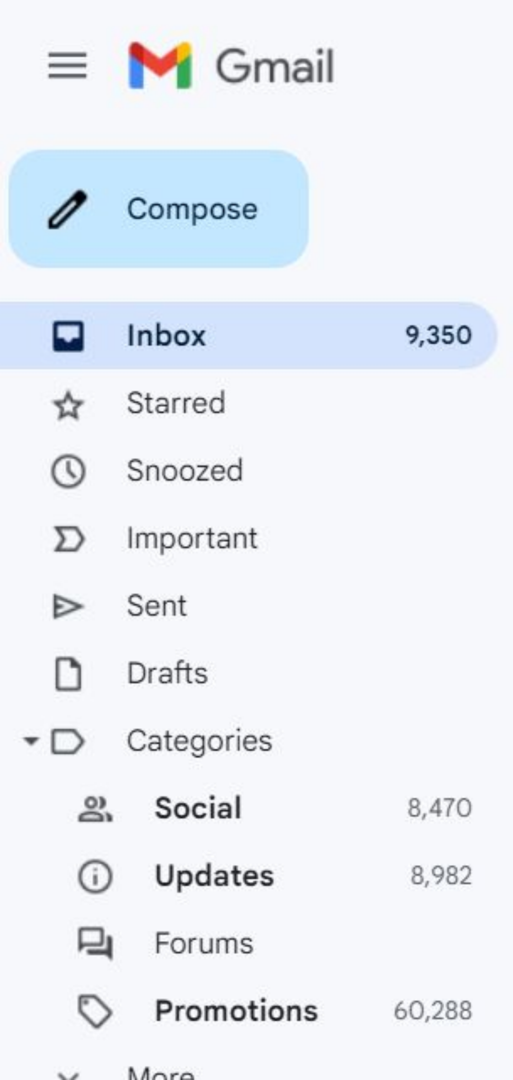
Ham vs Spam Emails

Project 4 Group 2

Esther Baumgartner
Matthew Byron
Angel Lee
Sam Schultz
Colin Vehmeier



INTRODUCTION

Problem:

- Receiving too many spam emails in your inbox
- Having to find a ham email in your spam/junk folders

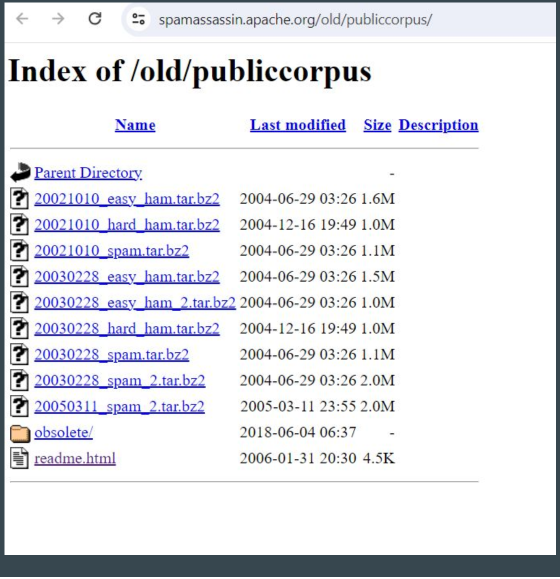
Question: How can we accurately sort spam and ham emails?

What are Spam and Ham emails?:

- **Spam**: unsolicited and unwanted junk email sent out in bulk to an indiscriminate
- **Ham**: non-spam email, “good/wanted” email

Data Source and Explanation of Data

- "Email Spam Dataset (Extended)" from Kaggle containing 9,000 files
- Original dataset came from SpamAssassin's Old Public Corpus which had about 6,000 files
- Altogether, contains 6,951 'ham' and 2,398 'spam' raw email files.
- Includes data between 2002 and 2005.
- Spam emails marked by use of HTML, unusual HTML markup, colored text, and "spammish-sounding" phrases



