresholds



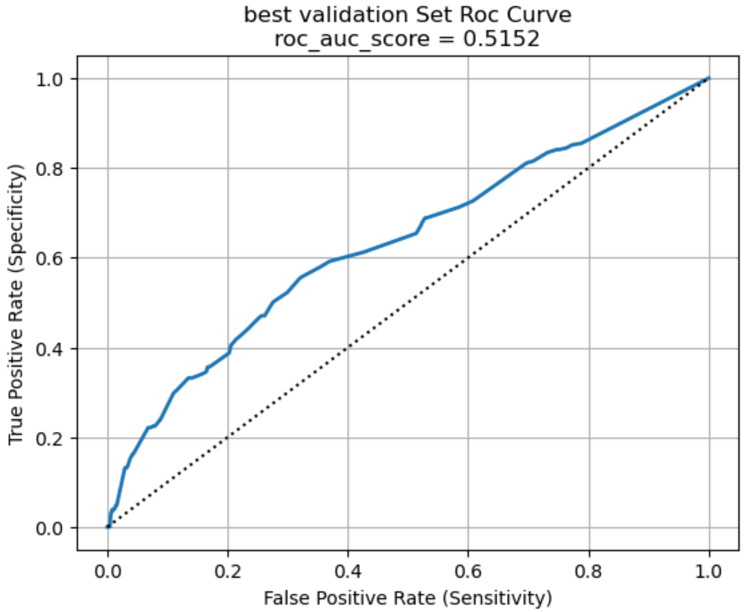
```
Check classification report
{'0': {'precision': 0.9665382763601348, 'recall': 0.9923381117152743, 'f1-score': 0.9792682926829268, 'support': 8092.0},
'1': {'precision': 0.1506849315068493, 'recall': 0.03806228373702422, 'f1-score': 0.06077348066298343, 'support': 289.0}, 'a
ccuracy': 0.9594320486815415, 'macro avg': {'precision': 0.5586116039334921, 'recall': 0.5152001977261492, 'f1-score': 0.520
0208866729551, 'support': 8381.0}, 'weighted avg': {'precision': 0.9384054023996767, 'recall': 0.9594320486815415, 'f1-scor
e': 0.9475960577856873, 'support': 8381.0}}
```



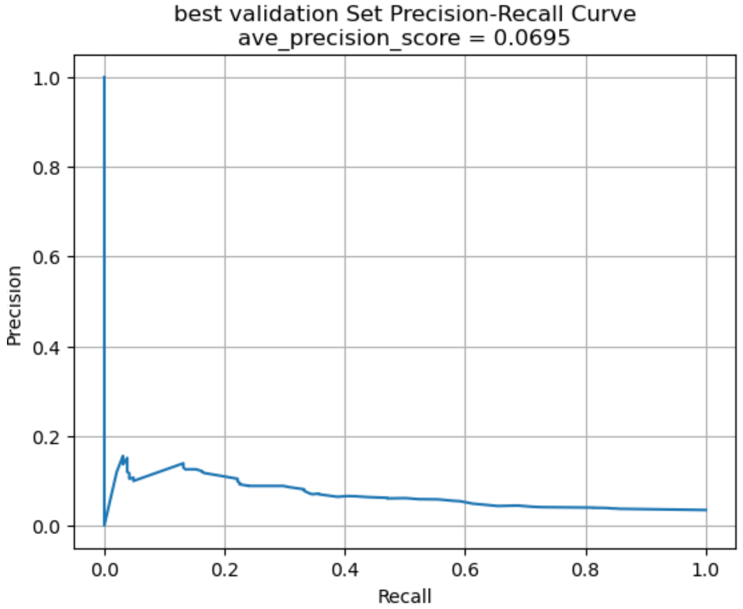
```
Check confusion matrix
best validation set confusion matrix:
[[8030   62]
 [ 278   11]]
True Positives =  8030
True Negatives =  11
False Positives(Type I error) =  62
False Negatives(Type II error) =  278
```



tuned validation fp and fn errors as a function of threshold



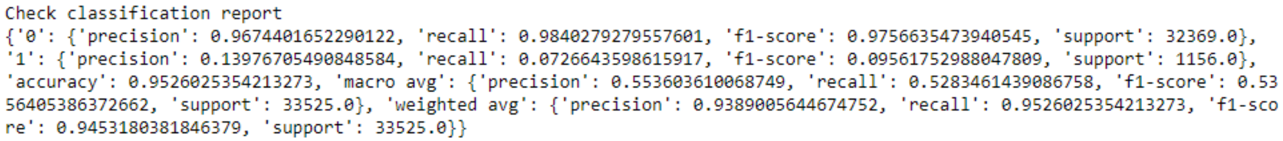
best validation Set Roc Curve
roc_auc_score = 0.5152



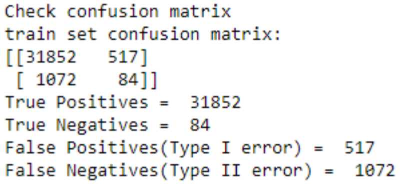
best validation Set Precision-Recall Curve
ave_precision_score = 0.0695

- **best_threshold** = 0.86. The adjustment of threshold is useful since the performance metrics are better.

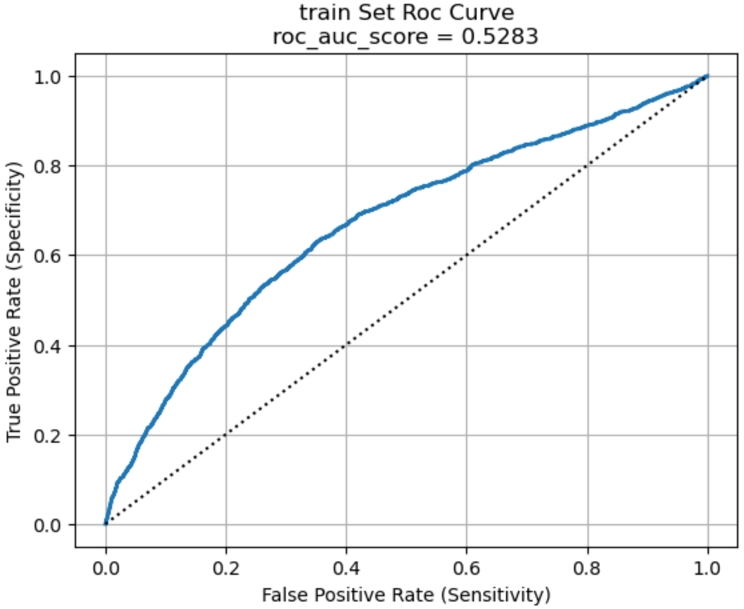
Adaboost Classifier – Performance on train

