0.1 Question 1a

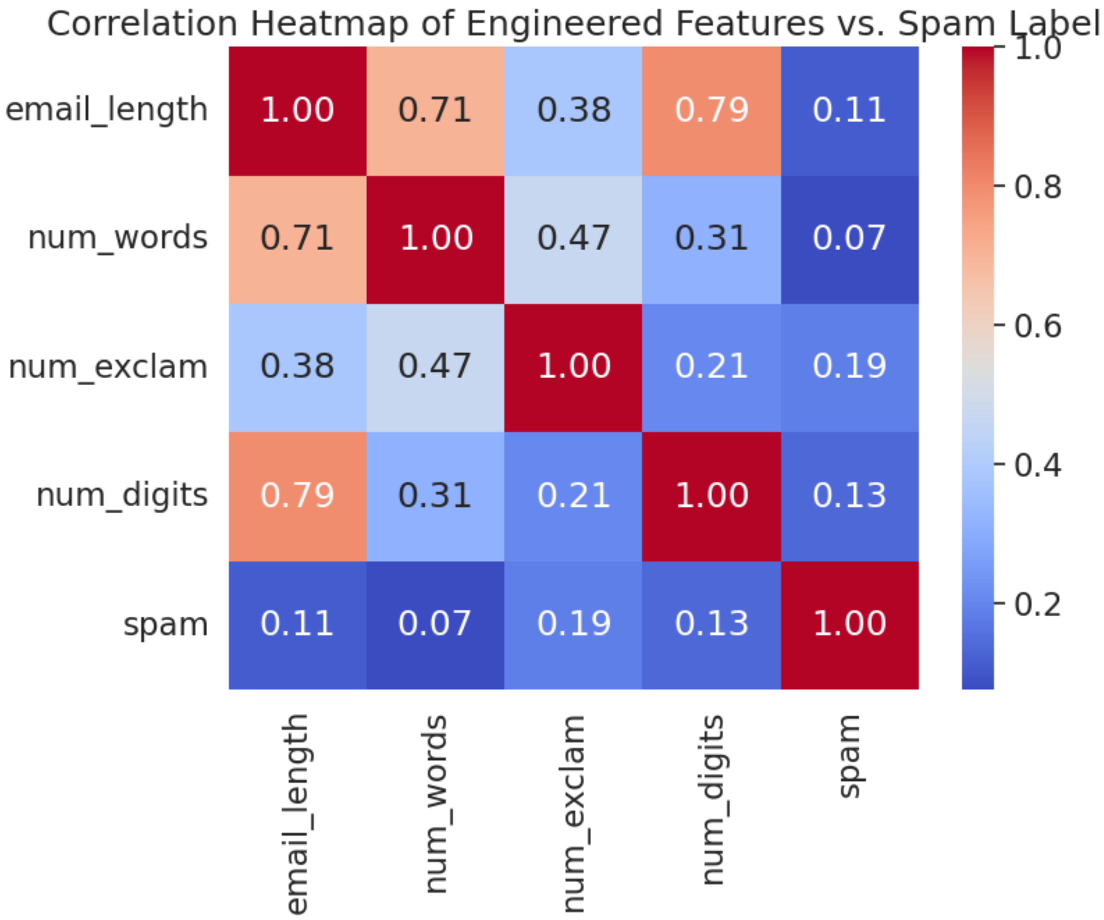
Generate your visualization in the cell below. As a friendly reminder, choose some plot other than the 1-dimensional distribution of some quantity for spam and ham emails.

```
In [54]: eda_df = train.copy()

         eda_df["email_length"] = eda_df["email"].str.len()
         eda_df["num_words"] = eda_df["email"].str.split().apply(len)
         eda_df["num_exclam"] = eda_df["email"].str.count("!")
         eda_df["num_digits"] = eda_df["email"].str.count(r"\d")

         corr = eda_df[["email_length", "num_words", "num_exclam", "num_digits", "spam"]].corr()

         plt.figure(figsize=(8,6))
         sns.heatmap(corr, annot=True, cmap="coolwarm", fmt=".2f")
         plt.title("Correlation Heatmap of Engineered Features vs. Spam Label")
         plt.show()
```


Correlation Heatmap of Engineered Features vs. Spam Label
email_length
num_words
num_exclam
num_digits
spam
1.00 0.71 0.38 0.79 0.11
0.71 1.00 0.47 0.31 0.07
0.38 0.47 1.00 0.21 0.19
0.79 0.31 0.21 1.00 0.13
0.11 0.07 0.19 0.13 1.00
email_length
num_words
num_exclam
num_digits
spam
1.0
0.8
0.6
0.4
0.2

0.2 Question 1b

In two to three sentences, describe what you plotted and its implications with respect to your features.

