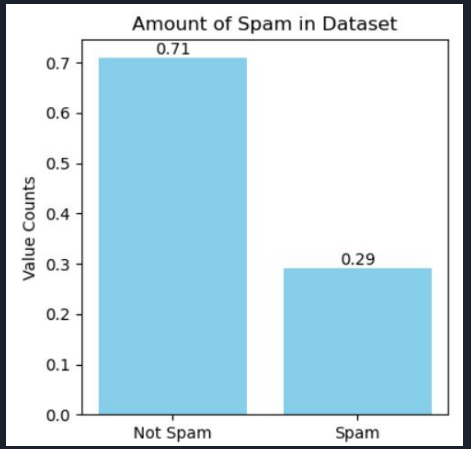
# Spam Classification

Mitchell Gallagher

# DATA




[Kaggle - Email Spam Classification]

5132 rows of emails

3002 columns of unique words with the number of times the word appears in the emails as the value

'Prediction': 0 for Not Spam, 1 for Spam

| | Email No. | the | to | ect | and | for | of | a | you | hou | ... | connevey | jay | valued | lay | infrastructure | military | allowing | ff | dry | Prediction |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | Email 1 | 0 | 0 | 1 | 0 | 0 | 0 | 2 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **1** | Email 2 | 8 | 13 | 24 | 6 | 6 | 2 | 102 | 1 | 27 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| **2** | Email 3 | 0 | 0 | 1 | 0 | 0 | 0 | 8 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **3** | Email 4 | 0 | 5 | 22 | 0 | 5 | 1 | 51 | 2 | 10 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **4** | Email 5 | 7 | 6 | 17 | 1 | 5 | 2 | 57 | 0 | 9 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |

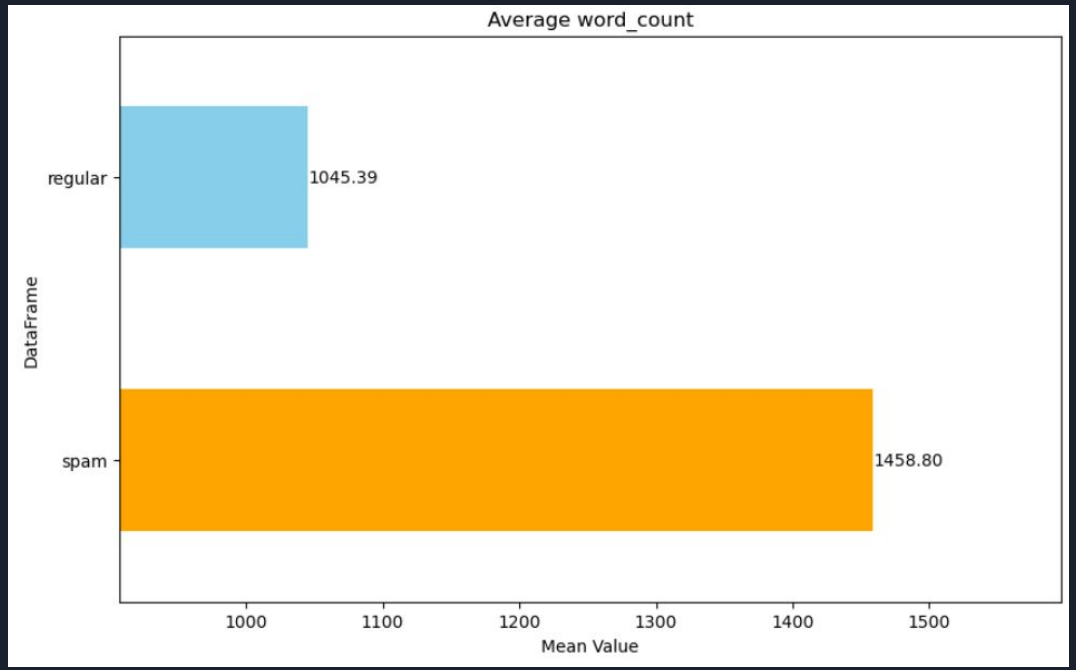
# Word Count

By counting up each rows total value according to number of each word used per email:

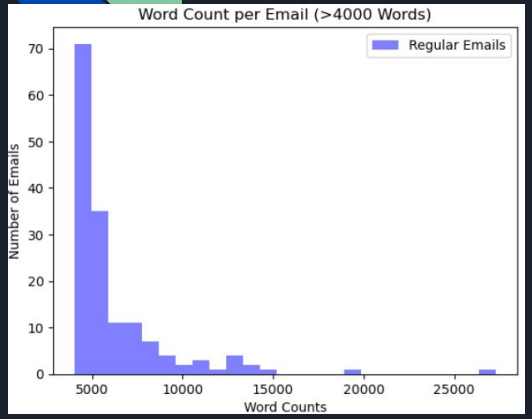
Spam emails contain 50% higher amount of words than regular emails.

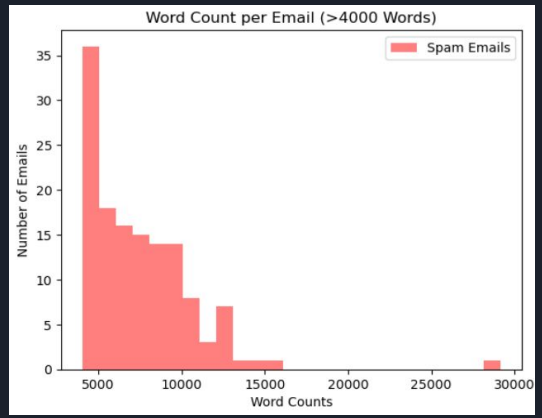
| REG EMAILS | |
|---|---|
| count | 3672.00 |
| mean | 1045.39 |
| std | 1478.04 |
| min | 21.00 |
| 25% | 244.00 |
| 50% | 551.50 |
| 75% | 1293.75 |
| max | 27319.00 |

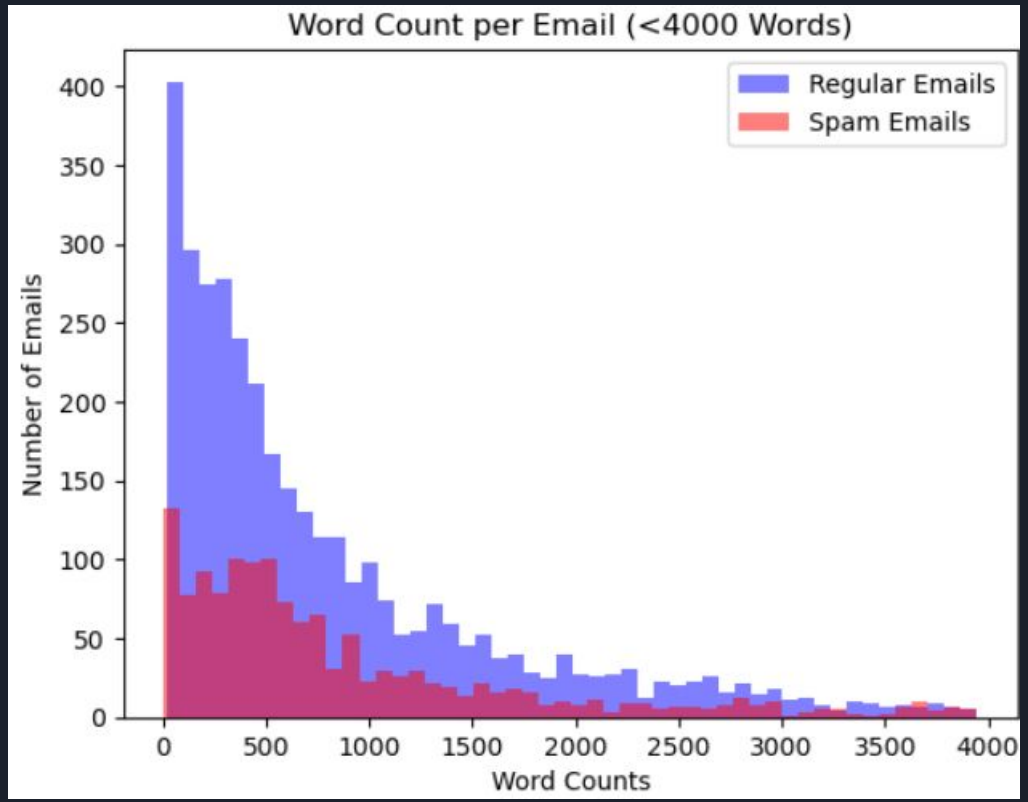
| SPAM EMAILS | |
|---|---|
| count | 1500.00 |
| mean | 1458.79 |
| std | 2283.84 |
| min | 8.00 |
| 25% | 316.00 |
| 50% | 632.00 |
| 75% | 1506.00 |
| max | 29178.00 |



Similar Distributions, Increased spam frequency (relative) at 5,000 - 10,000 words



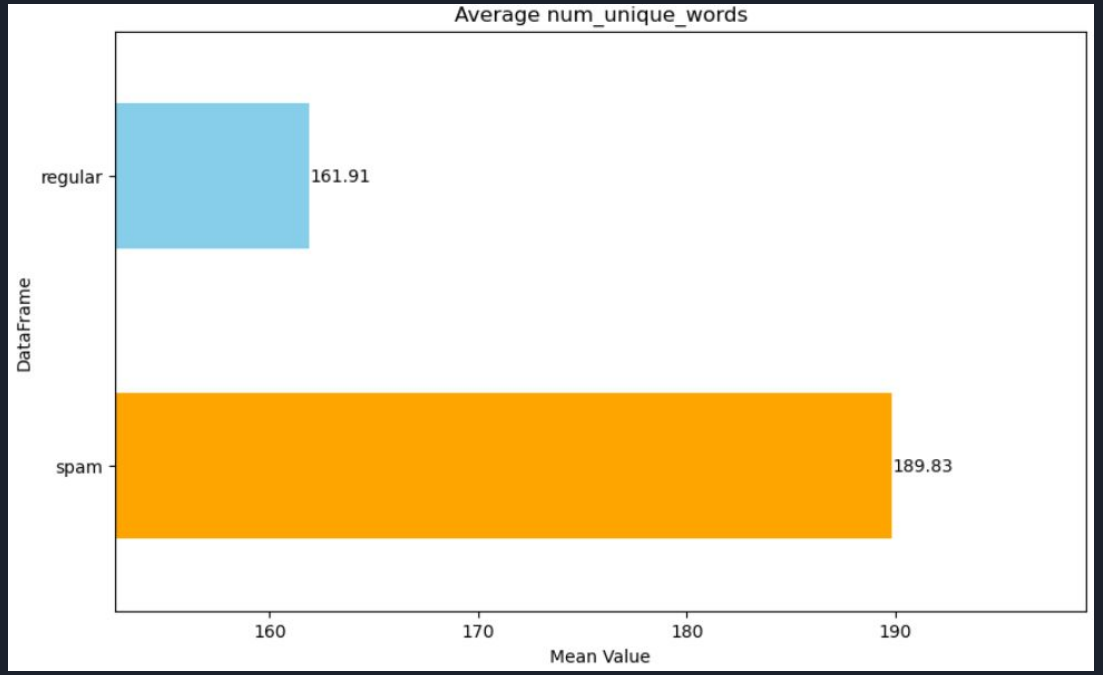
# Number of Unique Words

This feature contains the number of column values (words) that are 1 as opposed to 0, but does not care if a word is used more than once.

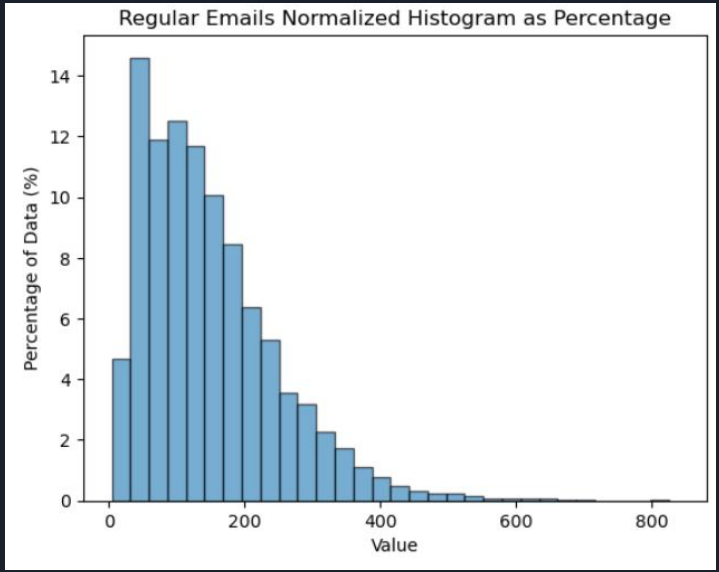
Spam emails contain 17.25% higher number of unique words than regular emails.

