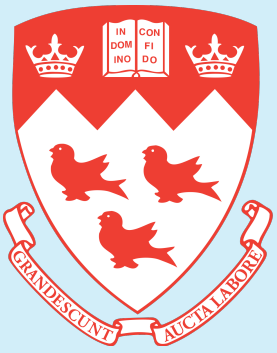


Applied Machine Learning

Assignment 2 Report
COMP 551

Authors: *Ling Fei Zhang, Brandon Ma, Giordano Di Marzio*

Instructor:	*Yue Li*
Department:	Computer Science
Date:	November 8, 2022

Contents

1 Abstract

In this assignment, we investigated the performance of two linear machine learning models on two benchmark datasets. The first one consisted of multiple features and binary classification, whereas the second dataset consisted of multi-class classification. For the first dataset, feature reduction techniques using z-scores were applied to reduce almost 90 000 features down to 200. We also analyzed the effect of different learning rates and training sizes on accuracy. We found that the learning rate of 1 scored the highest accuracy. The effect of training size was observed to be less impactful, as the accuracy reached a plateau at about 20% of the training size. We found that logistic regression achieved better accuracy (approx. 85%) than K - Nearest Neighbors (approx. 70.5%) and was significantly faster to train. The training speed of K - Nearest Neighbors was largely compromised due to the high dimensionality of our dataset. Ridge Regression and Linear Regression were also implemented and have shown to have a significant increase in accuracy compared to Logistic Regression for the first dataset, with both models scoring approximately 92%. For the second dataset, KNN acted very similarly, implying that it was repeatedly the slower model to train and was shown to have a much weaker accuracy than multiclass regression (approximately 55% vs 80% respectively). Attempting to better fit the learning rate, the best accuracy recorded was using a learning rate of 0.0005 at 80.52% accuracy.

2 Introduction

The main task of the assignment was to implement logistic regression and multiclass regression on two datasets. More precisely, the purpose of implementing logistic regression on the IMBD reviews was to predict whether the review would be positive or negative by the frequencies of keywords in positive or negative reviews respectively. This entailed a significant amount of preprocessing of the data, because it implied we needed to filter out words that would appear too often (such as determinants and other neutral words) or not often enough (less than 1% or more than 50%). We also excluded all stop words that were neutral to aid in our finding of keywords [1]. Additionally, we computed the z-scores for each word and chose the 100 most positive and 100 most negative words. In order to minimize the cost function used for our implementation of logistic regression, we used gradient descent, where we started from an initial assignment of the parameters w, and took small steps at each iteration, opposite to the gradient. It was observed that our cost function, for every iteration, continued to decrease until stagnation, which implied that our gradient had converged to a local minimum. This was performed for multiple learning rates, in order to maximize accuracy all while assuring the convergence of the gradient descent. It was also observed that the accuracy of the logistic model increased (from 65% to approximately 85%) when we increased the training size from 0.001% to 20%. When trying to fit the learning rate, it was observed that the rate of 1 scored the highest accuracy, following a similar trend whereas you increased the learning rate, the accuracy of the model increased. Finally, in order to ensure our gradient calculation and/or our loss function wasn't incorrect, we computed a small perturbation where we obtained a value of $4.23 \times 10^{-}24$ which was much smaller than the required $10^{-}8$ threshold.

In order to add to our experiment, we also decided to implement linear and ridge regression to try and predict the rating of an IMDB review. The accuracy reported for linear and ridge regression was very similar (approximately 92%), which was relatively higher than the accuracy reported for our logistic regression (approximately 84.44%).

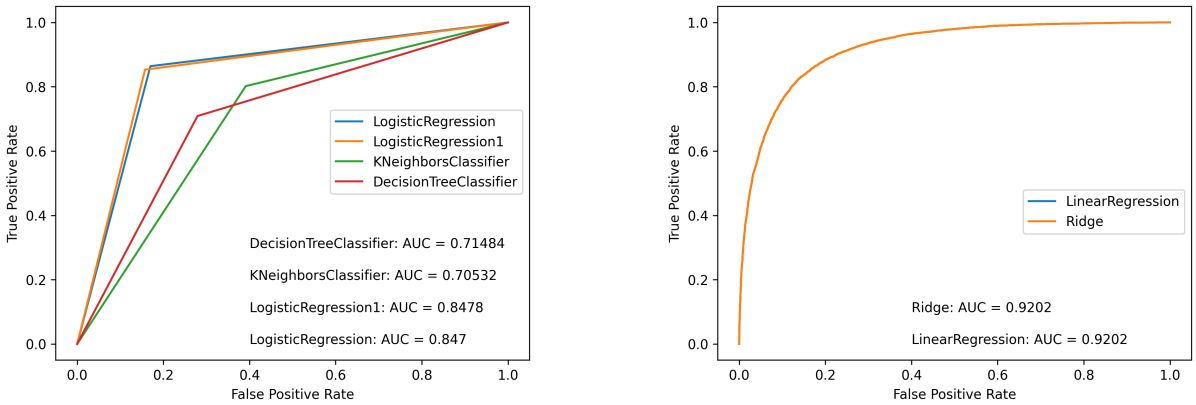
In regards to the 20 News groups, we proceeded by picking the top feature words per class using Mutual Information. A very similar process of filtering had to be used on the data (meaning words that were present under 1% and over 50% were not included) [1], but we also only considered 4 categories of words that also needed to be chosen . In our case, we chose "rec.sport.baseball', 'soc.religion.christian', 'comp.graphics' and 'rec.motorcycles' and we set the number of features to be equal to 100. We fit the

