Project A

CS135: Intro to Machine Learning

Model Behavior

Spring 2024

Jonathan Keusch, Theo Lanman

Problem 1: Bag-of-Words Feature Representation

1A: Bag-of-Words Design Decision Description

We represent documents as a **binary bag-of-words** using tokens extracted from the documents in the training set. We extract features from documents in three steps: pre-processing, token extraction, and feature value assignment. First, the document is pre-processed to convert all text to lowercase and strip accents using the Python `unicodedata.normalize` function with form “NFKD”. Then, tokens are extracted from documents as contiguous alphabetical unicode characters of length 3 or greater. Finally, each feature for a document is set to 1 if the document contains that token, or 0 otherwise. To construct the vocabulary, we extract tokens from each document in the training set, producing a **final vocabulary size of 4,359 tokens**. Tokens extracted from a document not in the vocabulary are ignored at prediction time. These pre-processing steps were selected using a grid search over lowercasing, accent-stripping, token length filtering, and binary-or-count feature values.

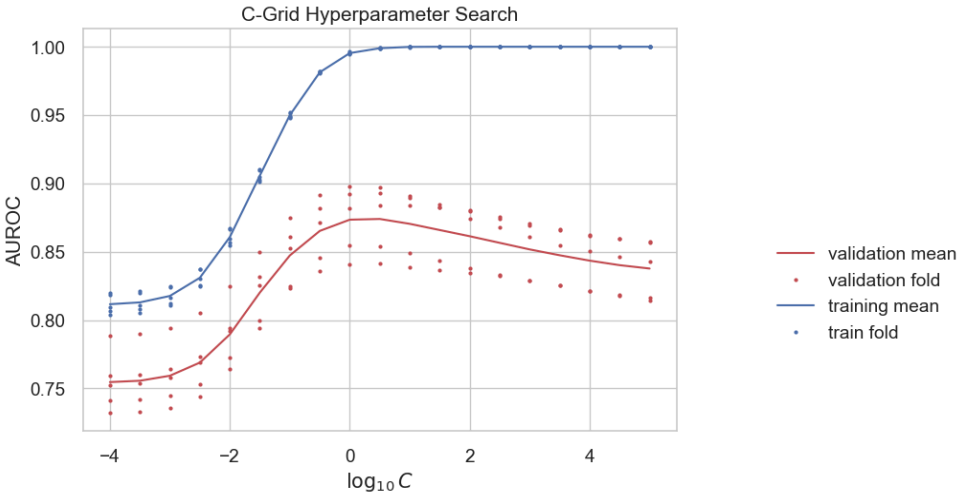
1B: Cross Validation Design Description

We selected model and preprocessing hyperparameters with **5-fold stratified cross-validation using AUROC scores**. Each fold contains 480 examples with the same proportion of classes as the entire training set. We evaluate using AUROC scores in order to select the best model for any decision threshold. We chose 5 folds to strike a balance between computation cost versus validation reliability. Finally, we use the hyperparameters with the greatest AUROC values to fit a model on the entire training set. To implement this cross validation approach, we used scikit-learn’s `GridSearchCV` class.

1C: Hyperparameter Selection for Logistic Regression Classifier

We used grid search with **5-fold stratified cross-validation** to select values for the hyperparameter C, the inverse of L2 regularization strength. Our C grid includes a search of **19 log-spaced values from 10^{-4} through 10^{5} inclusive**.

Figure 1: C-Grid Hyperparameter Search



In the figure above, we can observe values above $\log_{10}C=0.5$ where the validation score decreases and diverges from the training score, indicating overfitting. Conversely, we can observe underfitting in values less than $\log10_{10}C=0$ where both training and validation scores are less than optimal. Because we are able to observe both overfitting and underfitting within our search space, we are confident that we are able to choose a value that avoids underfitting and overfitting. The C value with the highest mean cross-validation AUROC score was **C=3.162 with AUROC=0.874**, which we selected for the final model.