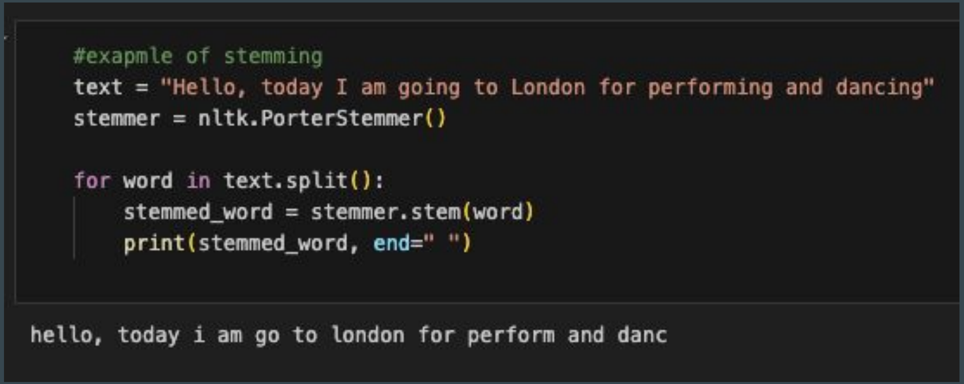
spamassassin.apache.org/old/publiccorpus/

Index of /old/publiccorpus

Name	Last modified	Size	Description
Parent Directory		-	
20021010_easy_ham.tar.bz2	2004-06-29 03:26	1.6M	
20021010_hard_ham.tar.bz2	2004-12-16 19:49	1.0M	
20021010_spam.tar.bz2	2004-06-29 03:26	1.1M	
20030228_easy_ham.tar.bz2	2004-06-29 03:26	1.5M	
20030228_easy_ham_2.tar.bz2	2004-06-29 03:26	1.0M	
20030228_hard_ham.tar.bz2	2004-12-16 19:49	1.0M	
20030228_spam.tar.bz2	2004-06-29 03:26	1.1M	
20030228_spam_2.tar.bz2	2004-06-29 03:26	2.0M	
20050311_spam_2.tar.bz2	2005-03-11 23:55	2.0M	
obsolete/	2018-06-04 06:37	-	
readme.html	2006-01-31 20:30	4.5K	

Data Cleaning and Preprocessing

1. Parse emails to just their text
2. BeautifulSoup to convert raw email files to plain text
3. Normalize text
    a. Lowercasing
    b. Removing punctuation
    c. Stemming
4. Create a vector of word counts



```
#exapmle of stemming
text = "Hello, today I am going to London for performing and dancing"
stemmer = nltk.PorterStemmer()

for word in text.split():
    stemmed_word = stemmer.stem(word)
    print(stemmed_word, end=" ")
```

```
hello, today i am go to london for perform and danc
```