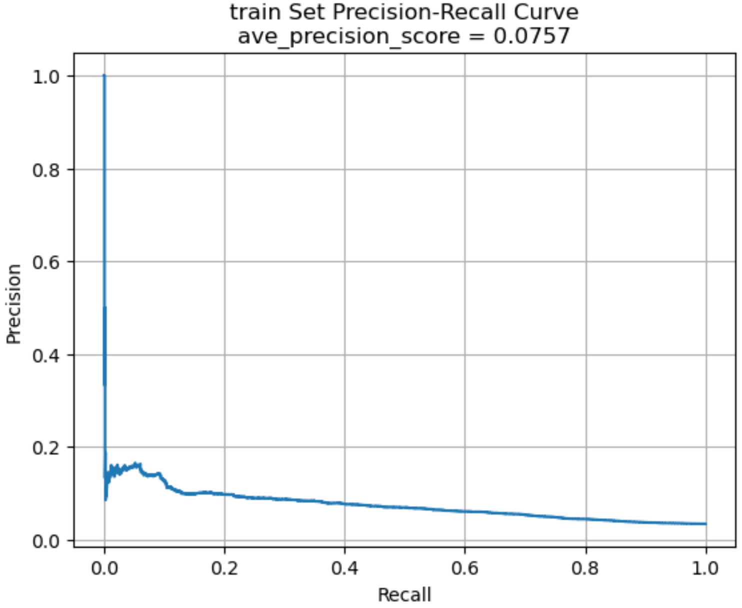


```
Check classification report
{'0': {'precision': 0.9674401652290122, 'recall': 0.9840279279557601, 'f1-score': 0.9756635473940545, 'support': 32369.0},
'1': {'precision': 0.13976705490848584, 'recall': 0.0726643598615917, 'f1-score': 0.09561752988047809, 'support': 1156.0},
'accuracy': 0.9526025354213273, 'macro avg': {'precision': 0.553603610068749, 'recall': 0.5283461439086758, 'f1-score': 0.53
56405386372662, 'support': 33525.0}, 'weighted avg': {'precision': 0.9389005644674752, 'recall': 0.9526025354213273, 'f1-sco
re': 0.9453180381846379, 'support': 33525.0}}
```



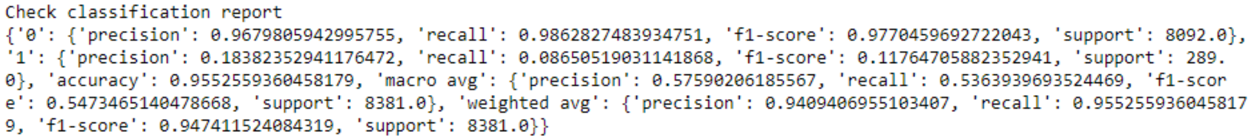
```
Check confusion matrix
train set confusion matrix:
[[31852   517]
 [ 1072    84]]
True Positives =  31852
True Negatives =  84
False Positives(Type I error) =  517
False Negatives(Type II error) =  1072
```



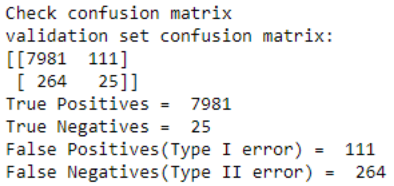


- Compared to the default model, the number of **false positives** greatly decreases although the number of **false negative** shows a subtle increase.

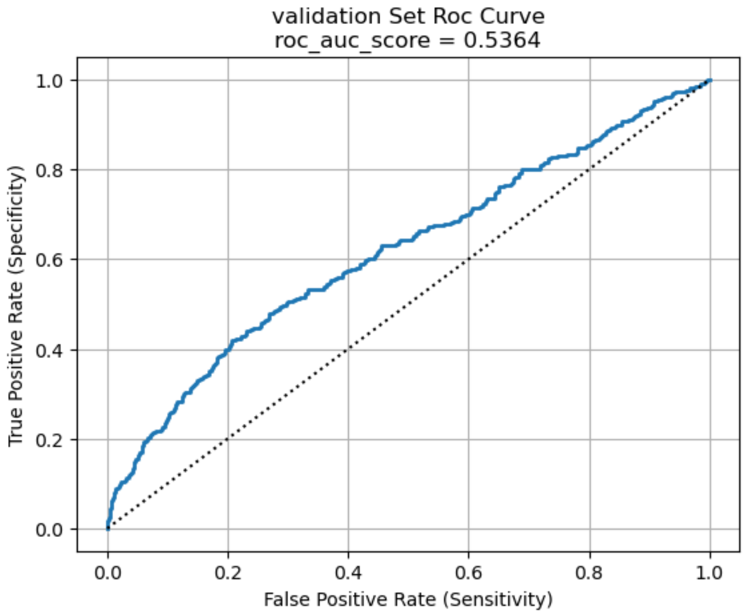
Adaboost Classifier – Performance on validation