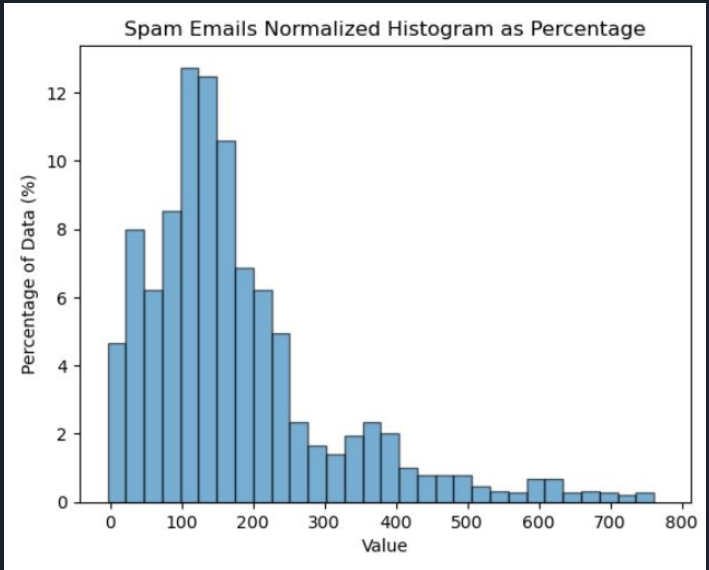
| REG EMAILS | |
|---|---|
| count | 3672.00 |
| mean | 161.90 |
| std | 100.07 |
| min | 18.00 |
| 25% | 87.00 |
| 50% | 142.00 |
| 75% | 214.00 |
| max | 839.00 |

| SPAM EMAILS | |
|---|---|
| count | 1500.00 |
| mean | 189.83 |
| std | 136.47 |
| min | 9.00 |
| 25% | 103.00 |
| 50% | 155.00 |
| 75% | 234.00 |
| max | 774.00 |



# Number of Unique Words



# Biserial Correlations

| Feature | Correlation | p-value |
|---|---|---|
| thanks | -0.271433 | 4.926148e-88 |
| hpl | -0.266518 | 7.952302e-85 |
| hanks | -0.266070 | 1.547126e-84 |
| thank | -0.262384 | 3.521933e-82 |
| attached | -0.236558 | 1.048551e-66 |
| daren | -0.236180 | 1.711801e-66 |
| forwarded | -0.230765 | 1.761133e-63 |
| subject | -0.227754 | 7.714997e-62 |
| hp | -0.225846 | 8.229589e-61 |
| aren | -0.206063 | 1.041754e-50 |
| nom | -0.202600 | 4.820134e-49 |
| farmer | -0.194693 | 2.353483e-45 |
| questions | -0.193163 | 1.167173e-44 |
| deal | -0.190407 | 2.021873e-43 |
| than | -0.188514 | 1.397021e-42 |
| volume | -0.188005 | 2.341012e-42 |
| enron | -0.186740 | 8.388549e-42 |
| question | -0.185967 | 1.822100e-41 |
| xls | -0.179113 | 1.524257e-38 |
| meter | -0.166499 | 1.817299e-33 |
| please | -0.162304 | 7.274247e-32 |
| btu | -0.162183 | 8.081026e-32 |
| pm | -0.161234 | 1.835590e-31 |
| mmbtu | -0.157753 | 3.565135e-30 |
| gas | -0.156652 | 8.987784e-30 |

| Feature | Correlation | p-value |
|---|---|---|
| more | 0.258152 | 1.607936e-79 |
| our | 0.228187 | 4.495713e-62 |
| able | 0.222219 | 6.968407e-59 |
| best | 0.221703 | 1.301847e-58 |
| ur | 0.220253 | 7.483086e-58 |
| sex | 0.220092 | 9.079039e-58 |
| sec | 0.217402 | 2.241652e-56 |
| money | 0.217215 | 2.799293e-56 |
| soft | 0.213382 | 2.498362e-54 |
| dr | 0.212413 | 7.671000e-54 |
| mo | 0.210056 | 1.146765e-52 |
| via | 0.204031 | 9.968704e-50 |
| prescription | 0.203896 | 1.157285e-49 |
| remove | 0.203384 | 2.036279e-49 |
| cheap | 0.200348 | 5.617777e-48 |
| meds | 0.198501 | 4.119483e-47 |
| drug | 0.197976 | 7.231285e-47 |
| of | 0.197234 | 1.598227e-46 |
| ali | 0.194936 | 1.821868e-45 |
| ic | 0.194706 | 2.321878e-45 |
| cia | 0.191742 | 5.105069e-44 |
| offer | 0.190368 | 2.103627e-43 |
| off | 0.189532 | 4.953437e-43 |
| your | 0.186149 | 1.518953e-41 |
| prices | 0.186026 | 1.717367e-41 |

# Logistic Regression Coefficients

| Feature | Coefficient |
|---|---|
| please | -0.096544 |
| thank | -0.097962 |
| hou | -0.102684 |
| xl | -0.104496 |
| xls | -0.108344 |
| for | -0.109866 |
| deal | -0.110454 |
| if | -0.121969 |
| on | -0.128271 |
| gas | -0.128509 |
| ct | -0.129961 |
| the | -0.130633 |
| to | -0.132465 |
| nom | -0.135767 |
| day | -0.138160 |
| ls | -0.141297 |
| wil | -0.163334 |
| as | -0.168625 |
| will | -0.171207 |
| hp | -0.207917 |
| ect | -0.219973 |
| j | -0.234368 |
| hpl | -0.235955 |
| da | -0.289221 |
| enron | -0.316404 |

| Feature | Coefficient |
|---|---|
| mo | 0.277554 |
| of | 0.272639 |
| re | 0.260195 |
| b | 0.206786 |
| nd | 0.184295 |
| al | 0.181908 |
| z | 0.172111 |
| s | 0.169402 |
| t | 0.168527 |
| o | 0.158004 |
| i | 0.156816 |
| v | 0.152320 |
| ve | 0.151052 |
| pa | 0.147167 |
| u | 0.145550 |
| le | 0.141780 |
| f | 0.141519 |
| one | 0.140153 |
| d | 0.136314 |
| men | 0.129771 |
| ur | 0.129573 |
| p | 0.127920 |
| l | 0.126672 |
| w | 0.123955 |
| n | 0.122334 |

Neg: Least Correlated with ‘Spam’

Pos: Most Correlated with ‘Spam’

# *If the model…*

1) *Never Misses a Spam Email*,

   Real emails will be marked as spam because of more spam predictions

2) *Never Classifies a Regular Email as Spam*,

   **S**pam emails will be allowed into inbox because of fewer spam predictions

3) *Overall Highest Accuracy*

   **Cost:** Undependable, inconsistent

**Goal:** Minimize the Costs by choosing proper training methodology (model, split, scale)

# Generalized Training: Spam Classification

The words (that indicate spam) in this dataset will be different from other datasets. By comparing model performances, we can answer the questions:

Which training methodology will be best suited to train a…

- ‘Strict’ Spam filter: Never Misses a Spam Email
- ‘Relaxed’ Spam filter: Never Misclassifies a Real Email as Spam
- ‘Balanced’ Spam filter: Least Number of Errors - period

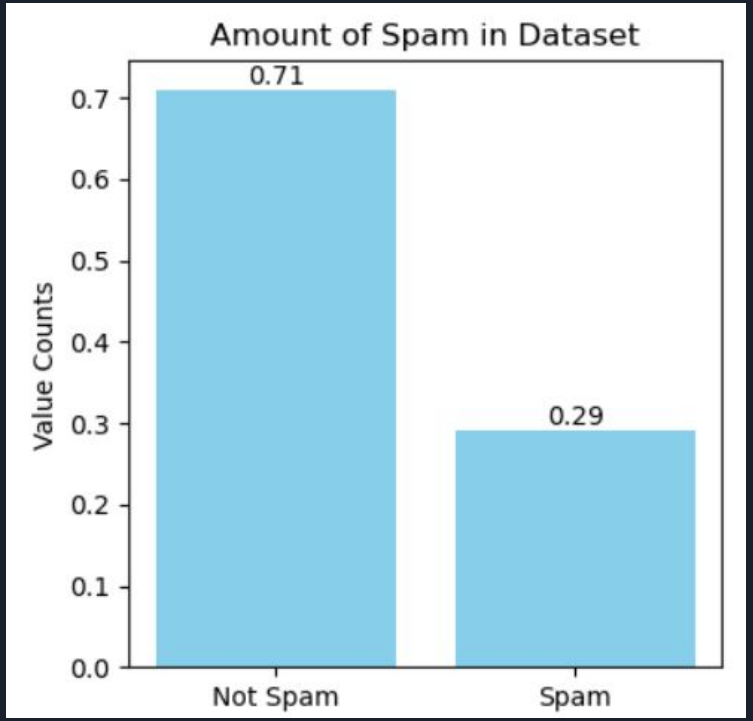
# Challenge: Class Imbalance

**Solutions**:

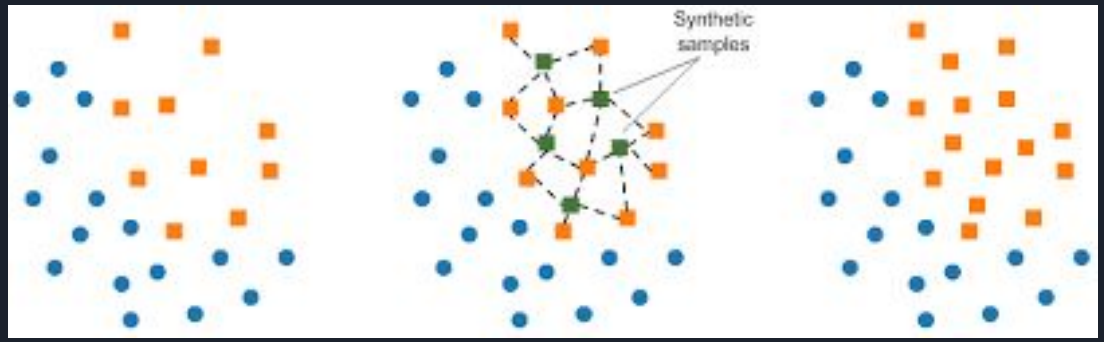
1) SMOTE
2) Stratification
3) Class Weights

3,672 Real emails (71%), 1,500 Spam emails (29%)

Studies show 46% of Emails are Spam and 54% are real emails, in reality.



# SMOTE




**Creates Synthetic data to create a balanced 50/50 class representation**

**This dataset will become 50% spam vs. 50% real from applying SMOTE**

# Class Weights: Conservative Use

To mimic a 54/46 class split, inverse proportionality results in a class weight of 200% for spam. However, I found the more optimal class weight to be much lower at 125% to minimize False Positives.

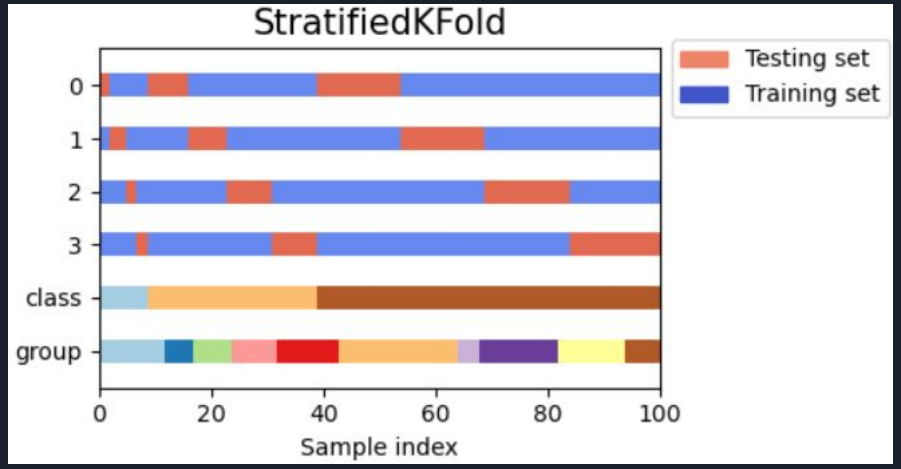
class_weight = {0:1, 1:1.28} ‘mimics’ a 63% / 37% Split of data

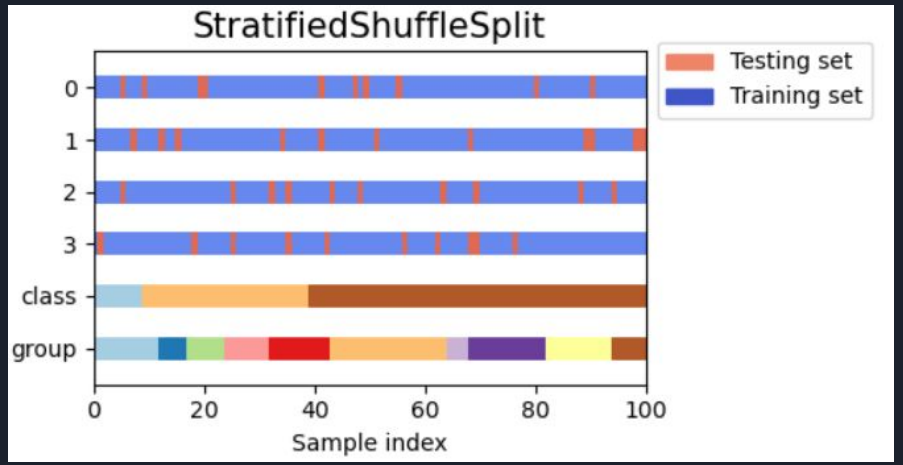
# Split Methods

**Class**: Spam
**Groups**: Word Count Features

SKF: Guarantees equal representations of spam/real in training set, and test set (71/29 split)