cost function very similarly, that is by using gradient descent and reporting the cost function's value at every iteration until stagnation towards 0. We then found that the optimal learning rate would be 0.0005 with an accuracy of approximately 80.52% when compared to other learning rates. As a comparative model, we were asked to compute KNN on the dataset, where we observed that its accuracy was significantly lower than the one reported for the multi class regression which was approximately 58% when using 60% for the training data, or 56% when using 80% for the training data. Similarly for both models, even though the accuracy slightly increased when increasing the training data size, it was clear that the multiclass regression generally had much better accuracy. A heatmap was used to show the magnitude of each word across all categories, where we saw that most of the group of words in a given category had been properly classified.

3 Datasets

The first dataset consisted of movie reviews written in English with a continuous rating score associated with the review. The first task of data processing was to convert the rating score into a binary classification, where any review rated 4 or below gets assigned the label 0 and any review rated 7 or above gets assigned the label 1. Next, we filtered out words that appeared in less than 1% or in more than 50% of the documents. We also filtered out stop words as well as special characters and single numerical digits. Lastly, Figure 3 shows the top 20 words calculated from their z-scores and their weight coefficients from Logistic Regression. We can clearly see that the majority of the words are overlapping. They also give a great representation of whether a movie is good or bad. The second dataset was loaded directly from a scikit-learn library consisting of various articles coming from 20 different categories of news. We only had to choose four to work with, which were the following: 'baseball', 'christian', 'graphics' and 'motorcycles'. We made sure to choose distinct categories to avoid overlapping words. Similarly to the first dataset, words that appeared in less than 1% or in more than 50% of the documents were filtered out. Additionally, any word appearing in scikit-learn's library of stopwords was also removed. After processing the data, we ended up with 2378 articles, each with the frequencies of 1081 tokens describing them.

4 Results



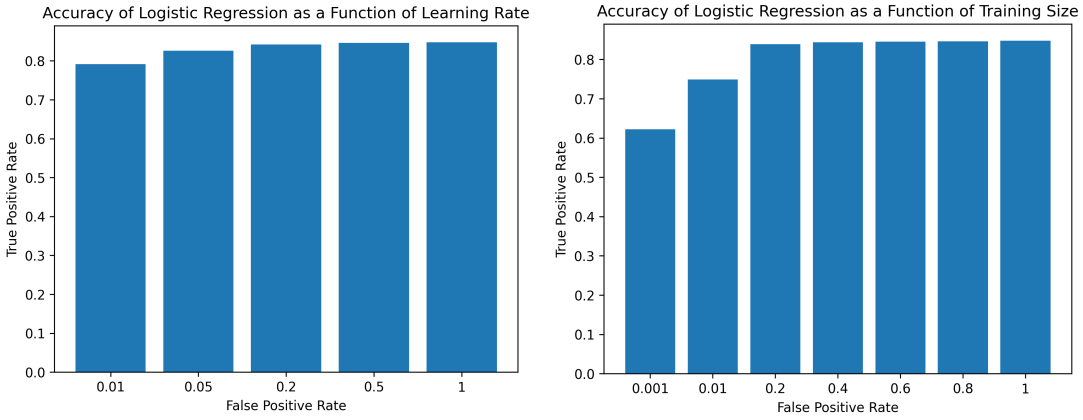
(a) AUROC of multi methods

(b) Ridge and Linear Regression

Figure 1: AUROC for Movie Reviews

It is important to note that LogisticRegression is the model imported from sk-learn, and LogisticRegression1 is the model we implemented from scratch. In the first dataset, we obtained high accuracy for

Logistic Regression, LogisticRegression1, Ridge Regression and Linear Regression. From Figure 1a, we can see that KNN and DT yielded slightly worse results.