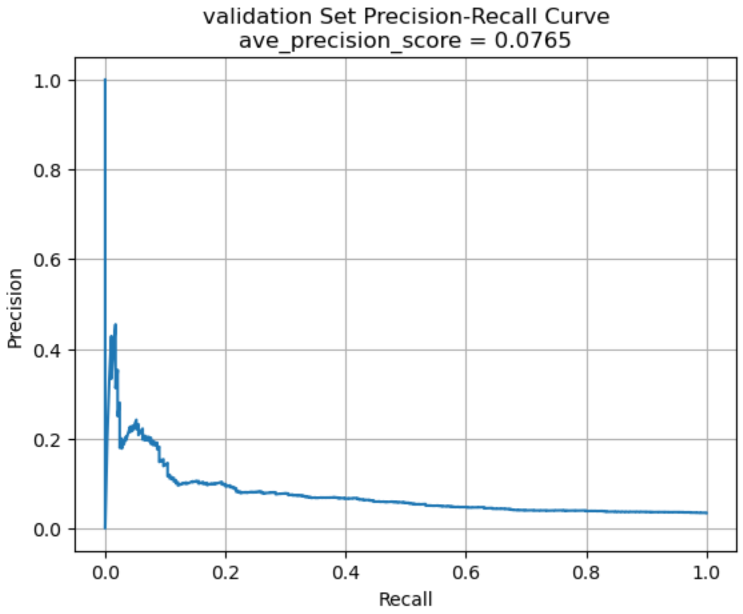


```
Check classification report
{'0': {'precision': 0.9679805942995755, 'recall': 0.9862827483934751, 'f1-score': 0.9770459692722043, 'support': 8092.0},
'1': {'precision': 0.18382352941176472, 'recall': 0.08650519031141868, 'f1-score': 0.11764705882352941, 'support': 289.
0}, 'accuracy': 0.9552559360458179, 'macro avg': {'precision': 0.57590206185567, 'recall': 0.5363939693524469, 'f1-scor
e': 0.5473465140478668, 'support': 8381.0}, 'weighted avg': {'precision': 0.9409406955103407, 'recall': 0.955255936045817
9, 'f1-score': 0.947411524084319, 'support': 8381.0}}
```



```
Check confusion matrix
validation set confusion matrix:
[[7981  111]
 [ 264   25]]
True Positives =  7981
True Negatives =  25
False Positives(Type I error) =  111
False Negatives(Type II error) =  264
```

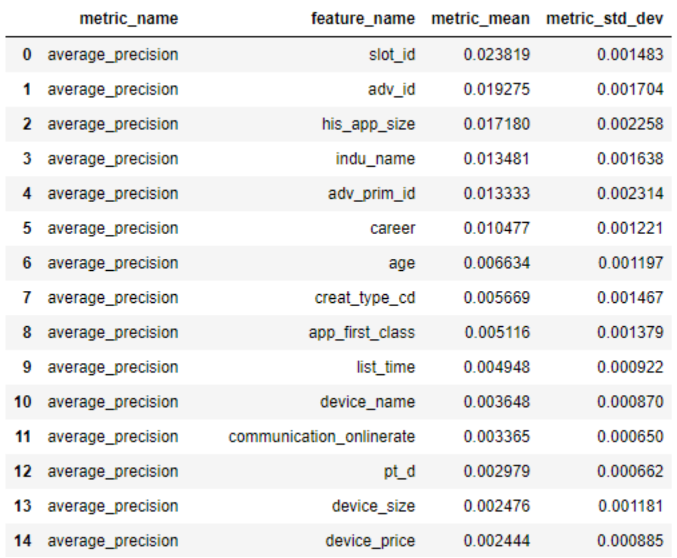




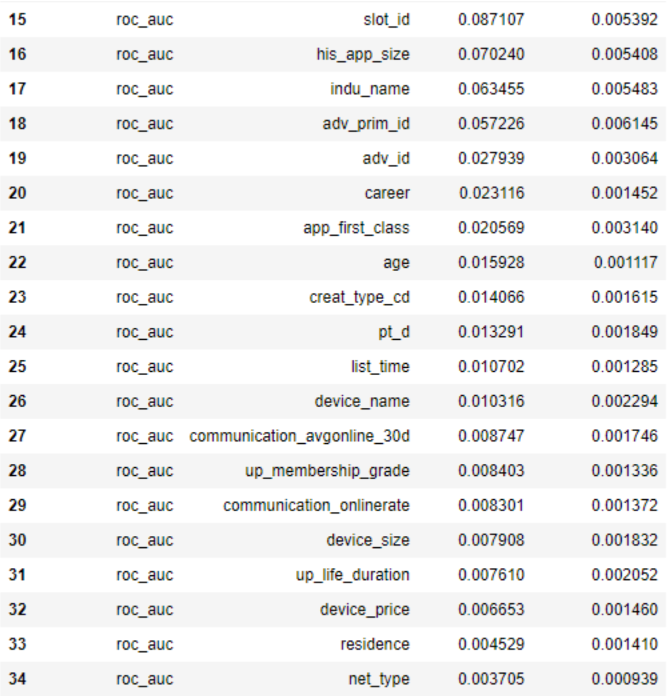
- The performance on the validation set is similar to the performance on the train set.

Adaboost Classifier– Permutation Feature Importance

- This is a list of the most significant attributes.

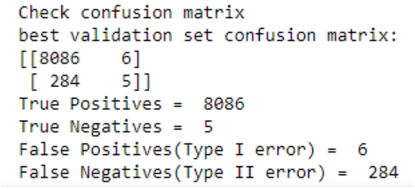


	metric_name	feature_name	metric_mean	metric_std_dev
0	average_precision	slot_id	0.023819	0.001483
1	average_precision	adv_id	0.019275	0.001704
2	average_precision	his_app_size	0.017180	0.002258
3	average_precision	indu_name	0.013481	0.001638
4	average_precision	adv_prim_id	0.013333	0.002314
5	average_precision	career	0.010477	0.001221
6	average_precision	age	0.006634	0.001197
7	average_precision	creat_type_cd	0.005669	0.001467
8	average_precision	app_first_class	0.005116	0.001379
9	average_precision	list_time	0.004948	0.000922
10	average_precision	device_name	0.003648	0.000870
11	average_precision	communication_onlinerate	0.003365	0.000650
12	average_precision	pt_d	0.002979	0.000662
13	average_precision	device_size	0.002476	0.001181
14	average_precision	device_price	0.002444	0.000885