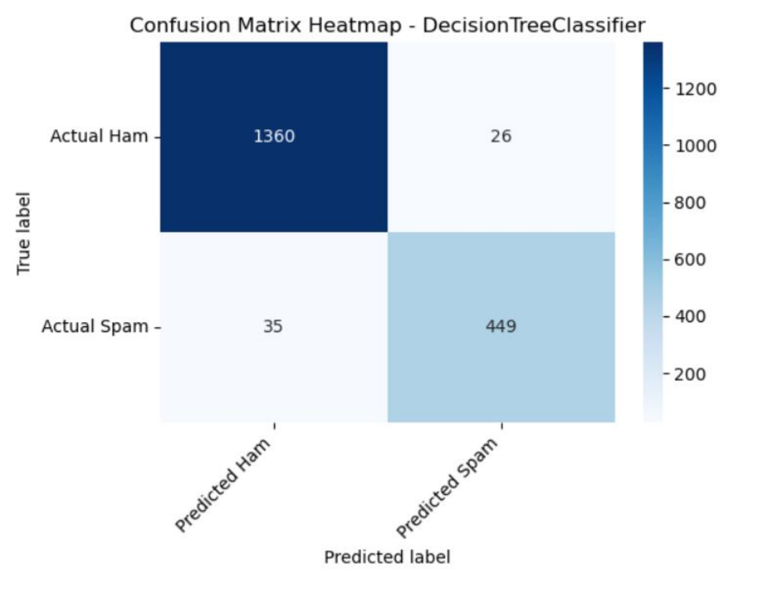
Models Used During Optimization Process

- Random Forest
- Decision Tree
- Logistic Regression

Decision Tree

"Tree-like model with nodes representing features and leaves representing outcomes. Used for both classification and regression."

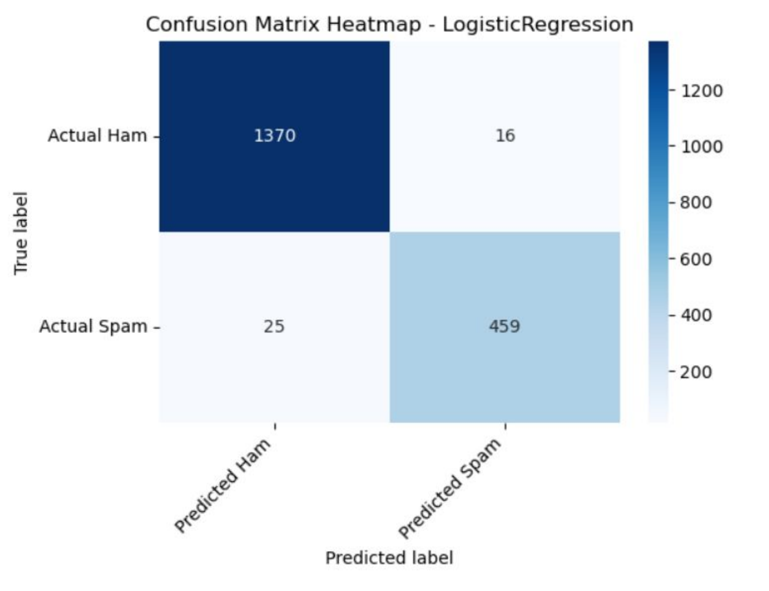
- 97% Accuracy
  - True Positive - 449
  - True Negative - 1360
  - False Positive (Spam) - 26
  - False Negative (Ham) - 35



Logistic Regression

"Linear model predicting probabilities for binary classification. Extends to multinomial logistic regression for multiple classes."

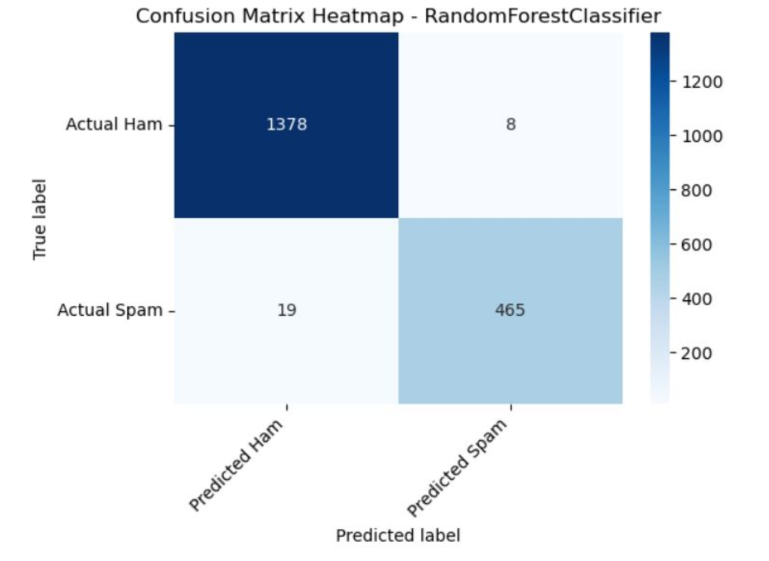
- Accuracy 98%
  - True Positive - 459
  - True Negative - 1370
  - False Positive (Spam) - 16
  - False Negative (Ham) - 25