SSS: Guarantees equal representations of word features and the spam/real classification



# Model Metrics

*Precision:* # of Correct Spam Predictions /  All prediction made

- **Precision 0.9:** For every 10 emails are in spam box, 9 will be spam, 1 will be real

*Recall:* # of Correct Spam Predictions /  All spam emails

- **Recall 0.8:** For every 10 spam emails, 9 will be in spam box, 1 will not in the inbox

*f1-score:* *harmonic mean of both*

# Modeling

**SMOTE vs. ‘Un-SMOTE’**
**Class Weights vs. Unweighted**
**Scalars:** MinMax, MaxAbs, Standard
**Split Methods:** TrainTestSplit, StratifiedKFold, StratifiedShuffleSplit
**Models:** LogisticRegression, RandomForest, CatBoost, LightGBM (‘LGBM’)

**Findings:**

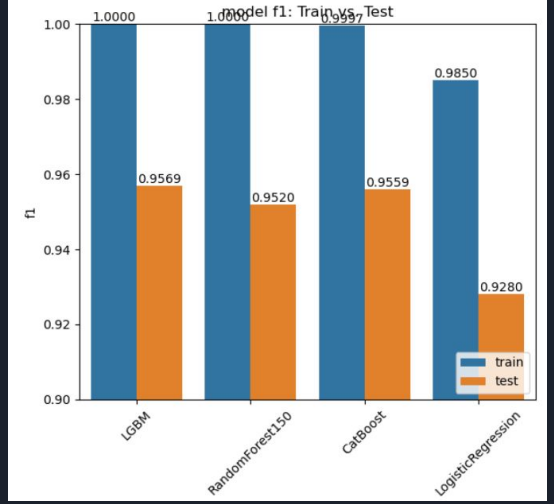
- SMOTE increases recall decreases precision
- LGBM agnostic towards SMOTE and Class Weights
- LR agnostic towards class weights
- CatBoost agnostic towards Scaling Method

# f1 by Model

SMOTE decreases test set performance. RandomForest benefits from class weights.  CatBoost/LGBM unchanged.

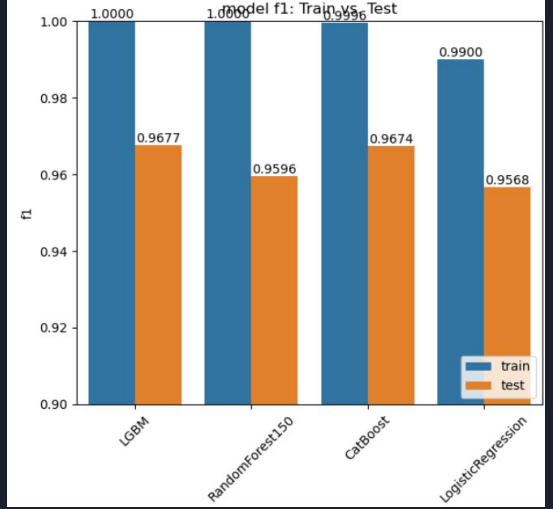
**1) SMOTE**

```
plot_metric(model_smote_train.reset_index(), model_smote.reset_index(), 'mode
```



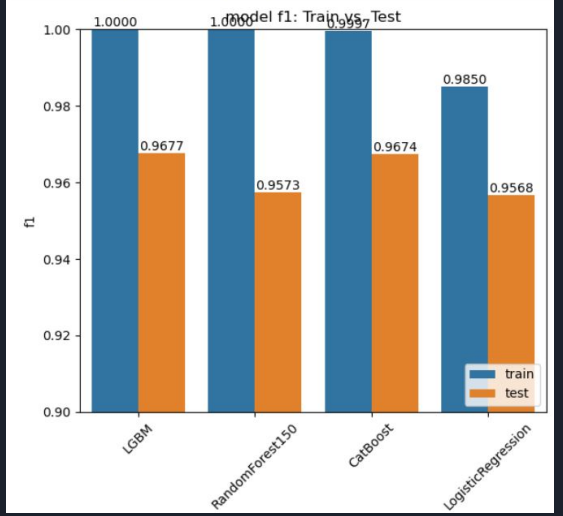

**2) 1:1.25 Class weights**

```
plot_metric(model_w_train.reset_index(), model_w.reset_index(), 'model', met
```




**3) Unweighted**

```
plot_metric(model_train.reset_index(), model_reg.reset_index(), 'model', metr
```



# Overfitting: f1

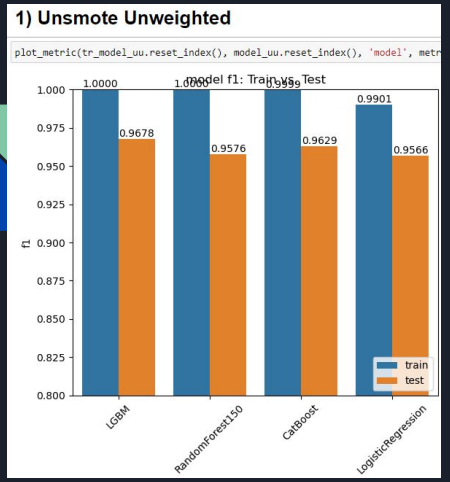
Smote decreases Logistic Regression, CatBoost

Class weights benefit CatBoost, RandomForest

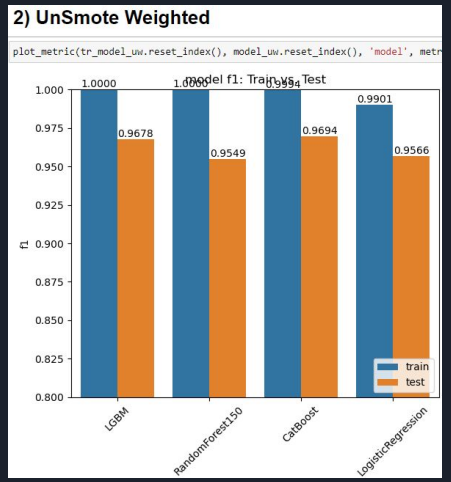
SMOTE Added

Weights Added

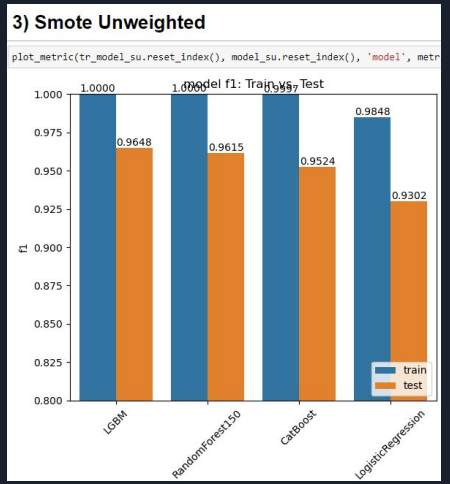
**1) Unsmote Unweighted**



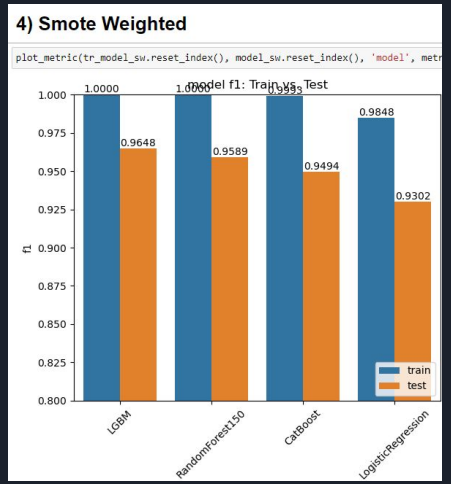

**2) UnSmote Weighted**




**3) Smote Unweighted**




**4) Smote Weighted**



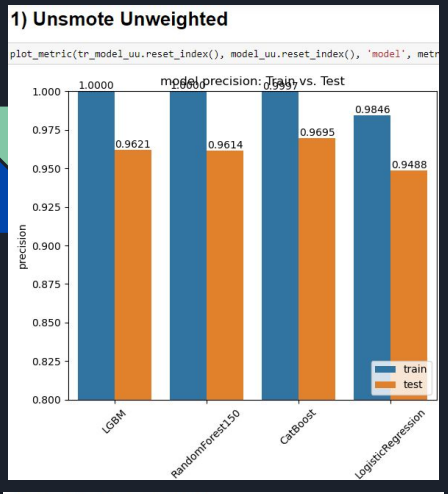
# Overfitting: precision

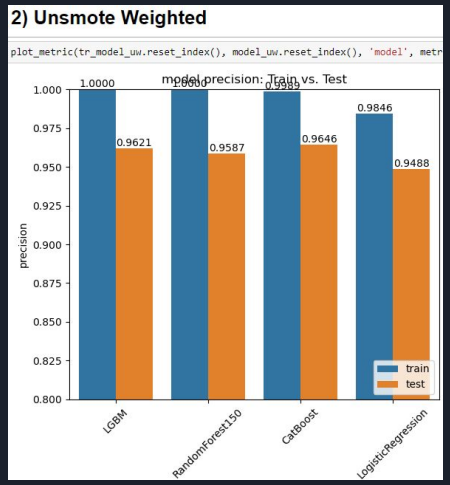
Smote decreases precision dramatically,
Especially in Logistic Regression, CatBoost

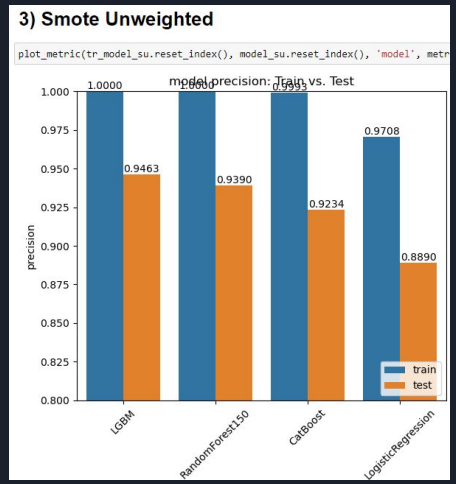
Class weights decrease precision slightly,
Except in Logistic Regression
Not in LGBM

