CSE 142: HW2 Writeup
Some Linear Classification Models

Amir Noori

04/29/25

1 Introduction

This assignment covers two classification problems using the Perceptron algorithm, Logistic Regression, and Linear SVC. The first part was spam detection using the given Spambase dataset. Next is a language identification task using English and Dutch text data files given to us. I implemented the Perceptron based on the pseudocode from Lecture 07 and compared it to scikit-learn classifiers by comparing their accuracies and confusion matrices. All experiments were implemented in Python using Google Colab. The file is divided into cells for data loading, Perceptron training, model evaluation, and feature engineering.

2 Perceptron Algorithm

Our Perceptron implementation follows the perception training algorithm:

- Initialize all weights $w_d = 0$ and bias $b = 0$
- Iterate through the training data for a fixed number of epochs
- For each training example (x, y), compute activation $a = \sum w_d x_d + b$
- If $y \cdot a \leq 0$, update weights and bias: $w_d = w_d + yx_d$, $b = b + y$

This model was based on the basic pseudocode provided by the lecture 07 slides provided in class, then trained and tested against its scikit counterparts.

3 Spam Classification for Spambase

Using the Spambase dataset, I:

- Split the data into 80% training, 10% development, and 10% test sets
- Trained the Perceptron model from scratch for 100 iterations
- Compared it against Logistic Regression and Linear SVC from scikit-learn
- Observed that results vary slightly on each run due to randomness

Results on Test Set

Model	**Accuracy (%)**
Perceptron	73.54
Logistic Regression	91.32
Linear SVC	92.19

Confusion Matrices

Perceptron

	Pred. 0	**Pred. 1**
Actual 0	270	4
Actual 1	118	69

Logistic Regression

	Pred. 0	**Pred. 1**
Actual 0	263	11
Actual 1	29	158

Linear SVC

	Pred. 0	**Pred. 1**
Actual 0	264	10
Actual 1	26	161

4 Language Identification

Next, using the given text files, `english.txt` and `dutch.txt` with each file containing sentences from the Universal Declaration of Human Rights, they were also trained and tested under the differing classification models. Afterwards, another manually added 40 additional sentences were tested. Labels were 1 for English and -1 for Dutch. The features to differentiate between the two languages were also manually implemented.

Feature Extraction

Influenced by the class and by my observations of the two files I implemented features based on the following:

- Letter counts: e, a, o
- Word and sentence length
- Language-specific digraphs: "ij", "sch", "the", "de", "het"
- Endings: "ing" (English), "lijk" (Dutch)
- Special characters: accented vowels (Dutch)
- Keywords: "freedom", "rechten", etc.

Data Splits

- Training: 80% of given text files
- Development Sets/ Test Sets had 20 English sentences + 20 Dutch sentences each

Results

Training Set Accuracy

Model	Accuracy (%)
Perceptron	85.71
Logistic Regression	100.00
Linear SVC	100.00

Development Set Accuracy

Model	Accuracy (%)
Perceptron	65.00
Logistic Regression	87.50
Linear SVC	92.50

Test Set Accuracy

Model	Accuracy (%)
Perceptron	67.50
Logistic Regression	92.50
Linear SVC	92.50

Confusion Matrices

Perceptron

	Pred. 0	Pred. 1
Actual 0	10	10
Actual 1	3	17

Logistic Regression

	Pred. 0	Pred. 1
Actual 0	18	2
Actual 1	1	19

Linear SVC

	Pred. 0	Pred. 1
Actual 0	18	2
Actual 1	1	19

5 Conclusion

The perceptron performed decently well, however, it couldn't really compare to the very high results of the scikit models for Logistic Regression and Linear SVC. Both the scikit models gave extremely similar results, and it's difficult to determine which of the two is better, but both are clearly more effective than the perceptron. I noticed while creating and adding, and implementing the features that they were extremely important to the final results. The better, well-thought-out features helped give better results as it helped train the model to differentiate the data better. It was important to find features that wouldn't overfit or underfit the data and aim for the best results. Overall the assignment, testing the perceptron and other classification models under the spambase data set as well as implementing feature extraction for the language comparisons was very informative and reinforced my understanding of the concepts.