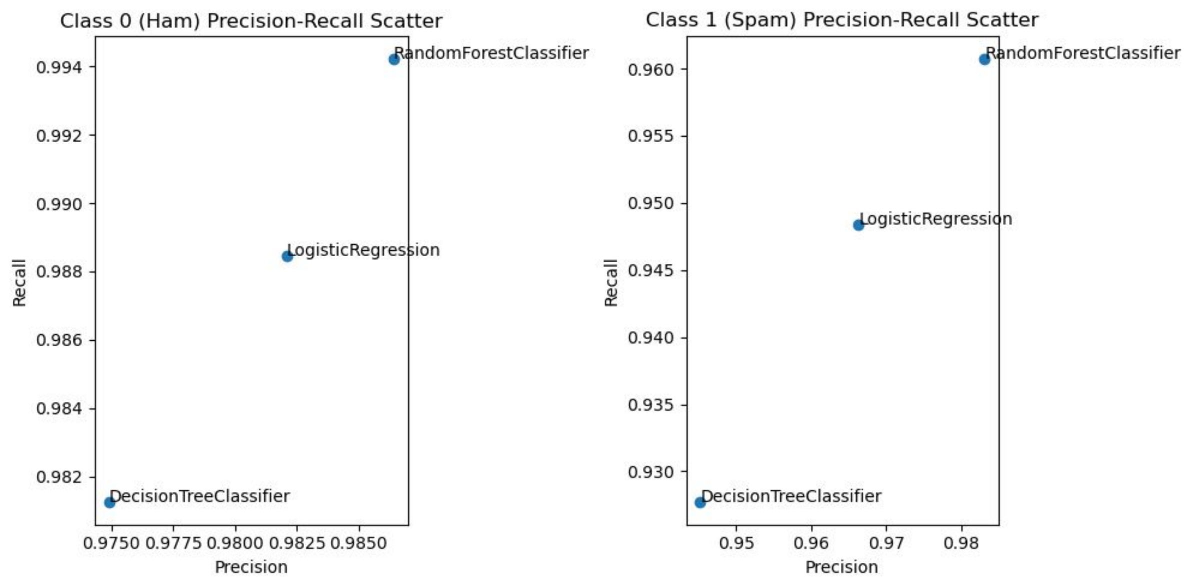
Random Forest Model

"Ensemble of decision trees. Each tree is built with randomness to improve generalization and reduce overfitting. Used for both classification and regression."

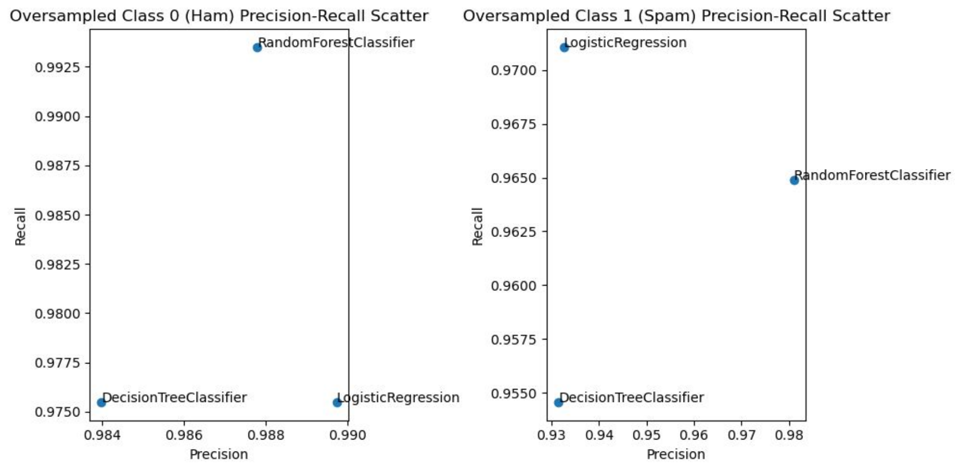
- Optimization - reducing false positives and false negatives.
- 99% Accuracy
  - True Positives - 465
  - True Negatives - 1378
  - False Positives (Spam) - 8
  - False Negatives (Ham) - 19