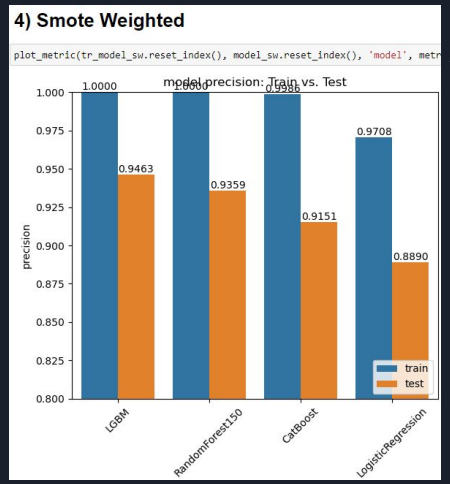
SMOTE Added

Weights Added



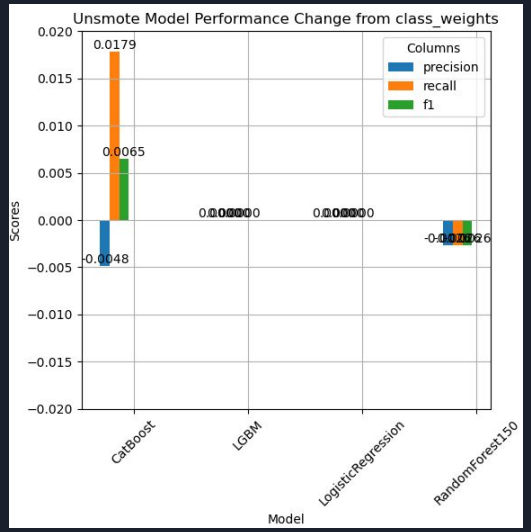



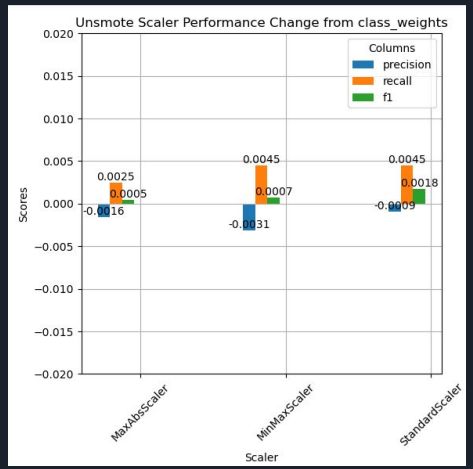



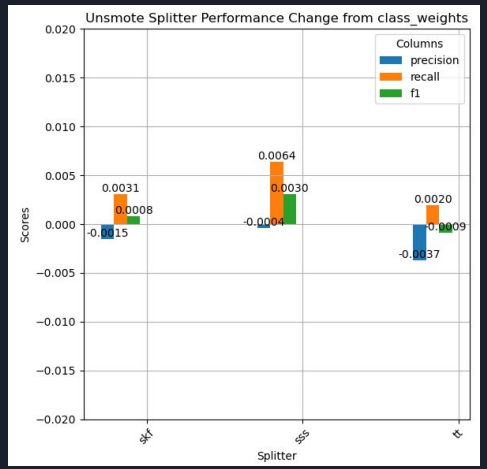



# Un-Smote Models: Adding Class Weights

CatBoost, SSS, StandardScaler: Increased f1

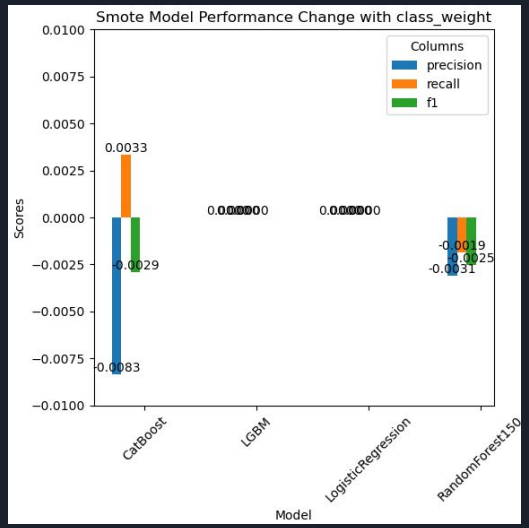


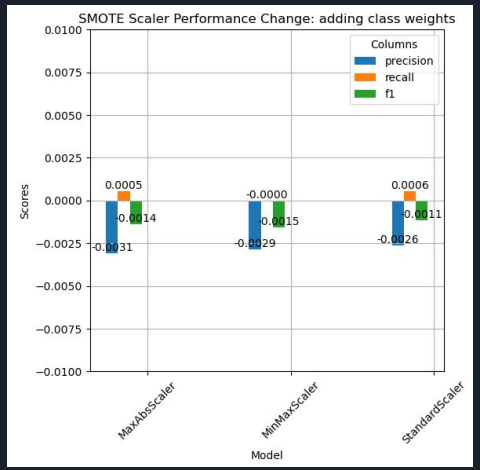



# Smote Models: Adding Class Weights

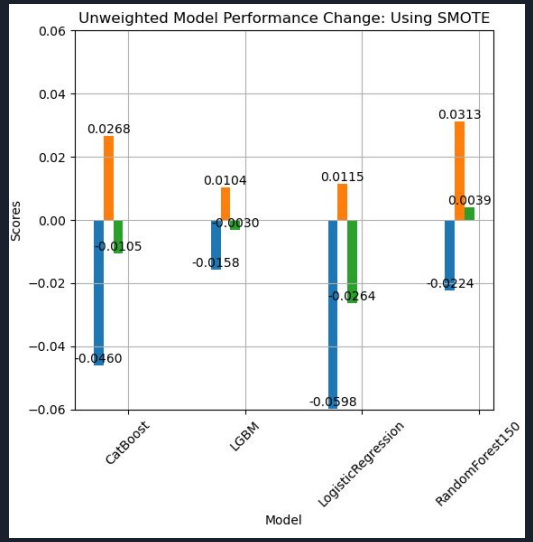
Decreased performance

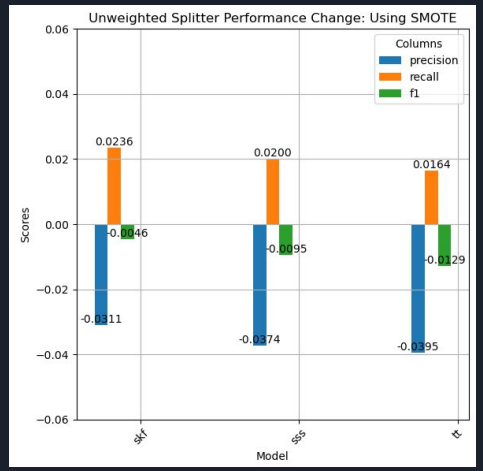


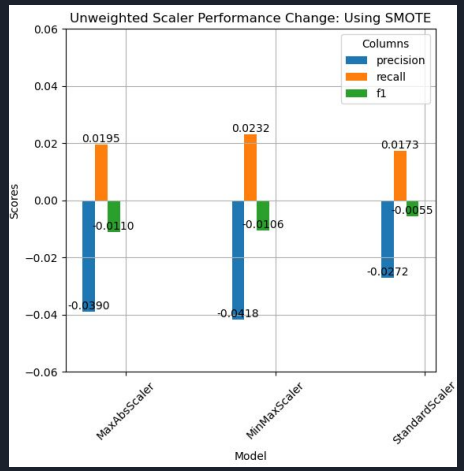




# Unweighted Models: Adding SMOTE



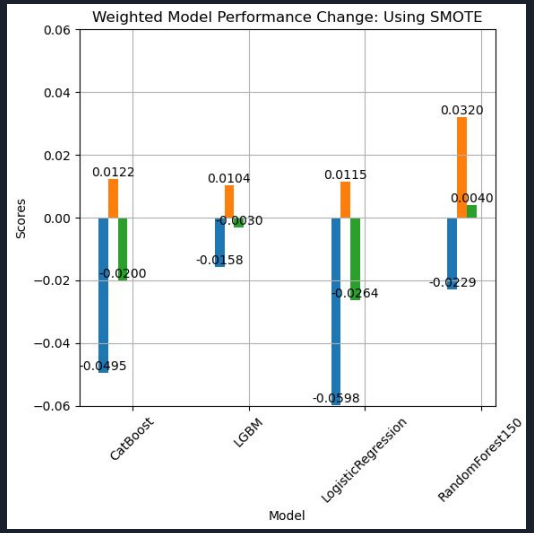



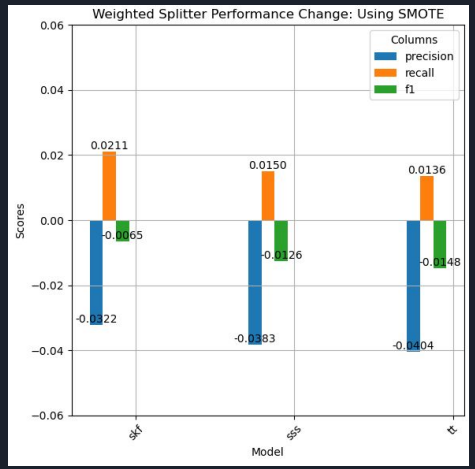

RandomForest benefits barely,

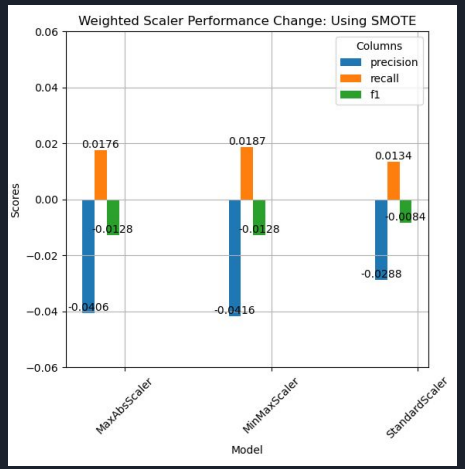
LogisticRegression falters

# Weighted Models: Adding SMOTE

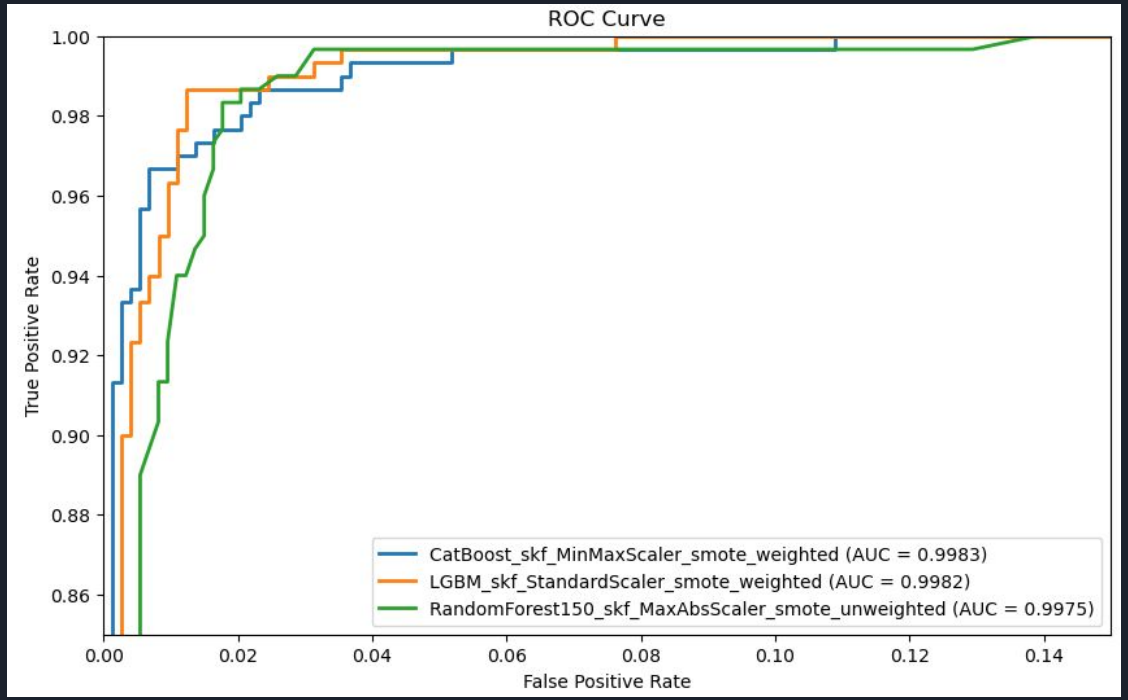
RandomForest benefits barely



# ROC AUC



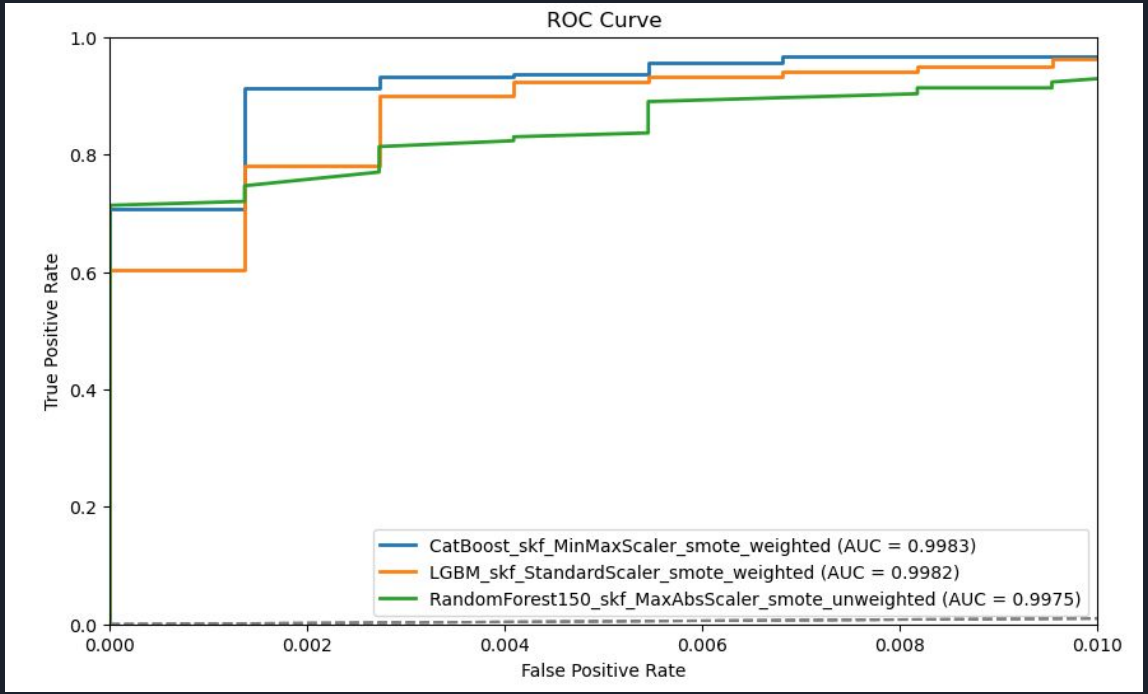

Top of the graph:

Farthest left curve classifies the least number of regular emails as spam, while ensuring every spam email is classified as spam

LGBM, SKF,

Any: Scaler, Smote, Weights

# ROC AUC




Left of the graph:

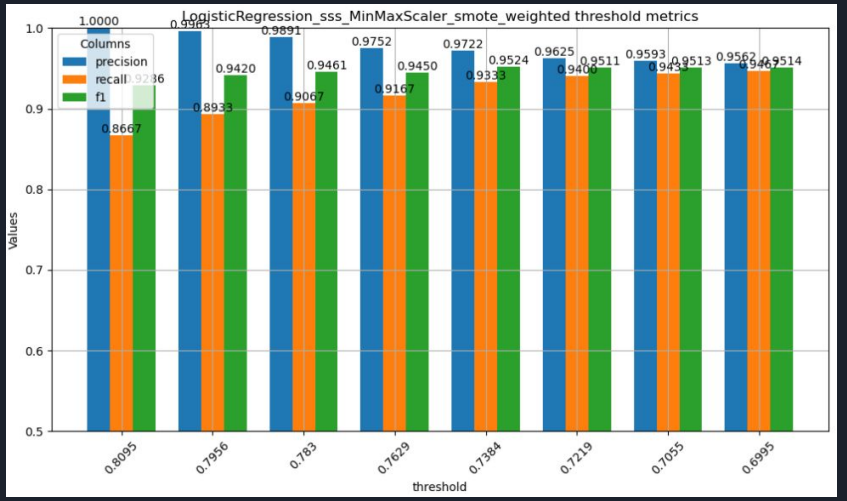
The highest curve classifies the highest number of spam (recall) at the moment it makes its first mistake in classifying the first regular email as spam

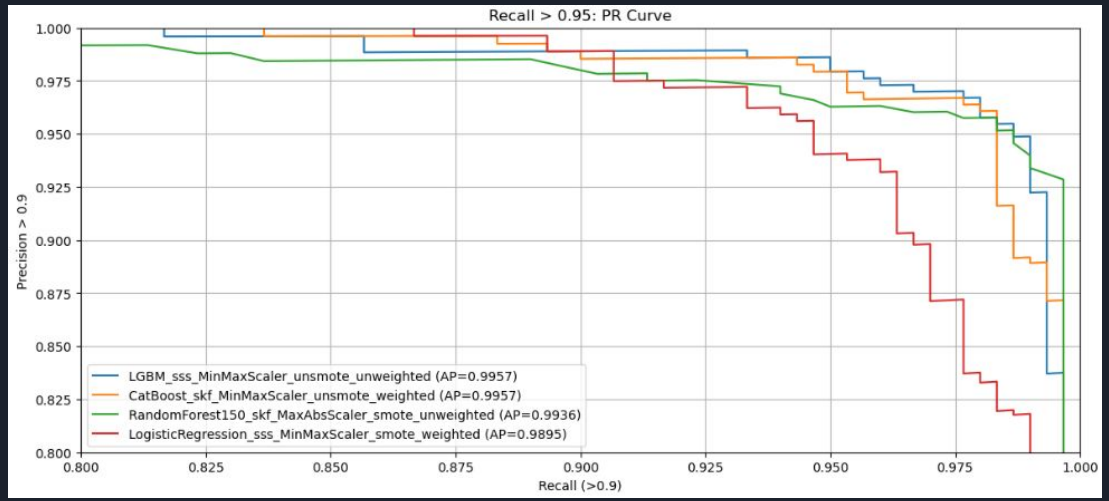
CatBoost, SKF,

Any Scaler, Smote, Weights

# Best ‘Relaxed’ Spam Filter

LogisticRegression, SSS, MinMax/MaxAbs, Smote
74% thresh: 13.33% of spam allowed
F1: 0.9286

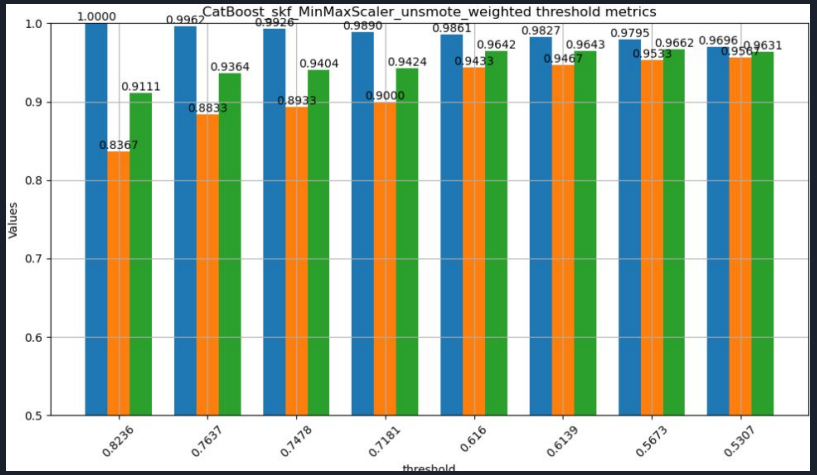


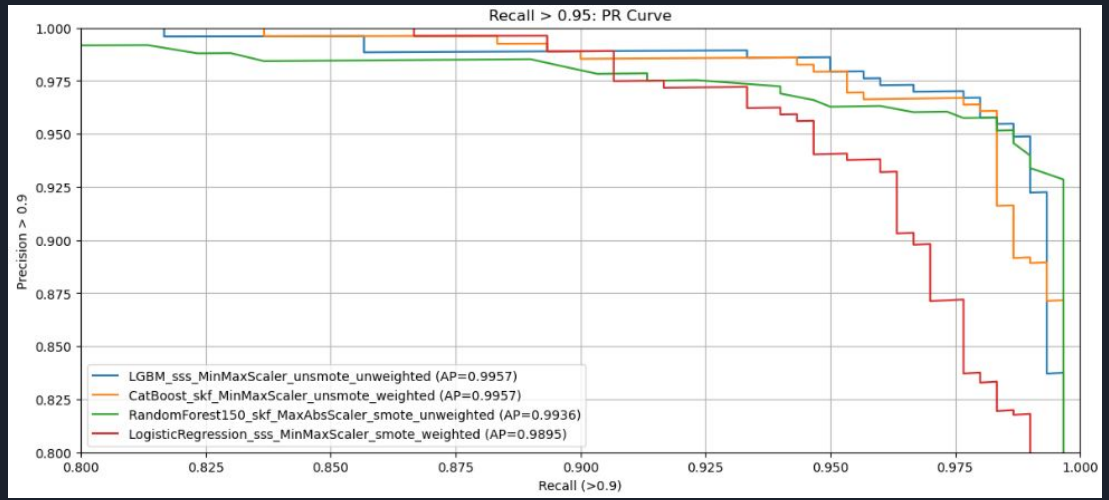




| model | recall_at_precision_1 | optimal_threshold | max_f1_score |
|---|---|---|---|
| LogisticRegression_sss_MinMaxScaler_smote_weighted | 0.866667 | 0.740441 | 0.952381 |
| CatBoost_skf_MinMaxScaler_unsmote_weighted | 0.836667 | 0.429214 | 0.971993 |
| CatBoost_skf_MinMaxScaler_unsmote_unweighted | 0.826667 | 0.400109 | 0.971901 |
| LGBM_sss_MinMaxScaler_unsmote_weighted | 0.816667 | 0.530220 | 0.973510 |
| LogisticRegression_sss_MinMaxScaler_unsmote_weighted | 0.786667 | 0.547190 | 0.964587 |
| LGBM_sss_StandardScaler_unsmote_weighted | 0.780000 | 0.443676 | 0.970199 |
| LGBM_skf_MinMaxScaler_unsmote_weighted | 0.763333 | 0.495812 | 0.968491 |
| RandomForest150_skf_StandardScaler_smote_unweighted | 0.760000 | 0.486667 | 0.961165 |

# Most Reliable ‘Relaxed’ Spam Filter

CatBoost, SKF, Un-SMOTE, Weighted
82% thresh: 16.33% of spam allowed
F1: 0.9111



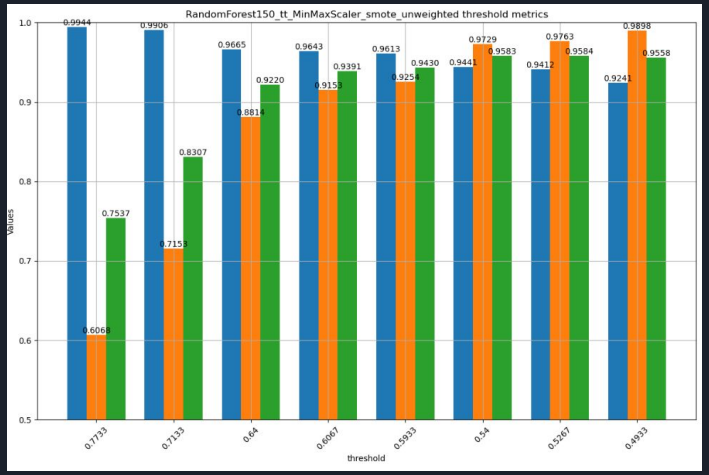



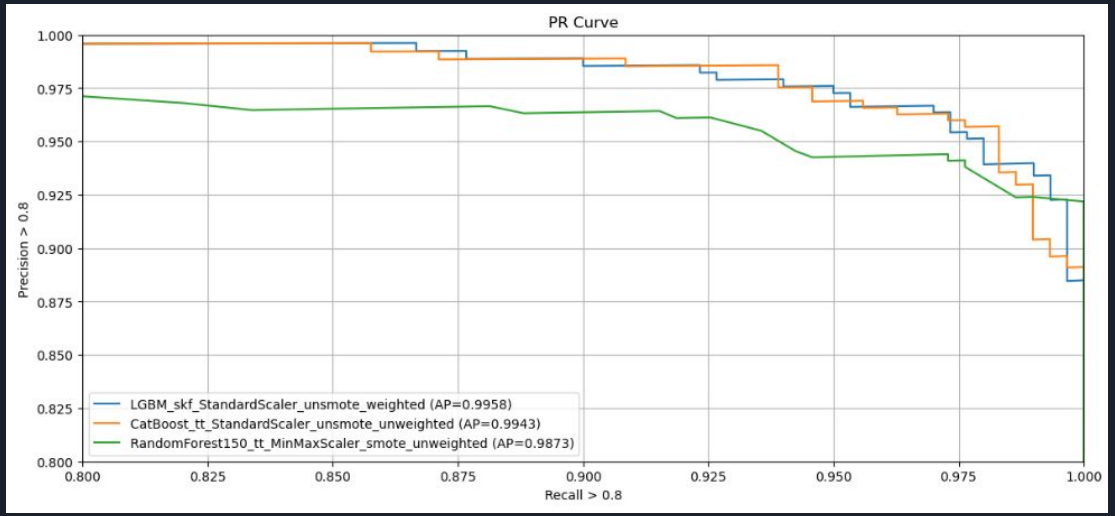

| model | recall_at_precision_1 | optimal_threshold | max_f1_score |
|---|---|---|---|
| LogisticRegression_sss_MinMaxScaler_smote_weighted | 0.866667 | 0.740441 | 0.952381 |
| CatBoost_skf_MinMaxScaler_unsmote_weighted | 0.836667 | 0.429214 | 0.971993 |
| CatBoost_skf_MinMaxScaler_unsmote_unweighted | 0.826667 | 0.400109 | 0.971901 |
| LGBM_sss_MinMaxScaler_unsmote_weighted | 0.816667 | 0.530220 | 0.973510 |
| LogisticRegression_sss_MinMaxScaler_unsmote_weighted | 0.786667 | 0.547190 | 0.964587 |
| LGBM_sss_StandardScaler_unsmote_weighted | 0.780000 | 0.443676 | 0.970199 |
| LGBM_skf_MinMaxScaler_unsmote_weighted | 0.763333 | 0.495812 | 0.968491 |
| RandomForest150_skf_StandardScaler_smote_unweighted | 0.760000 | 0.486667 | 0.961165 |