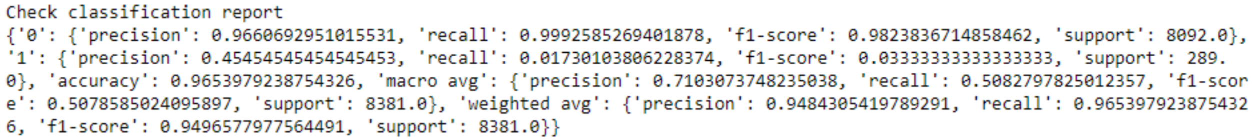


15	roc_auc	slot_id	0.087107	0.005392
16	roc_auc	his_app_size	0.070240	0.005408
17	roc_auc	indu_name	0.063455	0.005483
18	roc_auc	adv_prim_id	0.057226	0.006145
19	roc_auc	adv_id	0.027939	0.003064
20	roc_auc	career	0.023116	0.001452
21	roc_auc	app_first_class	0.020569	0.003140
22	roc_auc	age	0.015928	0.001117
23	roc_auc	creat_type_cd	0.014066	0.001615
24	roc_auc	pt_d	0.013291	0.001849
25	roc_auc	list_time	0.010702	0.001285
26	roc_auc	device_name	0.010316	0.002294
27	roc_auc	communication_avgonline_30d	0.008747	0.001746
28	roc_auc	up_membership_grade	0.008403	0.001336
29	roc_auc	communication_onlinerate	0.008301	0.001372
30	roc_auc	device_size	0.007908	0.001832
31	roc_auc	up_life_duration	0.007610	0.002052
32	roc_auc	device_price	0.006653	0.001460
33	roc_auc	residence	0.004529	0.001410
34	roc_auc	net_type	0.003705	0.000939

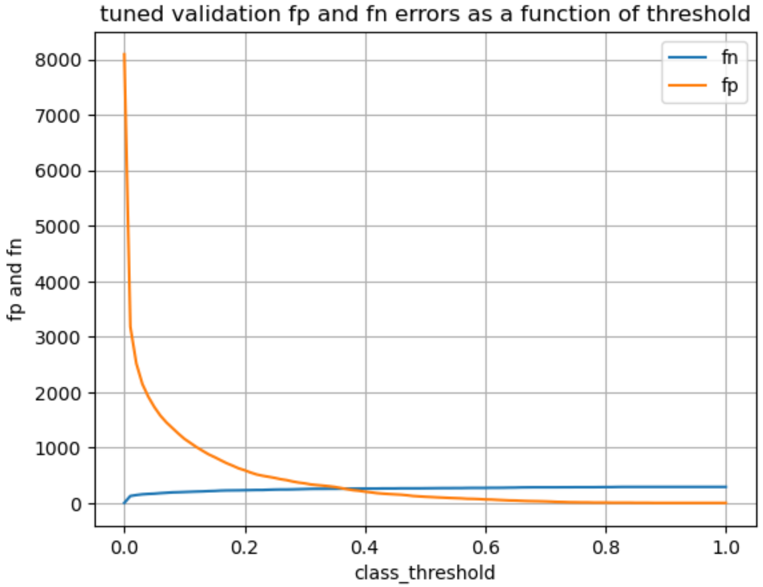
Adaboost Classifier – Assess classification thresholds

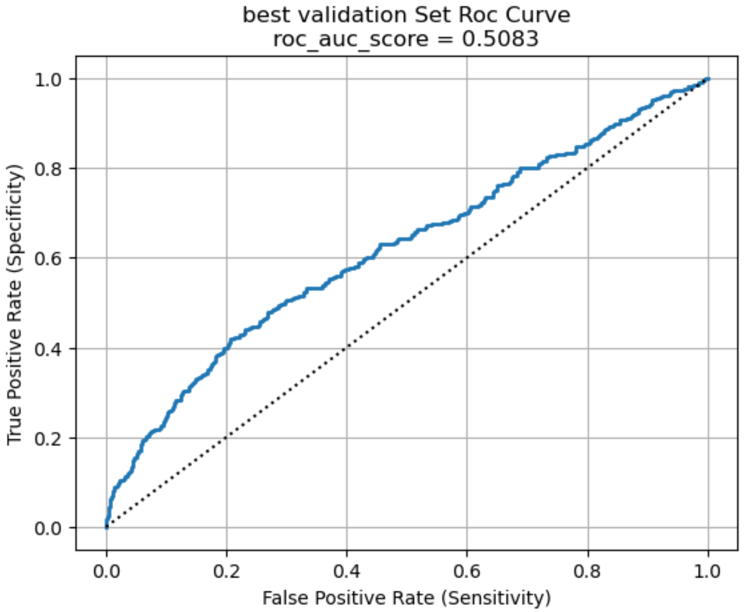


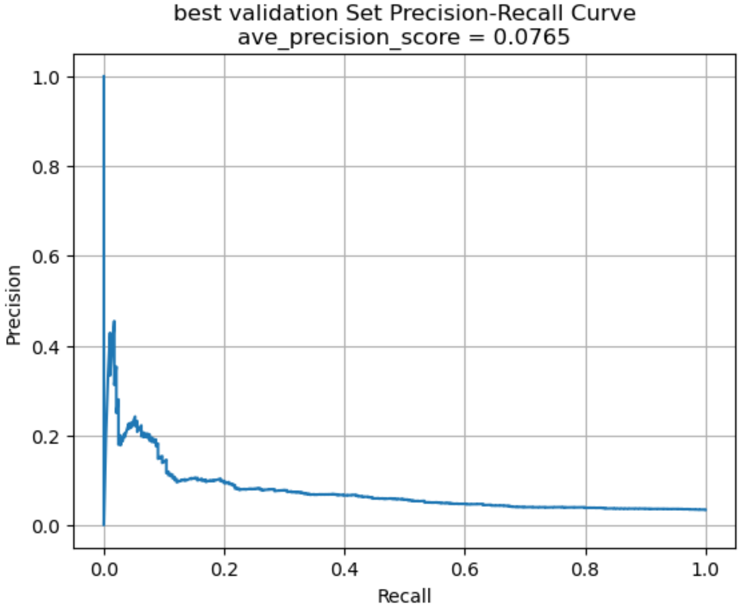
```
Check confusion matrix
best validation set confusion matrix:
[[8086    6]
 [ 284    5]]
True Positives =  8086
True Negatives =  5
False Positives(Type I error) =  6
False Negatives(Type II error) =  284
```



```
Check classification report
{'0': {'precision': 0.9660692951015531, 'recall': 0.9992585269401878, 'f1-score': 0.9823836714858462, 'support': 8092.0},
'1': {'precision': 0.45454545454545453, 'recall': 0.01730103806228374, 'f1-score': 0.03333333333333333, 'support': 289.
0}, 'accuracy': 0.9653979238754326, 'macro avg': {'precision': 0.7103073748235038, 'recall': 0.5082797825012357, 'f1-scor
e': 0.5078585024095897, 'support': 8381.0}, 'weighted avg': {'precision': 0.9484305419789291, 'recall': 0.965397923875432
6, 'f1-score': 0.9496577977564491, 'support': 8381.0}}
```