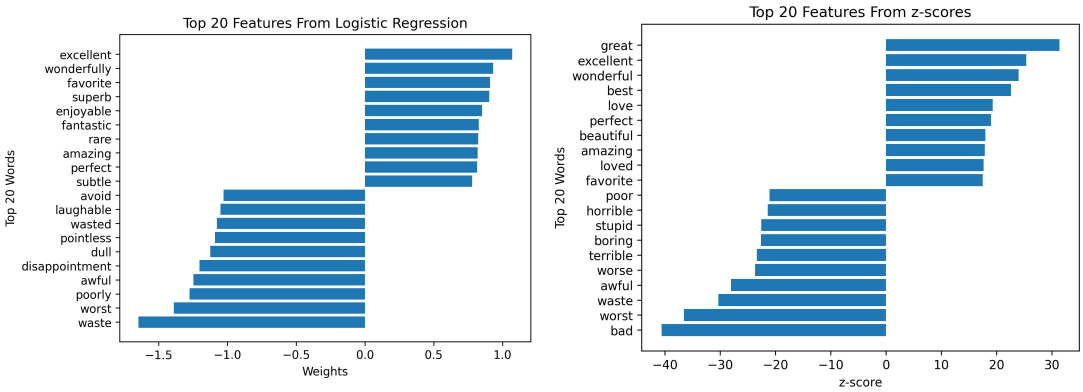


(a) Multiple Learning Rates

(b) Multiple Training Sizes

Figure 2: Movie Review Graphs

Both Logistic Regression models yielded similar accuracy scores and, from Figure 2a, our model was found to work best when the learning rate is set to 1. Furthermore, we observe that the accuracy of our model is only slightly affected by the training size. We can see from Figure 2b that the accuracy is very poor when we have 0.1% of the total training size, which is to be expected. To our surprise, at 1% of the training size, our accuracy has already increased significantly. At 20%, we can see that our accuracy reaches a plateau.



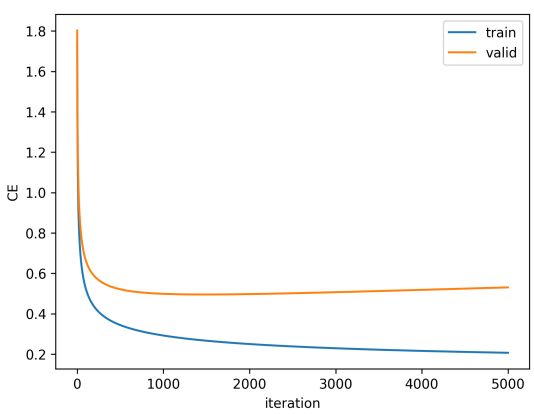
(a) Weights of Top 20 Words

(b) z-scores of Top 20 Words

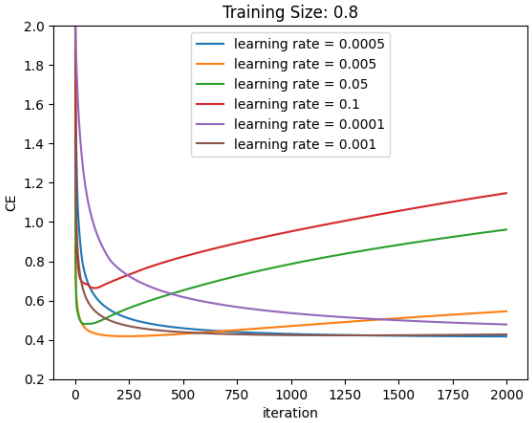
Figure 3: Top 20 Words

Next, Figure 3 tells us the importance of our top 20 words. We can easily see that both the z-scores and the logistic weights play an impactful role in telling us if a movie is good or bad.

For the second dataset, we chose $\alpha = 0.0005$. Using a validation set, we can make sure the data isn't being overfitted. Figure 4a also tells us the convergence point, which is when the validation curve flattens out (and starts to increase in some cases).