1D: Analysis of Predictions for the Best Classifier

We analyzed errors across all of the validation folds for our selected model parameters. We observed an even split of 232 false positives and 231 false negatives, with an overall **error rate 0.193**. We also notice a higher prevalence of error with IMDB reviews, with the **IMDB error rate 0.228**, as well as with text containing negations (the word “not” or any word ending with “n’t”), with the **negation error rate 0.235**.

Below is a selection of one False Positive (FP) and one False Negative (FN) review from our set of errors.

Figure 2: Problem 1 Error Samples by Source and Type

Source	**Text**	**Error Type**
Amazon	“You get extra minutes so that you can carry out the call and not get cut off."	FN
IMDB	“To call this movie a drama is ridiculous!”	FP
IMDB	“Predictable, but not a badly watch.”	FN
Yelp	“Omelets are to die for!”	FN

In examining the content of the error reviews, we also notice a prevalence of shorter reviews despite individual words we aligned with the true overall sentiment. Generally, we observe the model struggles with nuanced semantics. For example, a review “But it is entertaining, nonetheless.” is a false negative, while other reviews, “A bit predictable” and “Highly unrecommended” are false positives. It seems negations (“not” and associated contractions) may subvert the model’s predictions.

1E: Test Performance

The **test set AUROC score for this model is 0.886**. This is higher than the estimated performance from cross validation of 0.874. We believe this reflects two factors: first, **differences between our held-out validation set and the final test set**; and second, **differences in the amount of training data our model sees at validation and test time**. During validation, each cross-validation fold is trained on only 1,920 examples, because 480 examples are held out. However, the final model is trained on the full training set of 2,400 examples. This increase in training data likely contributes to the final leaderboard score being higher than our estimates from held out data.

Problem 2: Open-ended Challenge

2A: Feature Representation Description

We represent features as a **bag of words with TF-IDF weighting concatenated with one-hot encoding for the website name**. TF-IDF weighting allows us to compare the count of words in a document against the presence of the word across documents in the corpus as a heuristic for its importance with the intent to improve the model's AUROC score, which we continue to use as our primary success metric. We added the one-hot encoding of the website name to try to reduce disparity between performance for reviews from different websites. We also performed a grid search over the preprocessing steps described in **1A**, with the addition of filtering for maximum and minimum document frequency. We selected **lowercase-only preprocessing and filtering tokens by maximum document frequency of 0.158**. **The final vocabulary size is 4,526**.

2B: Cross Validation Description

We performed the same cross-validation process as described in **1B** above, with 5-fold stratified cross-validation using AUROC scores.

2C: Classifier Description with Hyperparameter Search

We experimented with **L1 and L2 regularization**. For each, we used grid search with 5-fold stratified cross-validation to select values for the hyperparameter C, the inverse of regularization strength. Our C grid includes a search of **55 log-spaced values from 10^{-4} through 10^{5} inclusive**. Compared to Problem 1, we chose a finer grid to select a more precise optimal C value.