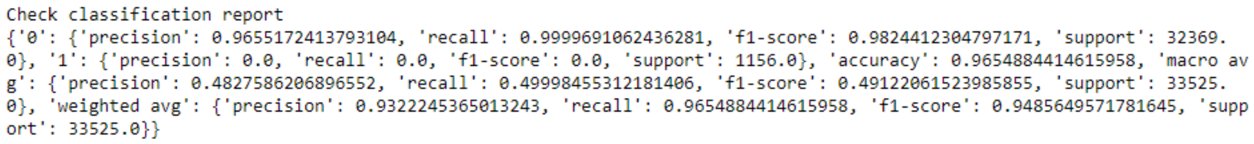




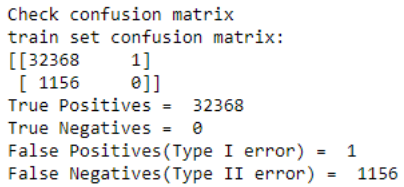


- **best_threshold** = 0.78. Adjusting threshold improves the performance since many metrics perform better.

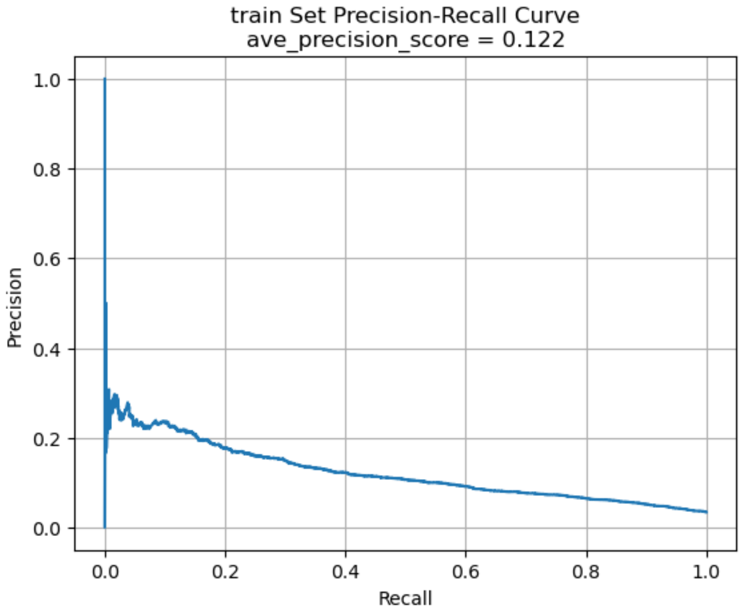
Gradient Boosting Classifier – Performance on train

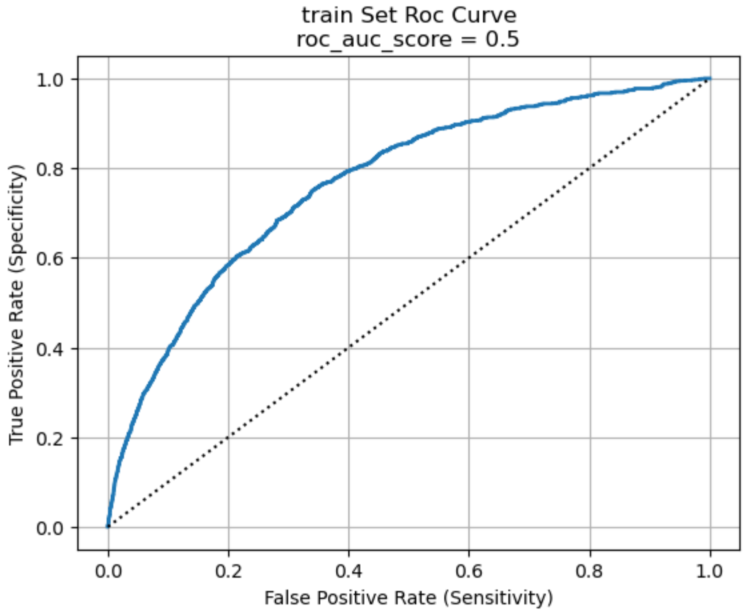


```
Check classification report
{'0': {'precision': 0.9655172413793104, 'recall': 0.9999691062436281, 'f1-score': 0.9824412304797171, 'support': 32369.
0}, '1': {'precision': 0.0, 'recall': 0.0, 'f1-score': 0.0, 'support': 1156.0}, 'accuracy': 0.9654884414615958, 'macro av
g': {'precision': 0.4827586206896552, 'recall': 0.49998455312181406, 'f1-score': 0.49122061523985855, 'support': 33525.
0}, 'weighted avg': {'precision': 0.9322245365013243, 'recall': 0.9654884414615958, 'f1-score': 0.9485649571781645, 'supp
ort': 33525.0}}
```



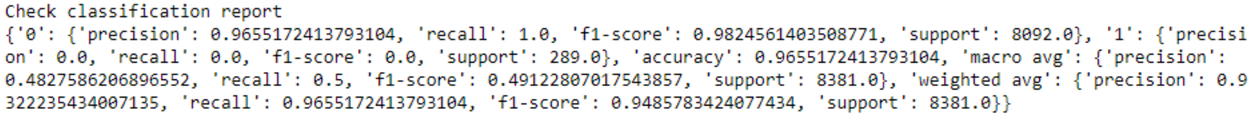
```
Check confusion matrix
train set confusion matrix:
[[32368     1]
 [ 1156     0]]
True Positives =  32368
True Negatives =  0
False Positives(Type I error) =  1
False Negatives(Type II error) =  1156
```



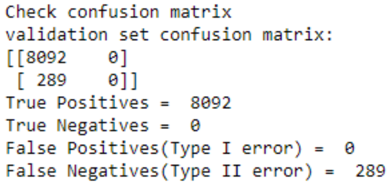


- The performances between the default model and the tuned model are similar.