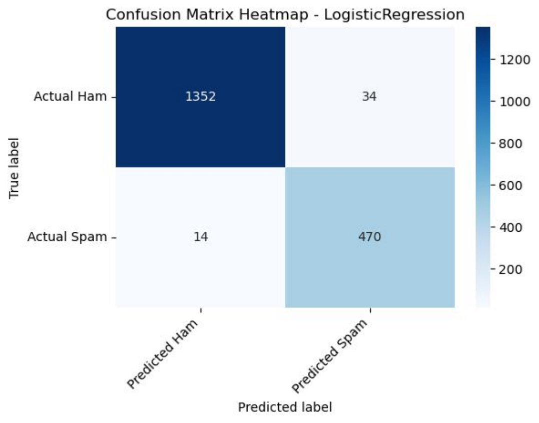


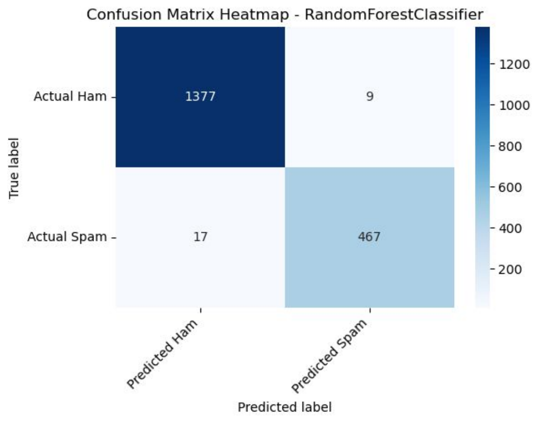
Comparison

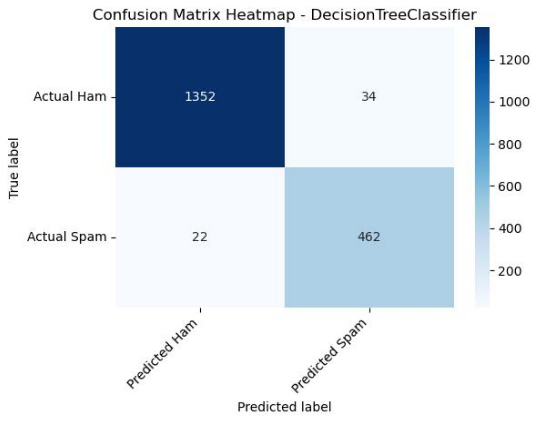

Class 0 (Ham) Precision-Recall Scatter
Recall
0.994
0.992
0.990
0.988
0.986
0.984
0.982
Precision
0.9750
0.9775
0.9800
0.9825
0.9850
RandomForestClassifier
LogisticRegression
DecisionTreeClassifier
Class 1 (Spam) Precision-Recall Scatter
Recall
0.960
0.955
0.950
0.945
0.940
0.935
0.930
Precision
0.95
0.96
0.97
0.98
RandomForestClassifier
LogisticRegression
DecisionTreeClassifier