# Best ‘Strict’ Spam Filter

RandomForest, MinMax, TTest, SMOTE, unweighted

49% threshold
7.82% of real emails caught as spam
F1: 0.9593



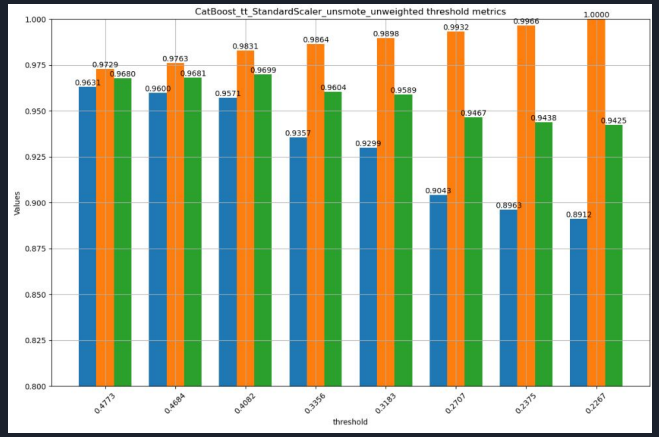



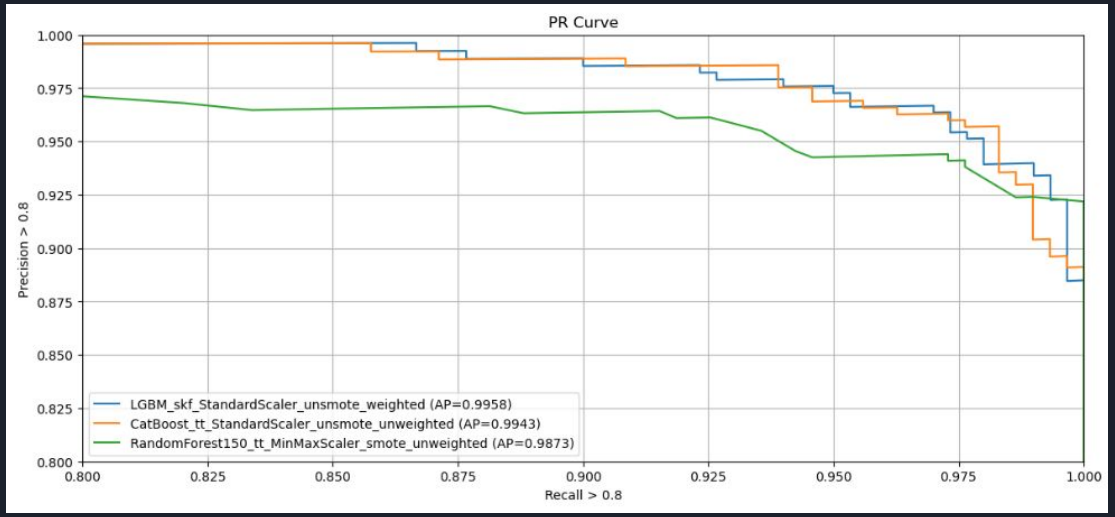

| model | precision_at_recall_1 | recall_at_precision_1 | optimal_threshold |
|---|---|---|---|
| RandomForest150_tt_MinMaxScaler_smote_unweighted | 0.921875 | 0.477966 | 0.493333 |
| RandomForest150_tt_MinMaxScaler_smote_weighted | 0.913313 | 0.633898 | 0.546667 |
| RandomForest150_tt_MinMaxScaler_unsmote_unweighted | 0.907692 | 0.359322 | 0.486667 |
| RandomForest150_tt_MaxAbsScaler_unsmote_weighted | 0.902141 | 0.416949 | 0.480000 |
| RandomForest150_tt_MaxAbsScaler_smote_unweighted | 0.899390 | 0.427119 | 0.513333 |
| CatBoost_tt_StandardScaler_unsmote_unweighted | 0.891239 | 0.508475 | 0.434900 |
| RandomForest150_tt_StandardScaler_unsmote_unweighted | 0.891239 | 0.457627 | 0.493333 |
| CatBoost_tt_MinMaxScaler_unsmote_unweighted | 0.891239 | 0.508475 | 0.434900 |
| CatBoost_tt_MaxAbsScaler_unsmote_unweighted | 0.891239 | 0.508475 | 0.434900 |
| CatBoost_tt_StandardScaler_smote_weighted | 0.888554 | 0.400000 | 0.709200 |
| CatBoost_tt_MinMaxScaler_smote_weighted | 0.888554 | 0.400000 | 0.709200 |
| CatBoost_tt_MaxAbsScaler_smote_weighted | 0.888554 | 0.400000 | 0.709200 |
| LGBM_skf_MinMaxScaler_unsmote_weighted | 0.884956 | 0.763333 | 0.495812 |
| LGBM_skf_StandardScaler_unsmote_weighted | 0.884956 | 0.763333 | 0.495812 |
| LGBM_skf_MaxAbsScaler_unsmote_weighted | 0.884956 | 0.763333 | 0.495812 |

# Most Reliable ‘Strict’ Spam Filter

CatBoost, MinMax, TTest, UnSMOTE, unweighted

23% threshold
11.88% of real emails caught as spam
F1: 0.9425



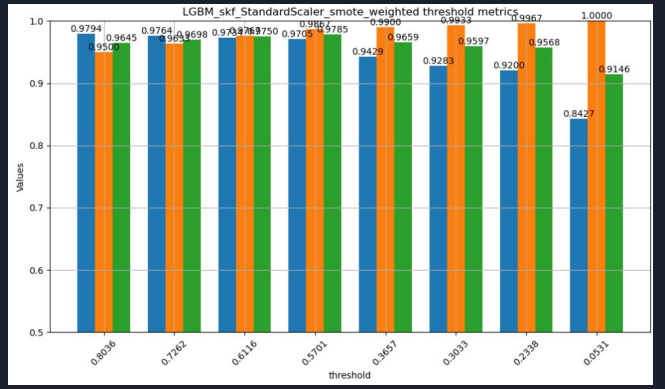




| model | precision_at_recall_1 | recall_at_precision_1 | optimal_threshold |
|---|---|---|---|
| RandomForest150_tt_MinMaxScaler_smote_unweighted | 0.921875 | 0.477966 | 0.493333 |
| RandomForest150_tt_MinMaxScaler_smote_weighted | 0.913313 | 0.633898 | 0.546667 |
| RandomForest150_tt_MinMaxScaler_unsmote_unweighted | 0.907692 | 0.359322 | 0.486667 |
| RandomForest150_tt_MaxAbsScaler_unsmote_weighted | 0.902141 | 0.416949 | 0.480000 |
| RandomForest150_tt_MaxAbsScaler_smote_unweighted | 0.899390 | 0.427119 | 0.513333 |
| CatBoost_tt_StandardScaler_unsmote_unweighted | 0.891239 | 0.508475 | 0.434900 |
| RandomForest150_tt_StandardScaler_unsmote_unweighted | 0.891239 | 0.457627 | 0.493333 |
| CatBoost_tt_MinMaxScaler_unsmote_unweighted | 0.891239 | 0.508475 | 0.434900 |
| CatBoost_tt_MaxAbsScaler_unsmote_unweighted | 0.891239 | 0.508475 | 0.434900 |
| CatBoost_tt_StandardScaler_smote_weighted | 0.888554 | 0.400000 | 0.709200 |
| CatBoost_tt_MinMaxScaler_smote_weighted | 0.888554 | 0.400000 | 0.709200 |
| CatBoost_tt_MaxAbsScaler_smote_weighted | 0.888554 | 0.400000 | 0.709200 |
| LGBM_skf_MinMaxScaler_unsmote_weighted | 0.884956 | 0.763333 | 0.495812 |
| LGBM_skf_StandardScaler_unsmote_weighted | 0.884956 | 0.763333 | 0.495812 |
| LGBM_skf_MaxAbsScaler_unsmote_weighted | 0.884956 | 0.763333 | 0.495812 |

# ‘Balanced Accuracy’ Spam Filter: Recall

LGBM, SKF, StandardScaler, SMOTE, Weighted

57% threshold
f1: 0.9785
2.95% of real emails in spam box,
1.33% of spam emails in inbox

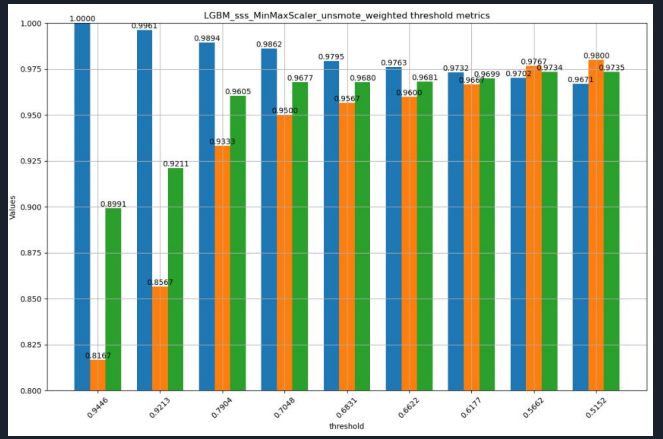



| model | recall_at_precision_1 | precision_at_recall_1 | max_f1_score | optimal_threshold |
|---|---|---|---|---|
| LGBM_skf_StandardScaler_smote_weighted | 0.603333 | 0.842697 | 0.978512 | 0.573696 |
| CatBoost_sss_MinMaxScaler_unsmote_weighted | 0.706667 | 0.777202 | 0.976589 | 0.556067 |
| LGBM_skf_MinMaxScaler_smote_weighted | 0.570000 | 0.808625 | 0.975042 | 0.655214 |
| CatBoost_skf_MinMaxScaler_smote_weighted | 0.706667 | 0.789474 | 0.974790 | 0.718492 |
| LGBM_sss_MinMaxScaler_unsmote_weighted | 0.816667 | 0.699301 | 0.973510 | 0.530220 |
| CatBoost_tt_MinMaxScaler_unsmote_weighted | 0.488136 | 0.870206 | 0.973064 | 0.540427 |
| CatBoost_skf_MinMaxScaler_unsmote_weighted | 0.836667 | 0.773196 | 0.971993 | 0.429214 |
| CatBoost_skf_MinMaxScaler_unsmote_unweighted | 0.826667 | 0.750000 | 0.971901 | 0.400109 |
| CatBoost_sss_MinMaxScaler_smote_unweighted | 0.710000 | 0.717703 | 0.971714 | 0.687580 |
| CatBoost_sss_MinMaxScaler_smote_weighted | 0.680000 | 0.733496 | 0.971524 | 0.759538 |
| RandomForest150_sss_StandardScaler_smote_unweighted | 0.233333 | 0.777202 | 0.970395 | 0.526667 |
| RandomForest150_skf_MaxAbsScaler_smote_unweighted | 0.713333 | 0.746269 | 0.970395 | 0.540000 |
| LGBM_sss_StandardScaler_unsmote_weighted | 0.780000 | 0.652174 | 0.970199 | 0.443676 |
| LGBM_tt_MinMaxScaler_unsmote_weighted | 0.698305 | 0.880597 | 0.970000 | 0.381633 |
| RandomForest150_tt_MinMaxScaler_unsmote_unweighted | 0.359322 | 0.907692 | 0.969900 | 0.486667 |

# ‘Balanced Accuracy’ Spam Filter: Precision

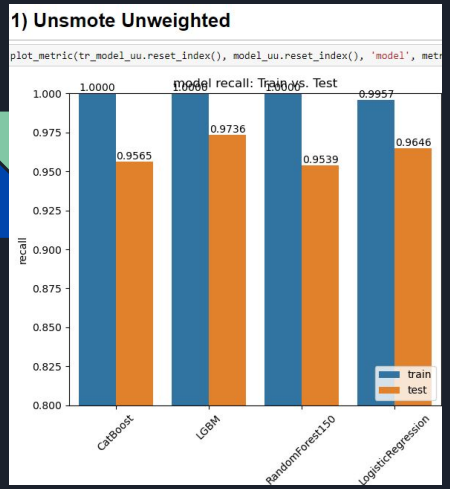
LGBM, SSS, MinMax/MaxAbs, unSMOTE

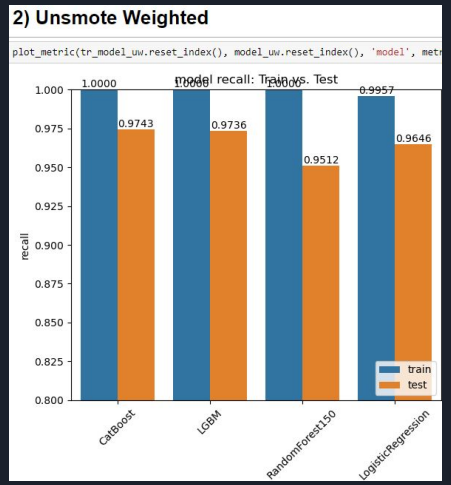
52% Threshold: 0.9735 f1-score
3.39% of real emails in spam box, 2% of spam emails in inbox