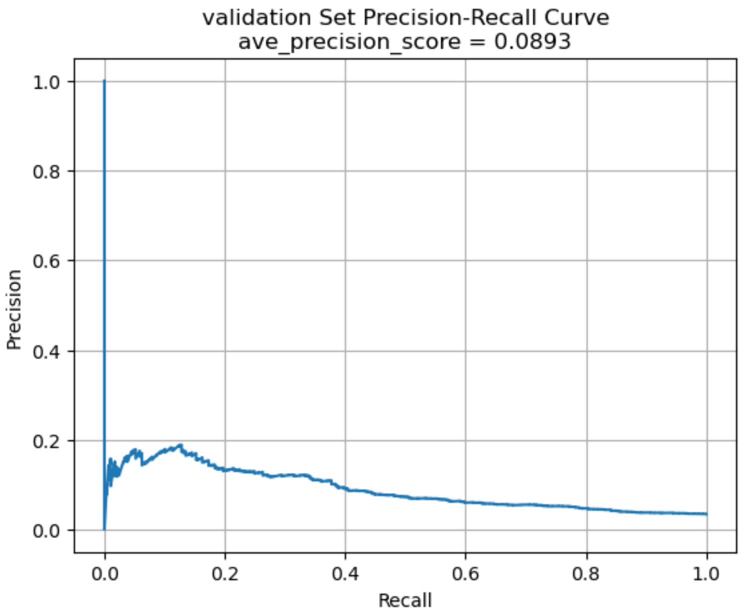
GBoosting Classifier – Performance on validation

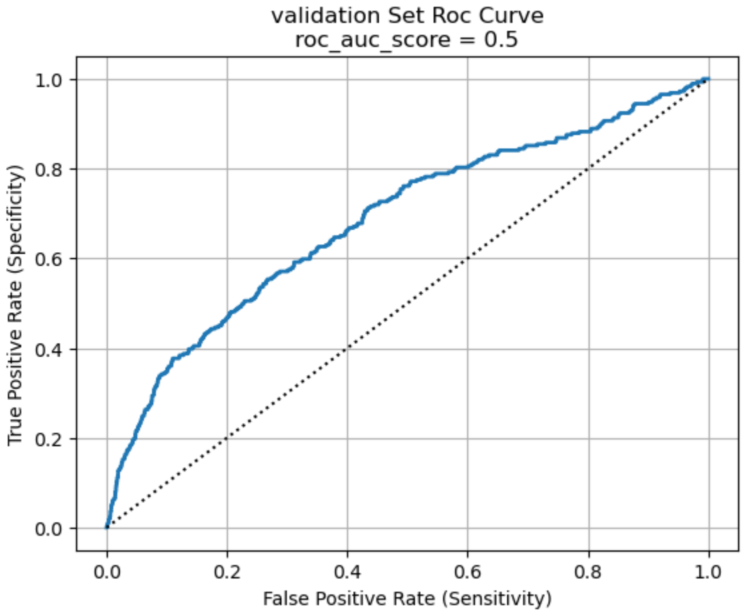


```
Check classification report
{'0': {'precision': 0.9655172413793104, 'recall': 1.0, 'f1-score': 0.9824561403508771, 'support': 8092.0}, '1': {'precisi
on': 0.0, 'recall': 0.0, 'f1-score': 0.0, 'support': 289.0}, 'accuracy': 0.9655172413793104, 'macro avg': {'precision':
0.4827586206896552, 'recall': 0.5, 'f1-score': 0.49122807017543857, 'support': 8381.0}, 'weighted avg': {'precision': 0.9
322235434007135, 'recall': 0.9655172413793104, 'f1-score': 0.9485783424077434, 'support': 8381.0}}
```



```
Check confusion matrix
validation set confusion matrix:
[[8092    0]
 [ 289    0]]
True Positives =  8092
True Negatives =  0
False Positives(Type I error) =  0
False Negatives(Type II error) =  289
```

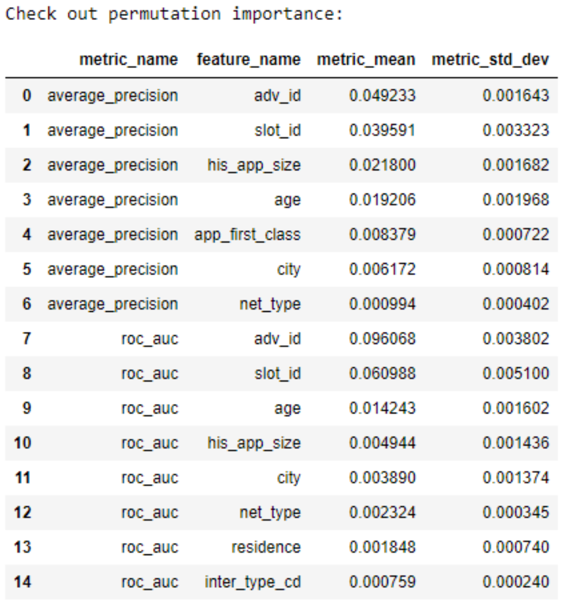




- The validation set also shows a similar performance.

GBoosting Classifier – Permutation Feature Importance

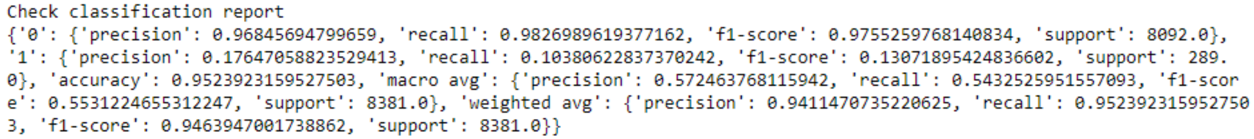
- This is a list of the most significant attributes.