Oversampling

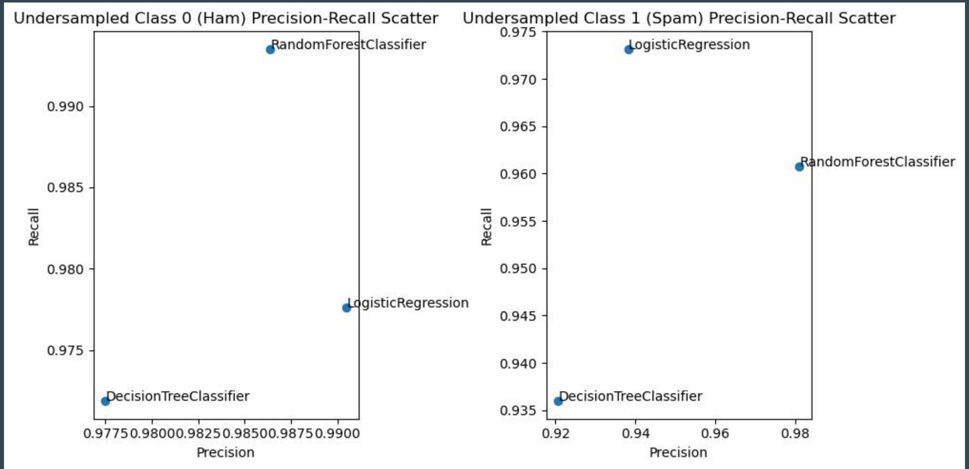


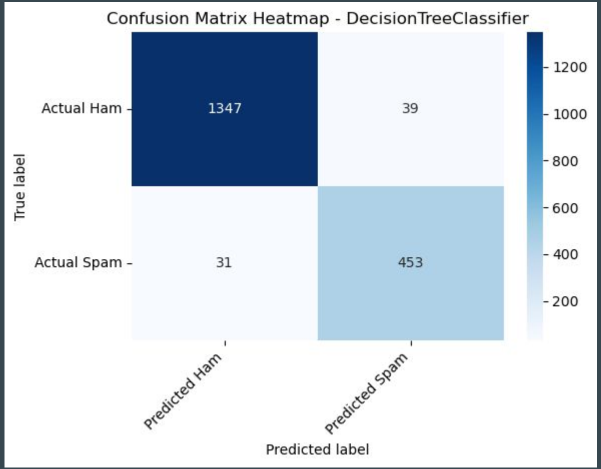


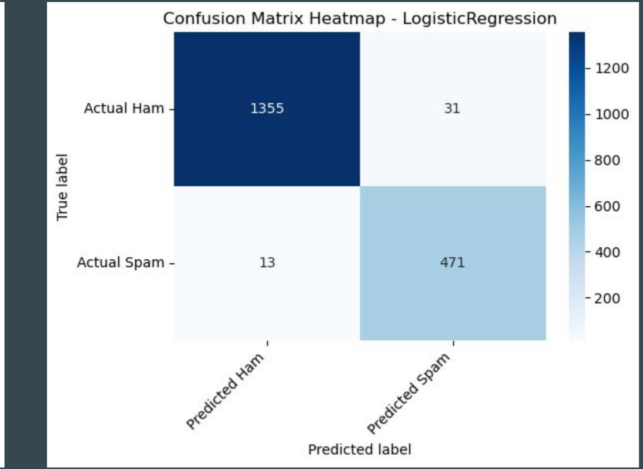


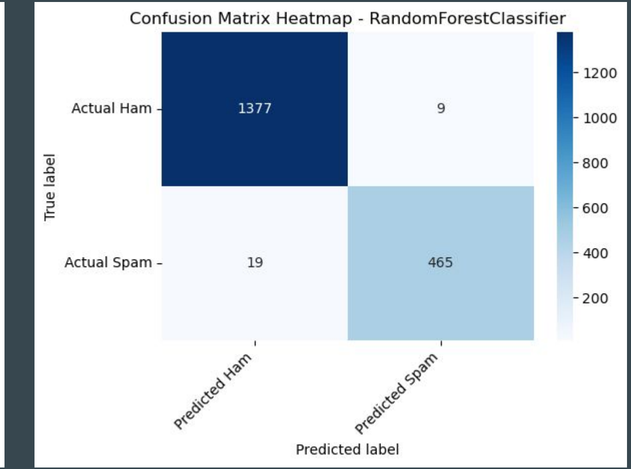


Undersampling

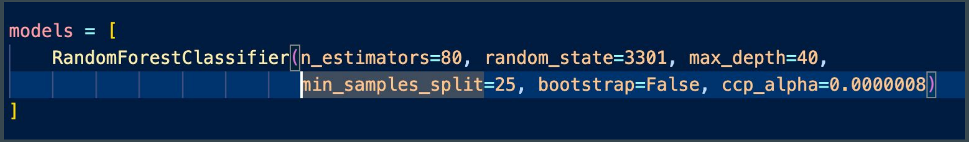




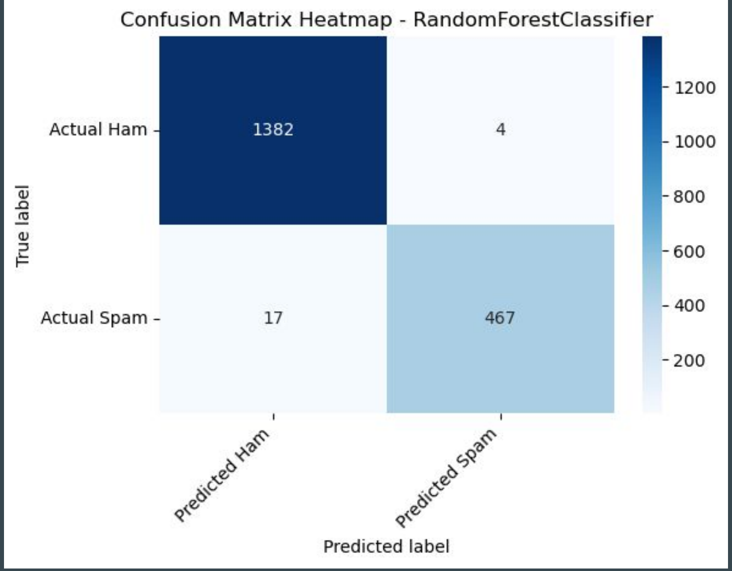




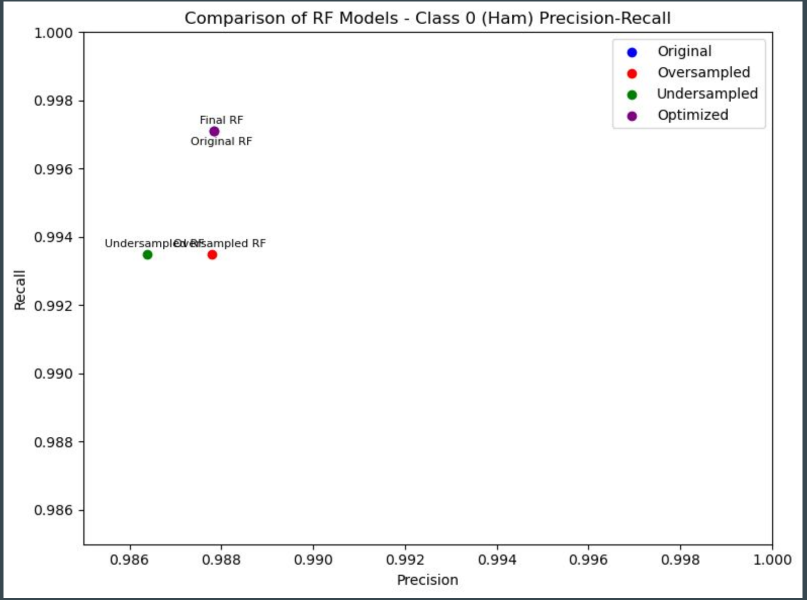
Hyperparameter Tuning



```
models = [
    RandomForestClassifier(n_estimators=80, random_state=3301, max_depth=40,
                           min_samples_split=25, bootstrap=False, ccp_alpha=0.0000008)
]
```



Comparison of Random Forest Models



Possible Future Steps

- Compare the accuracy of the computer model we created for this data set to a dataset containing more recent emails to see how the accuracy changes.
- Increase the number of most commonly used words in the word vector.
- Add any words that we believe could be important in distinguishing ‘ham’ vs ‘spam’ emails.