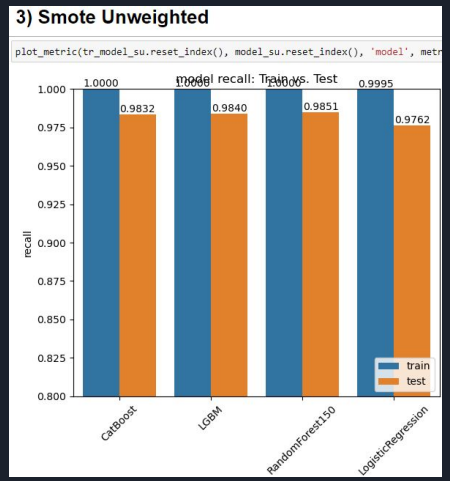

| model | recall_at_precision_1 | precision_at_recall_1 | max_f1_score | optimal_threshold |
|---|---|---|---|---|
| LGBM_skf_StandardScaler_smote_weighted | 0.603333 | 0.842697 | 0.978512 | 0.573696 |
| CatBoost_sss_MinMaxScaler_unsmote_weighted | 0.706667 | 0.777202 | 0.976589 | 0.556067 |
| LGBM_skf_MinMaxScaler_smote_weighted | 0.570000 | 0.808625 | 0.975042 | 0.655214 |
| CatBoost_skf_MinMaxScaler_smote_weighted | 0.706667 | 0.789474 | 0.974790 | 0.718492 |
| LGBM_sss_MinMaxScaler_unsmote_weighted | 0.816667 | 0.699301 | 0.973510 | 0.530220 |
| CatBoost_tt_MinMaxScaler_unsmote_weighted | 0.488136 | 0.870206 | 0.973064 | 0.540427 |
| CatBoost_skf_MinMaxScaler_unsmote_weighted | 0.836667 | 0.773196 | 0.971993 | 0.429214 |
| CatBoost_skf_MinMaxScaler_unsmote_unweighted | 0.826667 | 0.750000 | 0.971901 | 0.400109 |
| CatBoost_sss_MinMaxScaler_smote_unweighted | 0.710000 | 0.717703 | 0.971714 | 0.687580 |
| CatBoost_sss_MinMaxScaler_smote_weighted | 0.680000 | 0.733496 | 0.971524 | 0.759538 |
| RandomForest150_sss_StandardScaler_smote_unweighted | 0.233333 | 0.777202 | 0.970395 | 0.526667 |
| RandomForest150_skf_MaxAbsScaler_smote_unweighted | 0.713333 | 0.746269 | 0.970395 | 0.540000 |
| LGBM_sss_StandardScaler_unsmote_weighted | 0.780000 | 0.652174 | 0.970199 | 0.443676 |
| LGBM_tt_MinMaxScaler_unsmote_weighted | 0.698305 | 0.880597 | 0.970000 | 0.381633 |
| RandomForest150_tt_MinMaxScaler_unsmote_unweighted | 0.359322 | 0.907692 | 0.969900 | 0.486667 |

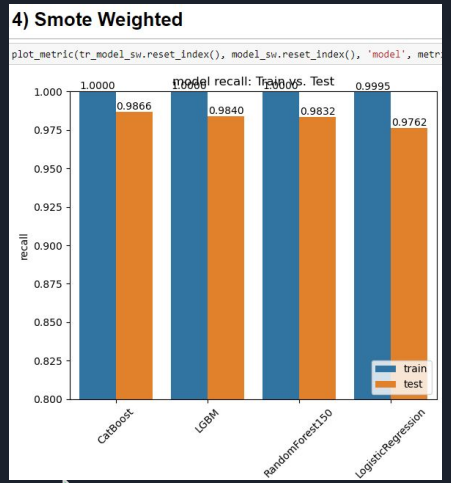
# Recall by Model




SMOTE Added




Weights Added

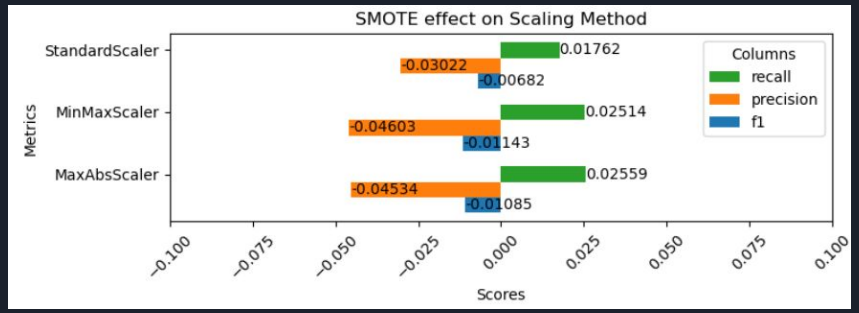
Smote increases recall dramatically

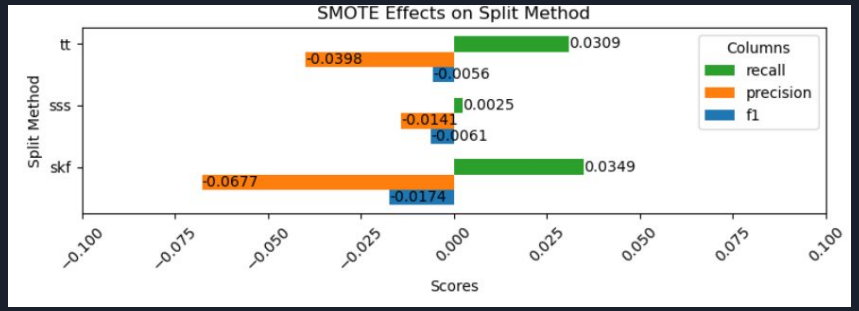
Random Forest
Unweighted > Weighted

For CatBoost and Logistic Regression

Class weights increase recall slightly,
Except in Logistic Regression
Not in LGBM

# Ruling out SMOTE



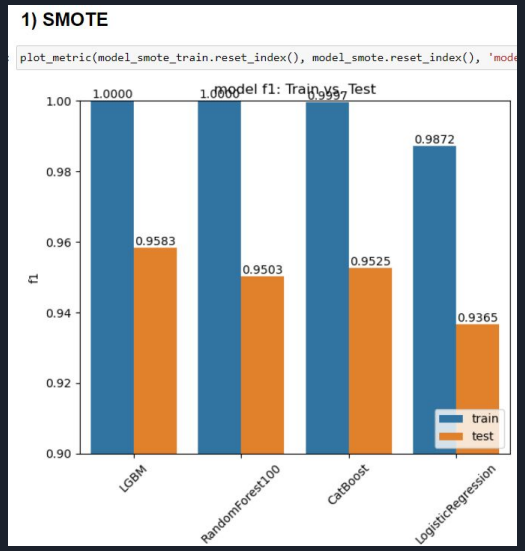

SMOTE works best with StandardScaler.

The effects of decreasing precision/increasing recall is milder, f1 performance is decreased.

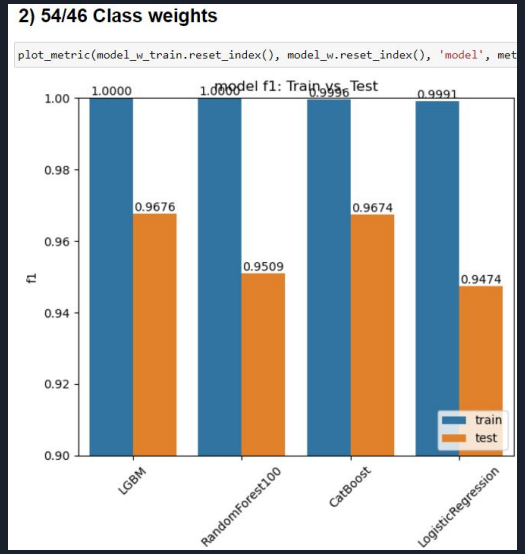
SSS shows smallest changes from using SMOTE, but test_train_split overall best performance (f1).

# F1: Smote vs. Class Weights vs. Unchanged

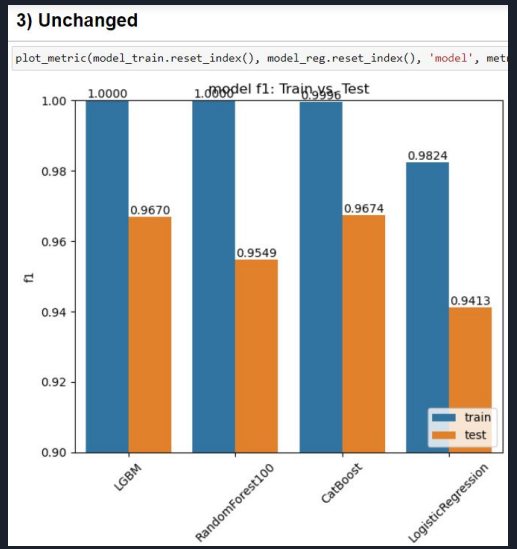
**1) SMOTE**




**2) 54/46 Class weights**




**3) Unchanged**



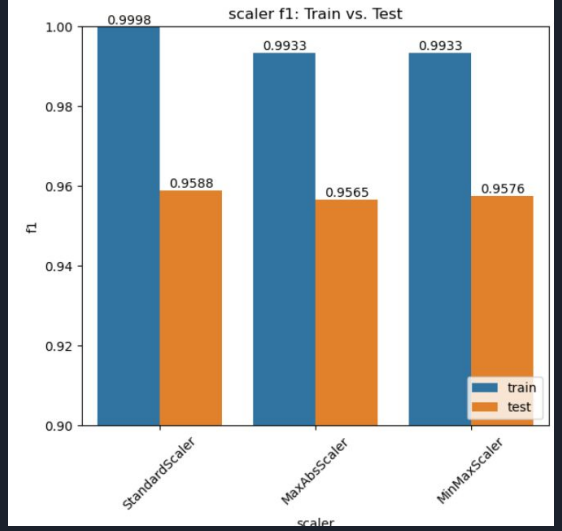

LGBM and LogisticRegression benefit the most from class weights.

SMOTE decreases for all models.

# Scalar: f1 Overfitting

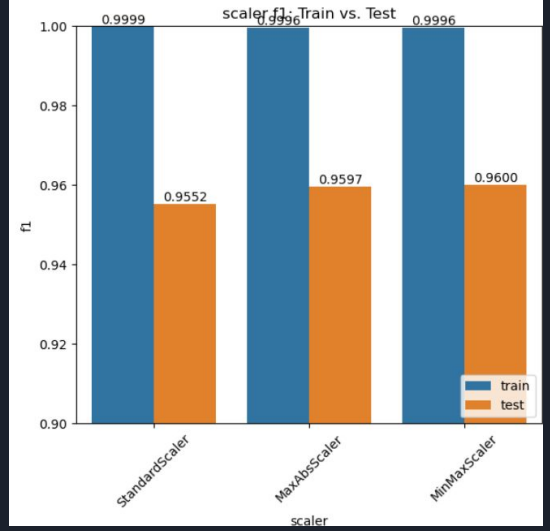
**1) Unchanged**

```
plot_metric(scalar_train.reset_index(), scalar_reg.reset_index(), 'scaler',
```




**2) Class Weights**

```
plot_metric(scalar_w_train.reset_index(), scalar_w.reset_index(), 'scaler',
```

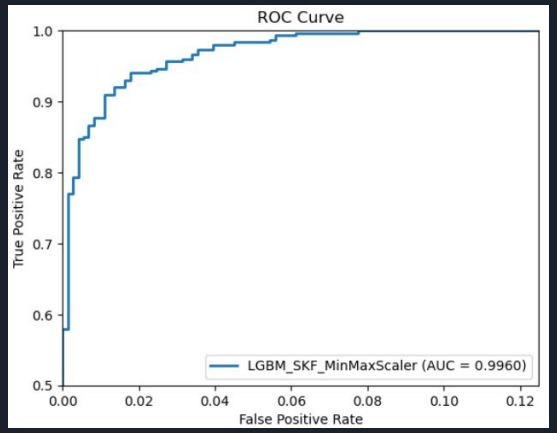


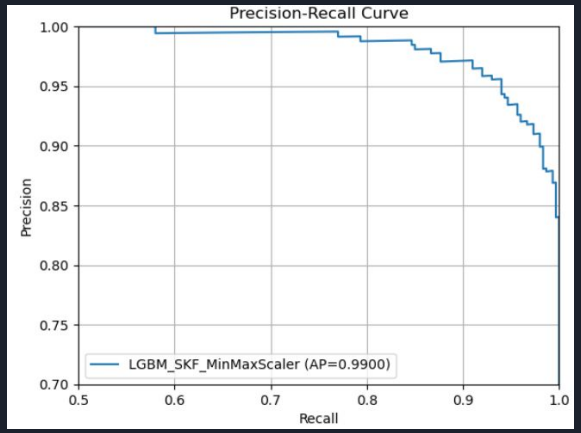

Class Weights: Slightly more overfit

MinMax Scaler Increased Performance

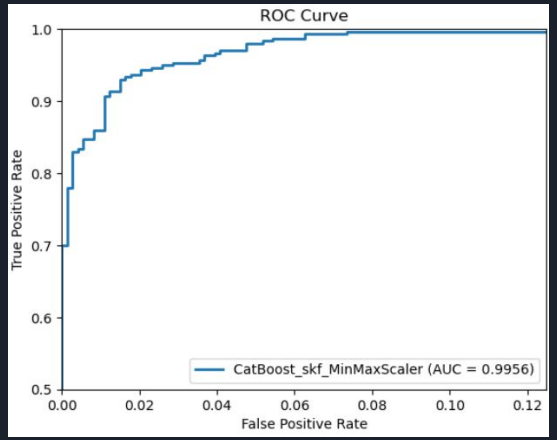
# LGBM vs CatBoost

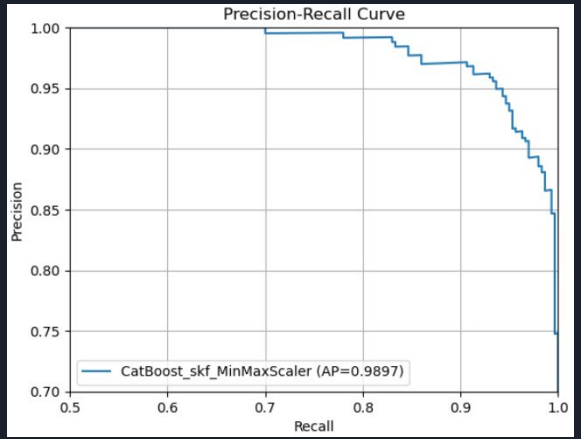
LGBM, SKF, Weighted




CatBoost, SKF, Weighted



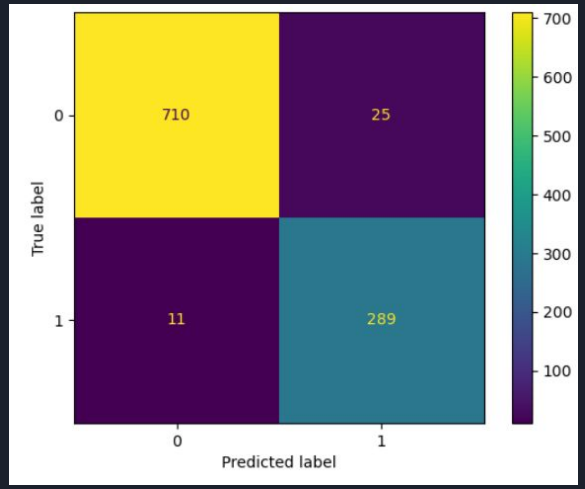



# LGBM vs CatBoost: 50% threshold

**LGBM, SKF, Weighted: 36 mistakes**

25 real emails in spam (0.92 Precision)