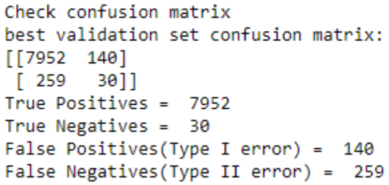
```
Check out permutation importance:
```

	metric_name	feature_name	metric_mean	metric_std_dev
0	average_precision	adv_id	0.049233	0.001643
1	average_precision	slot_id	0.039591	0.003323
2	average_precision	his_app_size	0.021800	0.001682
3	average_precision	age	0.019206	0.001968
4	average_precision	app_first_class	0.008379	0.000722
5	average_precision	city	0.006172	0.000814
6	average_precision	net_type	0.000994	0.000402
7	roc_auc	adv_id	0.096068	0.003802
8	roc_auc	slot_id	0.060988	0.005100
9	roc_auc	age	0.014243	0.001602
10	roc_auc	his_app_size	0.004944	0.001436
11	roc_auc	city	0.003890	0.001374
12	roc_auc	net_type	0.002324	0.000345
13	roc_auc	residence	0.001848	0.000740
14	roc_auc	inter_type_cd	0.000759	0.000240

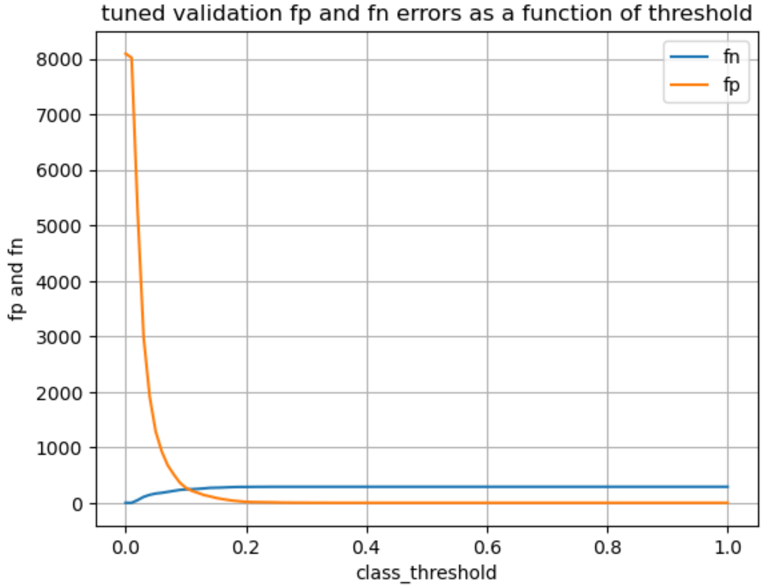
GBoosting Classifier – Assess classification thresholds

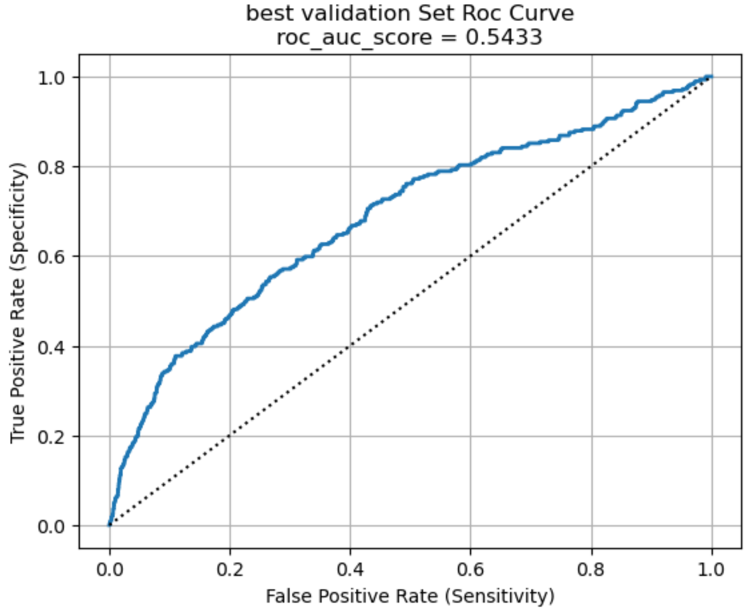


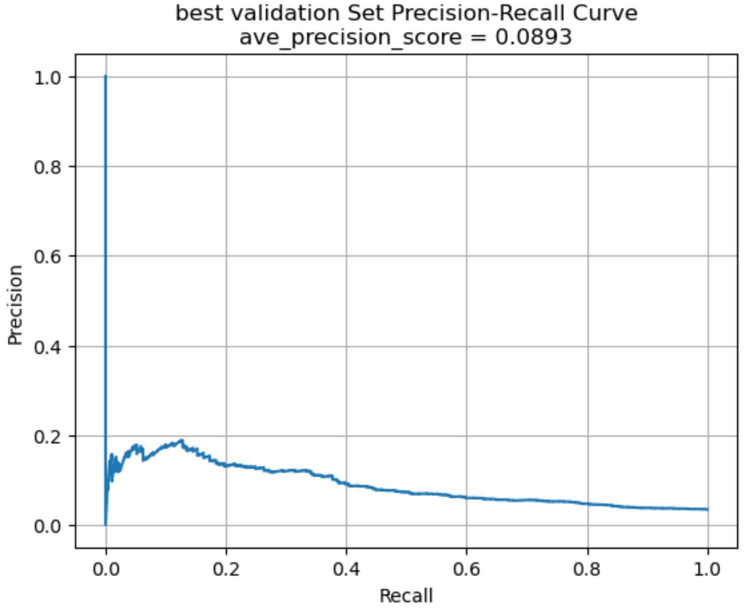
```
Check classification report
{'0': {'precision': 0.96845694799659, 'recall': 0.9826989619377162, 'f1-score': 0.9755259768140834, 'support': 8092.0},
'1': {'precision': 0.17647058823529413, 'recall': 0.10380622837370242, 'f1-score': 0.13071895424836602, 'support': 289.
0}, 'accuracy': 0.9523923159527503, 'macro avg': {'precision': 0.572463768115942, 'recall': 0.5432525951557093, 'f1-scor
e': 0.5531224655312247, 'support': 8381.0}, 'weighted avg': {'precision': 0.9411470735220625, 'recall': 0.952392315952750
3, 'f1-score': 0.9463947001738862, 'support': 8381.0}}
```



```
Check confusion matrix
best validation set confusion matrix:
[[7952  140]
 [ 259   30]]
True Positives =  7952
True Negatives =  30
False Positives(Type I error) =  140
False Negatives(Type II error) =  259
```