I plotted a correlation heatmap of several engineered features, such as email length, number of words, number of exclamation marks, and number of digits, and their relationship to the spam label. The plot shows that features like email length and the number of digits have moderate positive correlations with the spam label, suggesting they may be useful predictors. Meanwhile, exclamation marks and overall word count show weaker relationships, indicating they may contribute to distinguishing spam from ham.

1 Question 4

Describe the process of improving your model. You should use at least 2-3 sentences each to address the following questions:

1. How did you find better features for your model?
2. What did you try that worked or didn't work?
3. What was surprising in your search for good features?

1. To find better features for my model, I began by examining correlations between engineered variables and the spam label, which highlighted which patterns were actually predictive. I then added new features based on structural cues in emails, such as capitalization, punctuation, keyword presence, and subject-line patterns, which gave the model more meaningful signals.
2. Several attempts worked well, such as adding numeric features like digit counts, exclamation marks, and subject-line length, which noticeably increased accuracy. Other ideas, like overly broad keyword lists or redundant indicators, such as multiple ways of detecting the same word, had little effect or introduced noise, so I removed them.
3. The most surprising part of this process was how small, simple features, such as whether an email contains "$," "http," or excessive capitalization, had a much stronger impact than adding more sophisticated word lists. I also found that subject-line features were more informative than expected, suggesting that spammers follow consistent patterns in formatting their messages.

2 Question 5: ROC Curve

In most cases, we won't be able to get 0 false positives and 0 false negatives, so we have to compromise. For example, in the case of cancer screenings, false negatives are comparatively worse than false positives — a false negative means that a patient might not discover that they have cancer until it's too late. In contrast, a patient can receive another screening for a false positive.

Recall that logistic regression calculates the probability that an example belongs to a particular class. To classify an example, we say that an email is spam if our classifier gives it ≥ 0.5 probability of being spam. However, **we can adjust that cutoff threshold**. We can say that an email is spam only if our classifier gives it ≥ 0.7 probability of being spam, for example. This is how we can trade off false positives and false negatives.

The Receiver Operating Characteristic (ROC) curve shows this trade-off for each possible cutoff probability. In the cell below, plot an ROC curve for your final classifier (the one you use to make predictions for Pensieve) on the training data. Lecture 23 may be helpful.

Hint: You'll want to use the `.predict_proba` method (documentation) for your classifier instead of `.predict` to get probabilities instead of binary predictions.