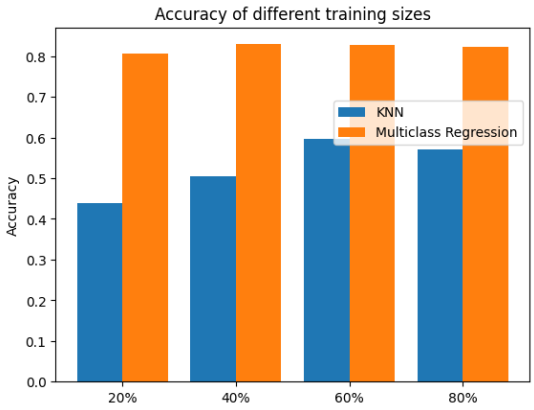
(a) Training Cross Entropy



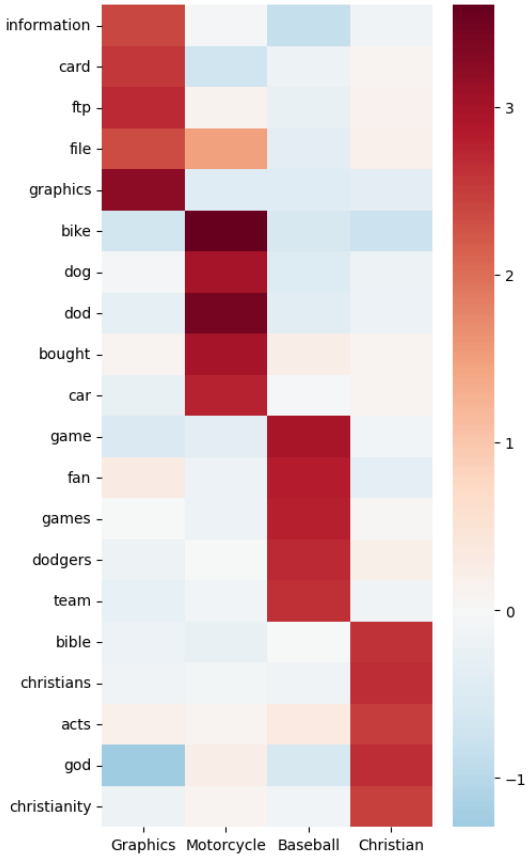
(b) 80% of training size

Figure 4: Cross Entropy for Second Dataset

We can see from Figure 5a that Multiclass regression is a much better classifier for this type of problem. The accuracies achieved by the regression are hovering around 80% while KNN struggles to get past 60%. From Figure 5b, we can see that the top 5 words from multiclass regression for each newsgroup are semantically related to the subject of each class.



(a) KNN vs Multiclass Regression



(b) Heatmap

Figure 5: 20 News Groups Graphs

For the 20 newsgroup dataset, we explored different learning rates for different training sizes. In order to compare them, we plotted their respective cross-entropy at each iteration. It was found from Figure 4b that a learning rate of $0.0005 \leq \alpha \leq 0.001$ gave the best accuracy. Setting it higher would result in the gradient diverging quickly and any lower would take a very long time to converge. Below is the cross entropy vs iteration graph for a training size of 80%.

5 Discussion and Conclusion

In the first dataset, Logistic Regression yielded high accuracy when classifying movie reviews, relative to its comparison model (KNN), which generally wielded poor accuracy. We also tested our model on different learning rates: a smaller rate would require a greater number of iterations to converge but would lower the risk of overshooting. On the other hand, a larger learning rate could converge faster but would come with a greater risk of overshooting. However, in our case, we observed that our model scored highest with a learning rate of 1. We also tested the impact of the training size as a function of the accuracy and found that the accuracy increases much slower after the first 20% of the training size. This is due to the enormous sample size. As a consequence, 20% of our entire 25 000 data points still amounts to a large enough training sample of 5 000. This can be interpreted as follows: the most important information comes from the first 20% of the training set, and the other 80% "repeats" a lot of the same information. However, the best accuracies were recorded for the linear and ridge regressions despite fitting the learning rate and the training size. In the second dataset, multiclass regression also yielded relatively high accuracy when classifying words to one of the 4 chosen class labels while KNN yielded poor results and significantly lower training speed. Having tested our model on different learning rates, and on multiple proportions of the training data size, multiclass regression performed best with alpha being equal to 0.0005 and with the training proportion being 80%. We also saw the accuracy increase less significantly after the first 20% of the training size. This is due once more to the repeats of information in our data.

6 Subject Extension/Originality

Word processing is an extremely relevant topic for Machine Learning given the accessibility of textual information through social media. Similar to the IMDB reviews, where we tried to evaluate where words were linked to positive or negative reviews, sentiment analysis is often used in text analysis in order to originate the emotions behind a sentence or a set of words. That being said, sentiment analysis is much more powerful when combined with Deep learning and NLP, natural language processing. Thus, it would be extremely interesting to see how accurate a model would be using NLP and deep learning to classify the data presented for IMDB reviews [2].

7 Statement of Contribution

Work was equally distributed among the three team members.

References

[1] A. C. Ashok Kumar Durairaj, "Sentiment analysis on mixed-languages," 2021. [Online]. Available: https://ieeexplore-ieee-org.proxy3.library.mcgill.ca/stamp/stamp.jsp?tp=&arnumber=9676277&tag=1

[2] H. Zheng, "Uber's deep learning applcations in nlp and conversational ai," 2020. [Online]. Available: https://www.oreilly.com/videos/ubers-deep-learning/0636920371113/