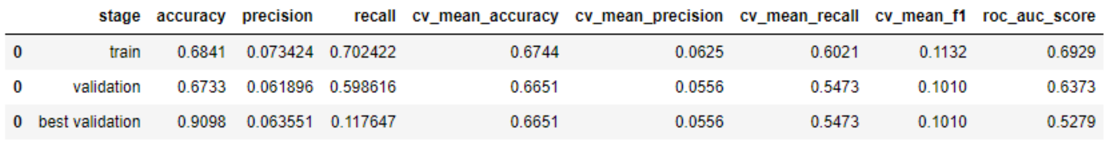




- best_threshold = 0.13. Adjusting classification thresholds changes the matrix. The number of false positives increases. The **precision score** for class 1 is increased.

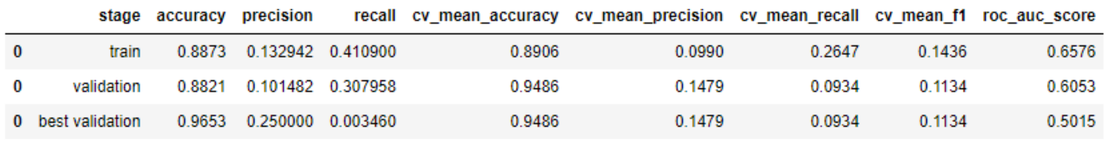
Classifiers – Comparison

- SGD Classifier



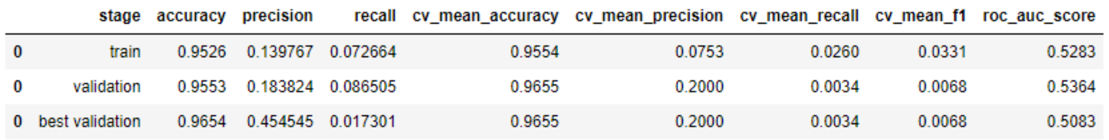
	stage	accuracy	precision	recall	cv_mean_accuracy	cv_mean_precision	cv_mean_recall	cv_mean_f1	roc_auc_score
0	train	0.6841	0.073424	0.702422	0.6744	0.0625	0.6021	0.1132	0.6929
0	validation	0.6733	0.061896	0.598616	0.6651	0.0556	0.5473	0.1010	0.6373
0	best validation	0.9098	0.063551	0.117647	0.6651	0.0556	0.5473	0.1010	0.5279

- Random Forest Classifier



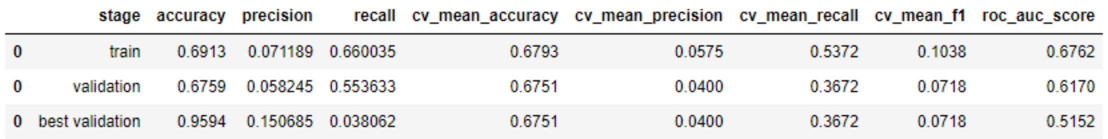
	stage	accuracy	precision	recall	cv_mean_accuracy	cv_mean_precision	cv_mean_recall	cv_mean_f1	roc_auc_score
0	train	0.8873	0.132942	0.410900	0.8906	0.0990	0.2647	0.1436	0.6576
0	validation	0.8821	0.101482	0.307958	0.9486	0.1479	0.0934	0.1134	0.6053
0	best validation	0.9653	0.250000	0.003460	0.9486	0.1479	0.0934	0.1134	0.5015

- Adaboost Classifier



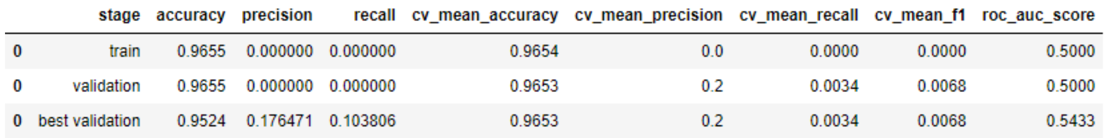
	stage	accuracy	precision	recall	cv_mean_accuracy	cv_mean_precision	cv_mean_recall	cv_mean_f1	roc_auc_score
0	train	0.9526	0.139767	0.072664	0.9554	0.0753	0.0260	0.0331	0.5283
0	validation	0.9553	0.183824	0.086505	0.9655	0.2000	0.0034	0.0068	0.5364
0	best validation	0.9654	0.454545	0.017301	0.9655	0.2000	0.0034	0.0068	0.5083

- Decision Tree Classifier



	stage	accuracy	precision	recall	cv_mean_accuracy	cv_mean_precision	cv_mean_recall	cv_mean_f1	roc_auc_score
0	train	0.6913	0.071189	0.660035	0.6793	0.0575	0.5372	0.1038	0.6762
0	validation	0.6759	0.058245	0.553633	0.6751	0.0400	0.3672	0.0718	0.6170
0	best validation	0.9594	0.150685	0.038062	0.6751	0.0400	0.3672	0.0718	0.5152

- Gradient Boosting Classifier



	stage	accuracy	precision	recall	cv_mean_accuracy	cv_mean_precision	cv_mean_recall	cv_mean_f1	roc_auc_score
0	train	0.9655	0.000000	0.000000	0.9654	0.0	0.0000	0.0000	0.5000
0	validation	0.9655	0.000000	0.000000	0.9653	0.2	0.0034	0.0068	0.5000
0	best validation	0.9524	0.176471	0.103806	0.9653	0.2	0.0034	0.0068	0.5433

- Gradient Boosting Classifier has a relatively high precision and balanced with other metrics.