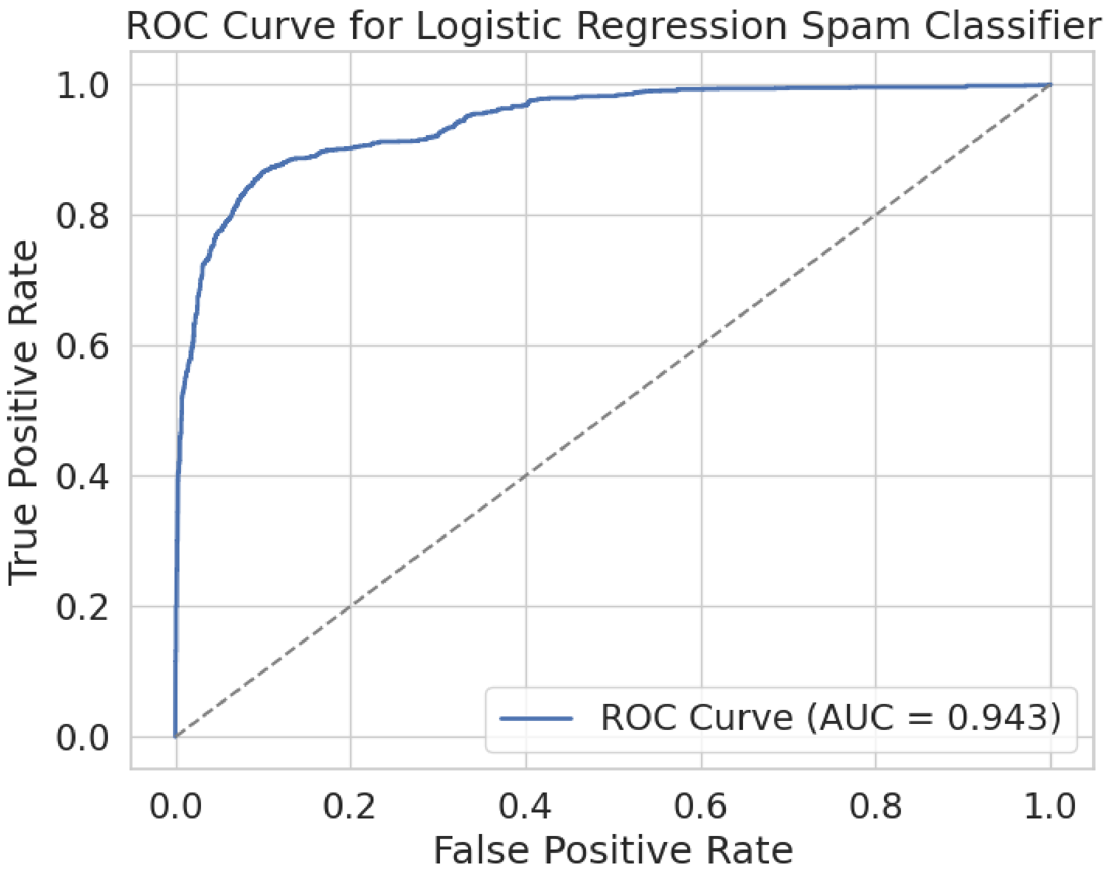
```
In [63]: from sklearn.metrics import roc_curve, auc

         # Get predicted probabilities for the POSITIVE class ("spam")
         # predict_proba returns an array of shape (n_samples, 2), where:
         # column 0 = probability of NOT spam, column 1 = probability of spam
         probs = my_model.predict_proba(X_train)[:, 1]

         fpr, tpr, thresholds = roc_curve(y_train, probs)

         roc_auc = auc(fpr, tpr)

         plt.figure(figsize=(8, 6))
         plt.plot(fpr, tpr, label=f"ROC Curve (AUC = {roc_auc:.3f})", linewidth=2)
         plt.plot([0, 1], [0, 1], linestyle="--", color="gray")  # baseline
         plt.xlabel("False Positive Rate")
         plt.ylabel("True Positive Rate")
         plt.title("ROC Curve for Logistic Regression Spam Classifier")
         plt.legend()
         plt.grid(True)
```


ROC Curve for Logistic Regression Spam Classifier
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
ROC Curve (AUC = 0.943)

2.0.1 Question 6a

Pick **one** of the emails provided above to comment on. How would you classify the email (e.g., spam or ham), and does this align with the classification provided in the training data? What could be a reason someone would disagree with *your* classification of the email? In 2-3 sentences, explain your perspective and potential reasons for disagreement.

I would classify Example 3 as spam, and this matches the label in the training data. The message contains commercial promotions, multiple external links, and aggressive language about saving money, which are patterns of typical spam emails. Someone may disagree with me because it also contains legitimate-looking travel information and disclaimers, which could make the email seem like a standard newsletter rather than malicious spam.

2.0.2 Question 6b

As data scientists, we sometimes take the data to be a fixed “ground truth,” establishing the “correct” classification of emails. However, as you might have seen above, some emails can be ambiguous; people may disagree about whether an email is actually spam or ham. How does the ambiguity in our labeled data (spam or ham) affect our understanding of the model’s predictions and the way we measure/evaluate our model’s performance?