Resources:

Maharshipandya. (n.d.). Email Spam Dataset (Extended). Kaggle. Retrieved from: https://www.kaggle.com/datasets/maharshipandya/email-spam-dataset-extended

Spam Assassin’s Old Public Corpus. (n.d.). Index of/old/publiccorpus. Apache Software Foundation. Retrieved from: https://spamassassin.apache.org/old/publiccorpus/

Maharshipandya. (n.d.). Email Spam Classification [98%]. Kaggle. Retrieved from:https://www.kaggle.com/code/maharshipandya/email-spam-classification-98

Cisco. (n.d.). What Is Spam Email? Cisco. Retrieved from: https://www.cisco.com/c/en/us/products/security/email-security/what-is-spam.html

Cwiki. (2009, September 20th). Ham. CWiki. Retrieved from: https://cwiki.apache.org/confluence/display/spamassassin/Ham#:~:text=%22Ham%22%20is%20e%2Dmail,non%2Dspam%22%2C%20instead

Narkhede, S. (2018, May 9th). Understanding Confusion Matrix. Medium. Retrieved from: https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Parr, K. (2022, August 22nd). Markdown Cheat Sheet- How to Write Articles in Markdown Language. FreeCodeCamp. Retrieved from: https://www.freecodecamp.org/news/markdown-cheatsheet/

OpenAI. (n.d.) Can you explain to me what Random Forest, Logistic Regression, and Decision Tree models are? Retrieved on 2023, December 6th. Retrieved from: https://chat.openai.com/

Koehrsen, W. (2018, January 9th). Hyperparameter Tuning the Random Forest in Python. Retrieved from: https://towardsdatascience.com/hyperparameter-tuning-the-random-forest-in-python-using-scikit-learn-28d2aa77dd74

StackOverflow. (2023, March 10th). How to increase the model accuracy of logistic regression in Scikit python? StackOverflow. Retrieved from: https://stackoverflow.com/questions/38077190/how-to-increase-the-model-accuracy-of-logistic-regression-in-scikit-python

StackOverflow. (2020, June 5th). Improve precision of my predictive technique in Python. StackOverflow. Retrieved from: https://stackoverflow.com/questions/62224518/improve-precision-of-my-predictive-technique-in-python

SciKit Learn. (n.d.). Sklean.model_selection.RandomizedSearchCV. Scikit Learn Documentation. Retrieved from: https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.RandomizedSearchCV.html

StackOverflow. (n.d.). Proper use of “class_weight” parameter in Random Forest classifier. StackOverflow. Retrieved from: https://stackoverflow.com/questions/58275113/proper-use-of-class-weight-parameter-in-random-forest-classifier

StackOverflow. (n.d.). How to penalize False Negatives more than False positives. StackOverflow. Retrieved from: https://stackoverflow.com/questions/49151325/how-to-penalize-false-negatives-more-than-false-positives

Scikit Learn. (n.d.). Sklearn.ensemble.RandomForestClassifier. SciKit Learn Documentation. Retrieved from: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html

Brownlee, J. (2021, January 5th). Random Oversampling and Undersampling for Imbalanced Classification. Machine Learning Mastery. Retrieved from: https://machinelearningmastery.com/random-oversampling-and-undersampling-for-imbalanced-classification/

Thank you!

Special thanks to Hunter, Sam, and Randy. Questions?