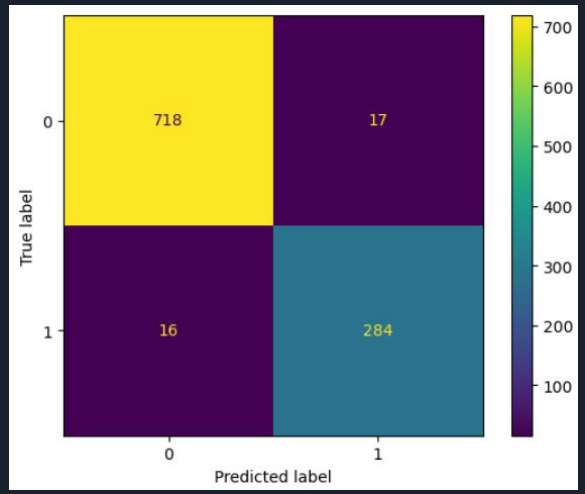
13 spam emails in inbox (0.96 Recall)




**CatBoost, SKF, Weighted: 33 mistakes**

17 real emails in spam (0.94 Precision)

16 spam emails in inbox (0.95 Recall)



# LGBM vs CatBoost:
# 1 Real Email
# In Spam
# (0.999 Precision)

LGBM, SKF, Weighted, 97% threshold:

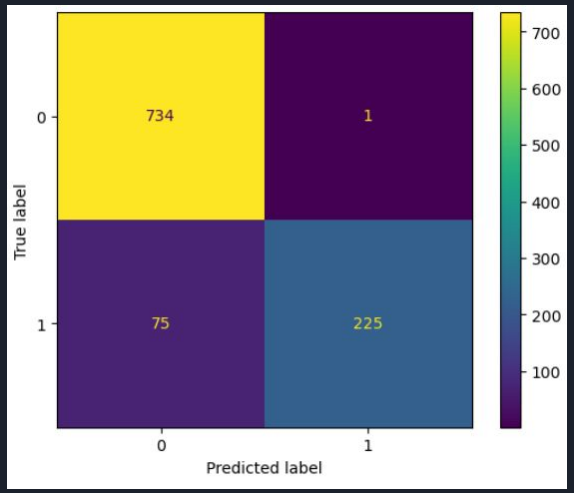
76 mistakes

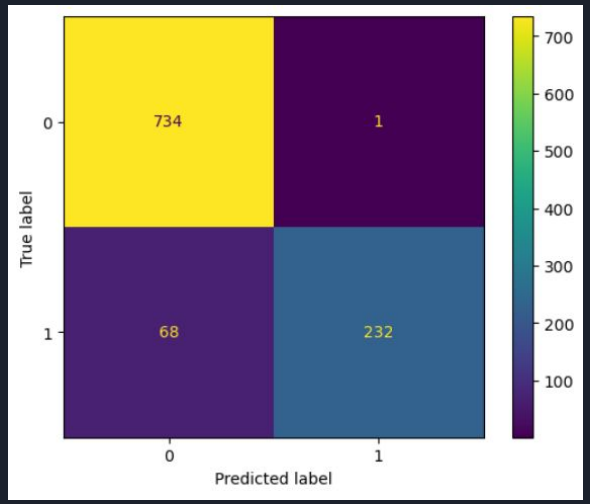
**75% Recall**

CatBoost, SKF, Weighted, 84.5% threshold

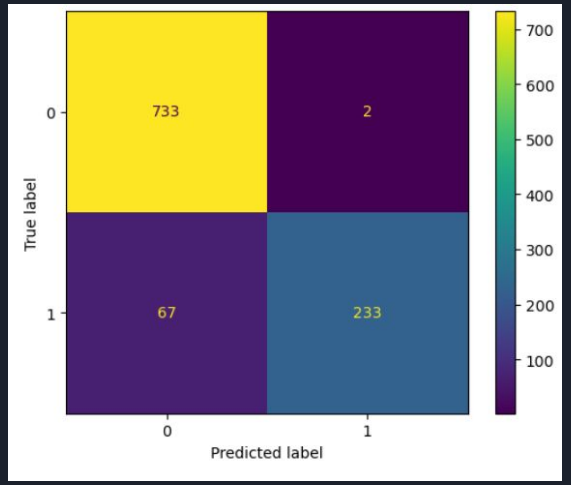
69 mistakes

**77% Recall**



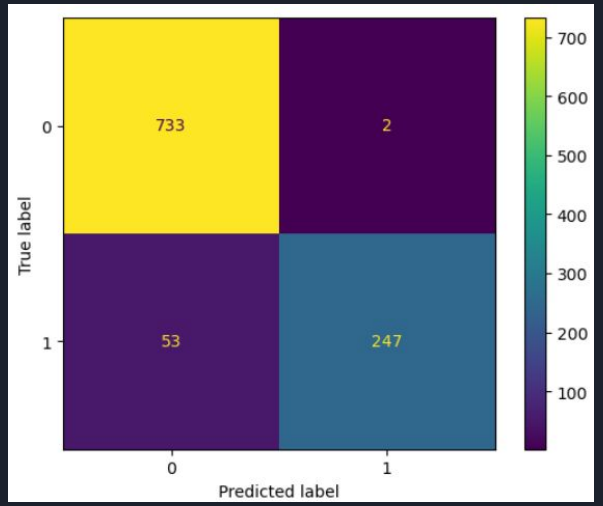
# LGBM vs CatBoost: 2 Real Emails In Spam (0.997 Precision)




LGBM, SKF, Weighted, 96.5% threshold

69 mistakes

**78% Recall**




CatBoost, SKF, Weighted, 80% threshold

55 mistakes

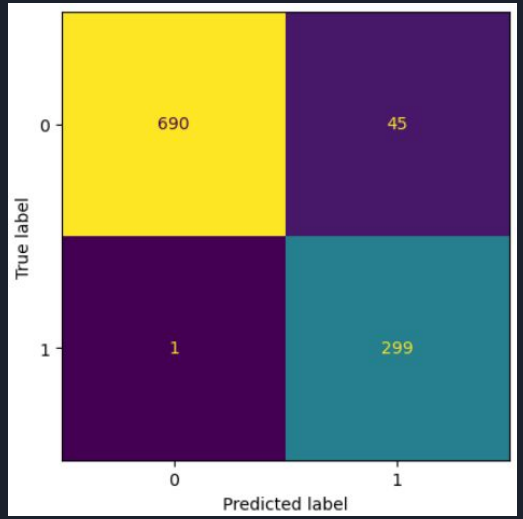
**82% Recall**

# LGBM vs CatBoost: Allowing 1 Spam Email (0.999 Recall)

LGBM, SKF, Weighted, 15%  threshold

46 mistakes

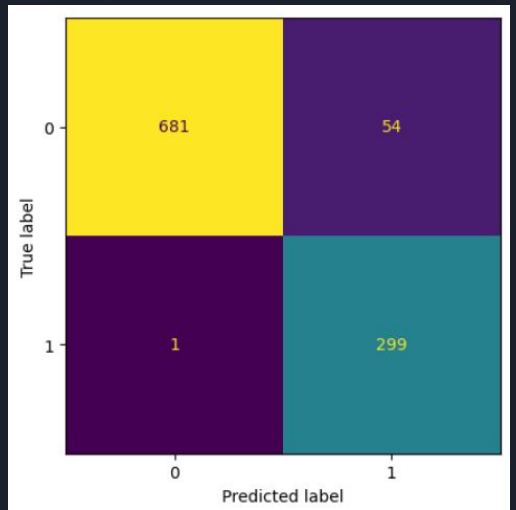
87% Precision




CatBoost, SKF, Weighted, 22% threshold

55 mistakes

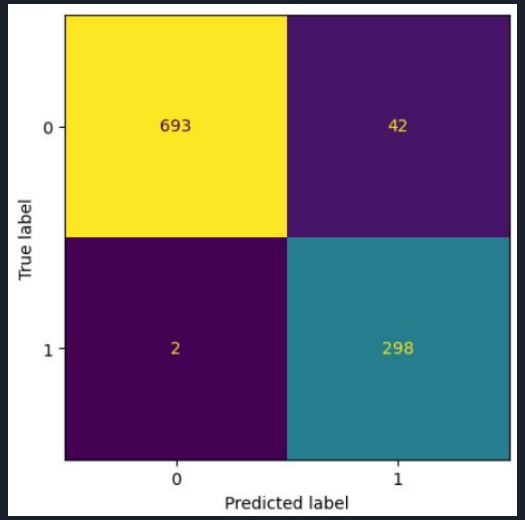
85% Precision



# LGBM vs CatBoost: Allowing 2 Spam Email (0.9933 Recall)

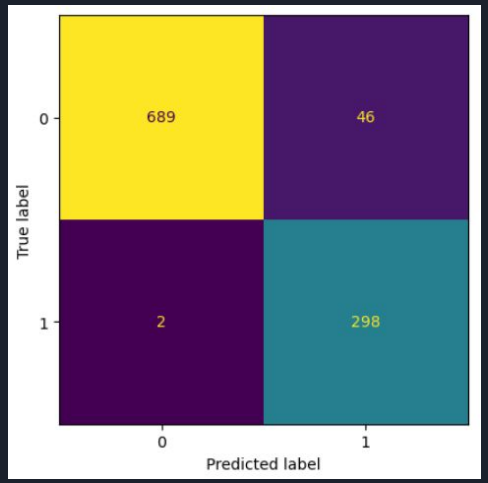
LGBM, SKF, Weighted, 20% threshold

- 44 mistakes
- 88% Precision




CatBoost, SKF, Weighted, 27% threshold

- 48 mistakes
- 87% Precision



# Feature Importances: LGBM vs. CatBoost

| Feature | |
|---|---|
| the | 55 |
| will | 52 |
| daren | 38 |
| z | 35 |
| day | 32 |
| deal | 32 |
| hp | 30 |
| questions | 29 |
| th | 29 |
| forwarded | 28 |
| v | 27 |
| employee | 26 |
| texas | 26 |
| mo | 26 |
| you | 26 |
| attached | 25 |
| s | 25 |
| p | 24 |
| money | 24 |
| ali | 24 |
| sitara | 23 |
| r | 23 |
| ii | 23 |
| volume | 23 |
| off | 22 |

| Feature | |
|---|---|
| daren | 5.133936 |
| attached | 5.015983 |
| http | 3.461221 |
| hp | 3.190983 |
| subject | 2.880842 |
| forwarded | 2.575656 |
| hanks | 2.300258 |
| gas | 2.108973 |
| the | 2.060274 |
| ali | 1.977688 |
| nom | 1.829881 |
| will | 1.806719 |
| thanks | 1.785052 |
| aren | 1.529001 |
| mo | 1.523938 |
| ii | 1.393912 |
| questions | 1.320140 |
| z | 1.315276 |
| deal | 1.201164 |
| for | 1.172264 |
| our | 1.081976 |
| meter | 1.029935 |
| sex | 0.986035 |
| houston | 0.879391 |
| th | 0.859483 |

# Conclusion

1) Relaxed Spam Filter: No real emails in spam box
   a) LogisticRegression, SSS, SMOTE: 13.33% of spam allowed
   b) CatBoost, SKF, no SMOTE: 16.33% of spam allowed

2) Strict Spam Filter: No spam allowed in inbox
   a) RandomForest, TrainTestSplit, SMOTE, unweighted: 7.82% of real emails in spam
   b) CatBoost, TrT, no SMOTE, Unweighted: 11.88% of real emails in spam

3) Balanced Accuracy
   a) LGBM, SKF, SMOTE: 2.95% of real emails in spam, 1.33% of spam emails inbox
   b) LGBM, SSS, no SMOTE: 3.39% of real emails in spam, 2% of spam emails in inbox