Ambiguity in the labeled data makes the grounded truth less reliable, which can artificially lower our model’s measured accuracy even when the model is behaving reasonably. If the labels themselves reflect subjective judgments or inconsistent criteria, then errors may reflect disagreement with the labeler rather than a true mistake by the model. As a result, interpreting model performance becomes more complex; we must consider not only how well the model predicts but also whether the labels we evaluate against are themselves worthy.

Part ii Please provide below the index of the email that you flipped classes (`email_idx`). Additionally, in 2-3 sentences, explain why you think the feature you chose to remove changed how your email was classified.

I flipped classes with email_idx: 27. Removing the feature “bank” changed this email’s class because, in our simple model, words like “bank” are highly correlated with spam and carry a lot of weight in the logistic regression. When that word is present, the model’s predicted spam probability is about 55.6%, so it labels the messages as spam; once we drop the “bank” feature, the remaining words look much less spam-like, the probability falls to about 24.3%, and the classification flips to ham.

Part i In this context, do you think you could easily find a feature that could change an email's classification as you did in part a)? Why or why not?

Np, it would likely be much harder to find a single feature that flips that classification in a model with 1000 features. With so many predictors, the model's decision is distributed across many small signals rather than relying heavily on one word. As a result, removing any single feature usually had only a minor effect on the overall prediction.

Part ii Would you expect this new model to be more or less interpretable than `simple_model`?

Note: A model is considered interpretable if you can easily understand the reasoning behind its predictions and classifications. For example, the model we saw in part a), `simple_model`, is considered interpretable as we can identify which features contribute to an email's classification.

I would expect the new model to be less interpretable than the simple model. With 1000 features, the model's decisions are influenced by many small contributions, making it difficult to pinpoint which features meaningfully drive a prediction. In contrast, simple_model uses only a few clear features, so we can easily trace how one affects the classification.

2.0.3 Question 7c

Now, imagine you're a data scientist at Meta, developing a text classification model to decide whether to remove certain posts / comments on Facebook. In particular, you're primarily working on moderating the following categories of content: * Hate speech * Misinformation * Violence and incitement

Pick one of these types of content to focus on (or if you have another type you'd like to focus on, feel free to comment on that!). What content would fall under the category you've chosen? Refer to Facebook's Community Standards, which outline what is and isn't allowed on Facebook.

I choose to focus on Violence and Incitement. Under Meta's Community Standards, this category includes content that directly threatens, advocates for, or encourages violence against individuals or groups. Examples include explicit threats of physical harm, statements that incite others to commit violence, or praise/support for violent acts. Content that recruits others to engage in violence or targets people with calls for coordinated harm would also fall under this category. Meta removes such content because it poses a real-world safety risk and could escalate into offline harm. Even implicit or coded language may be removed if the surrounding context makes the intent to incite violence clear.

2.0.4 Question 7d

What are the stakes of misclassifying a post in the context of a social media platform? Comment on what a false positive and false negative means for the category of content you've chosen (hate speech, misinformation, or violence and incitement).

In the context of social media moderation for Violence and Incitement, the stakes of misclassification are extremely high. A false negative, or failing to remove content that encourages or threatens violence, can lead to real-world harm by enabling coordinated attacks, escalating harassment, or legitimizing violent behavior. It undermines user safety and exposes the platform to legal, ethical, and reputation risks. On the other hand, a false positive, which removes content that does not contain violent intent, can suppress legitimate expression, such as political advocacy, news reporting, or conversations about personal experiences with violence. This risks chilling free speech and disproportionately silencing certain users or communities. Because both types of errors have serious consequences, moderation models for this category must balance safety with fairness while being especially cautious about ambiguous or context-dependent language.

2.0.5 Question 7e

As a data scientist, why might having an interpretable model be useful when moderating content online?

Having an interpretable model is especially useful in online content moderation because moderators and users need to understand why a piece of content was flagged or removed. Interpretability helps ensure fairness and accountability by making it easier to identify whether a model is relying on appropriate signals rather than biased or irrelevant patterns. It also allows data scientists to debug harmful behavior, such as if the model disproportionately flags certain phrases or communities, and to justify moderation decisions to stakeholders. In a high-stakes setting like content moderation, being able to trace a prediction back to understand features builds trust, supports transparency, and helps ensure that the system aligns with policy goals and societal values.