Figure 3: C-Grid Search with L1 Regularization

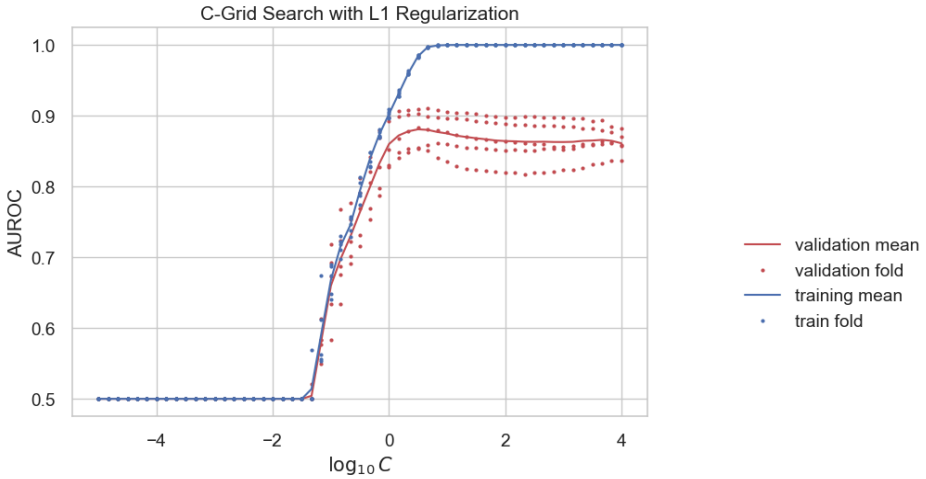
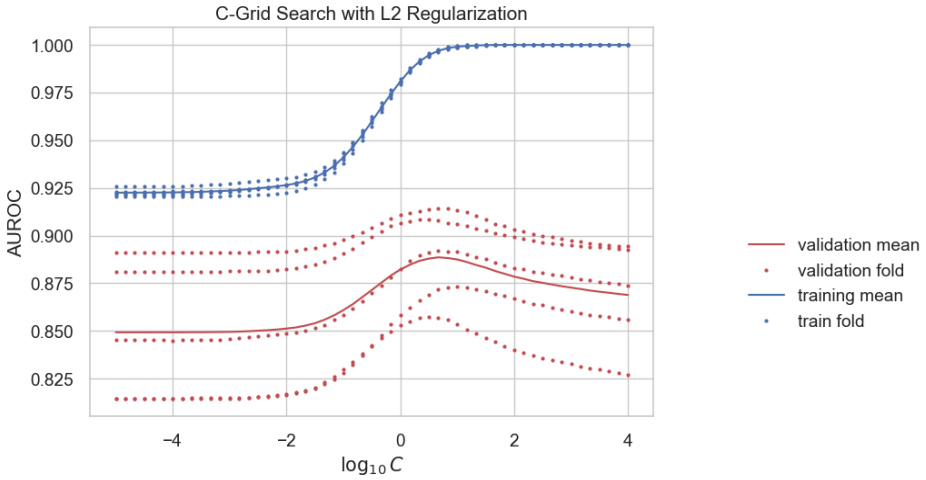


Figure 4: C-Grid Search with L2 Regularization



For both L1 and L2 regularization, we see that our search covers both underfitting, with low C values, and overfitting, with high C values. The peak AUROC for L1 regularization was **C=3.162 with AUROC=0.881**. The peak AUROC for L2 regularization was **C=4.642 with AUROC=0.889**. **We selected the model with L2 regularization because it had a higher AUROC score**. As in Problem 1, we used scikit-learn's `GridSearchCV` class to implement the search. **The AUROC of 0.889 improves on the score of 0.874 from Problem 1**.

2D: Error Analysis

We analyzed errors across all of the validation folds for our selected model parameters. We observed an even split of 232 false positives and 218 false negatives, with an overall **error rate 0.188** which is a lower overall error rate than Problem 1's 0.193. We continue to notice a higher prevalence of error with IMDB reviews, with the **IMDB error rate 0.218**, as well as with text containing negations (the word "not" or any word ending with "n't"), with the **negation error rate 0.223**, slightly lower than Problem 1's 0.235.

Below is the selection of one False Positive (FP) and one False Negative (FN) review from our set of errors in Problem 1. This table indicates whether the model in Problem 2 retained the error as examples.

Figure 6 identifies examples of reviews that the model in Problem 2 misclassified.

Figure 5: Problem 2 Error Samples by Source and Type

Source	Text	Error Type
Amazon	"I've bought $5 wired headphones that sound better than these."	FP
IMDB	"Totally different, with loads of understatement and black comedy, this is a film few get to see, but those who do will remember it."	FN
Yelp	"I dressed up to be treated so rudely!"	FP
Yelp	"The lighting is just dark enough to set the mood."	FN

In comparing persistent and different errors for Problem 2's model, we continue to observe the model's struggle with nuance and contradictory clauses. For example the first clause in the FP Yelp review in Figure 6 above begins as "I dressed up", which might be considered a positive clause, compared to the second clause "to be treated so rudely!".

2E: Test Performance

The **test set AUROC score for this model is 0.912**. This is higher than the estimated performance from cross validation of 0.889. It is also higher than the test set score from Problem 1 of 0.886. We think the improvement in test performance over validation performance is due to the same factors described in Problem 1. The improvement in test performance over the model from Problem 1 is expected, and tracks the improvement in validation performance.