COMP 551 Project 2 Report

Krystal Xuejing Pan
260785873

Nghi Huynh
260632588

Corinne Robert
260795131

Abstract

This project investigates the performance of multi-class logistic regression and compares it against K nearest neighbour (KNN) classifier. We implement a softmax regression model with gradient descent and momentum from scratch, complete with 5-fold cross-validation. We also experiment with different hyperparameter and termination conditions for the gradient descent optimizer and see how they affect performance by applying grid-search. We have found the final test set accuracy between the implemented softmax model and the scikit-learn KNN model to achieve similar performance in both datasets, with the KNN model has a slightly higher accuracy score. For hyperparameter tuning, we found that decreasing the learning rate decreases the average validation accuracy. However, decreasing β, λ increases the average validation accuracy. Even though the average validation accuracy was the same, the run time of cross validation and test set were faster as we decreased the epsilon factor in the termination condition.

1 Introduction

In this project, we implemented a softmax regression model with mini-batch gradient descent, momentum, L_2 regularization and 5-fold cross-validation. We ran our model on two datasets, the digits dataset from the Scikit-Learn package and the letter datasets (version 1) from Openml. Each dataset had 10 and 26 classes respectively, and there were no missing data in either dataset. Since our logistic regression model had 4 hyperparameters: the learning rate α, the momentum β, the batch size B and the regularization parameter λ, we performed a simple grid-search to find the optimal combination of hyperparameter for our model. We also implemented multiple termination conditions in the gradient descent optimizer, including a maximum number of iterations, a small change in the gradient or an increase (or a plateau) in the validation error (cost). Our model achieved a 94% validation accuracy and 90% test accuracy on the digits dataset, while a KNN model with K = 1 achieved a 97% validation accuracy and a 95% test accuracy on the same dataset. As for the letter dataset, our model obtained a validation accuracy of 71% and test accuracy of 71%, while a KNN model with K = 1 achieved a validation accuracy of 94% and test accuracy of 94%.

2 Datasets

The first dataset (the digit dataset) is provided by project guidelines and accessed by Scikit-Learn. It is a digit classification dataset: each data point is an 8x8 image, and each label is a digit from 0 to 9. Thus, there are ten different classes. There are 1800 instances, and the numbers of instances belonging to each class have a mean of 179.7 with a standard deviation of only 2.6. Therefore, the instances are roughly evenly distributed in the digit dataset. The second dataset (the letter dataset) is accessed through Openml. It is a letters dataset with 26 different class labels from A to Z. There are 20000 instances, and the numbers of instances belonging to each class have a mean of 769 with a standard deviation of 22. Thus, the instances in the letter dataset are roughly evenly distributed. We normalized all the features of each dataset before running our model, and we kept the last 20% of both datasets as the test set to obtain an unbiased evaluation of our model.

3 Results

3.1 Optimal validation and test accuracy of the softmax model

By performing a grid-search, we found that the optimal set of hyperparameter to be $\alpha = 0.2$, the momentum $\beta = 0.9$, B = $1/3$ the size of the validation set and $\lambda = 0.001$ for the digits dataset. The optimal results of the softmax regression model on the digits dataset can be foun in fig 1. For the letters dataset, we found that the optimal set of hyperparameters to be $\alpha = 1$, the momentum $\beta = 0.999$, B = $1/3$ the size of the validation set and $\lambda = 0.001$.The optimal results of the softmax regression model on the letter dataset can be found in fig 2.

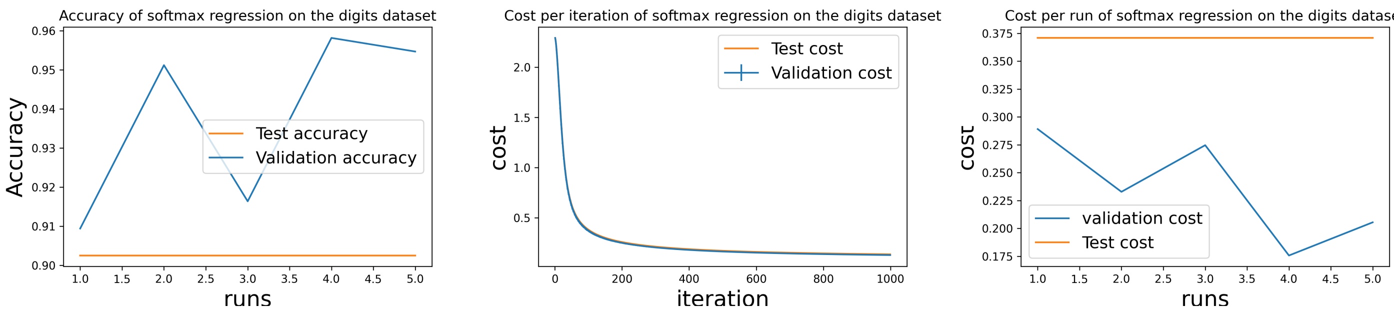

Figure 1: Validation and test accuracy (fisrt graph), cost per iteration (middle graph) and cost per run (last graph) of the softmax regression model with a 5-fold cross-validation on the digits dataset. Notice that the test was only carried out once but is displayed as being carried on each run of the cross-validation in the graphs only for displayed purposes.

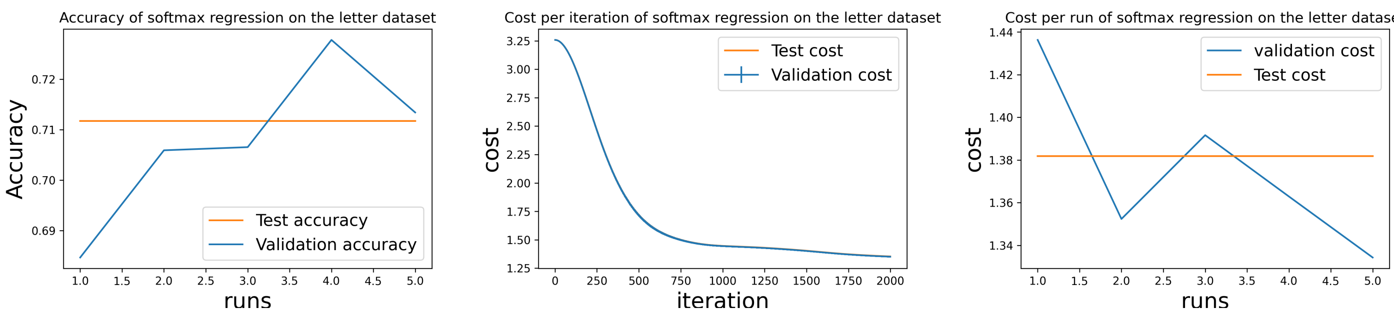

Figure 2: Validation and test accuracy (fisrt graph), cost per iteration (middle graph) and cost per run (last graph) of the softmax regression model with a 5-fold cross-validation on the letter dataset. Notice that the test was only carried out once but is displayed as being carried on each run of the cross-validation in the graphs only for displayed purposes.

3.2 Hyperparameter tuning and termination condition

To better understand the effect of different hyperparameters, we altered one hyperparameter at a time; we found that despite a faster run time of cross validation, the average validation accuracy decreased as alpha decreased. This is shown in the first row of fig 3. As the learning rate α decreased, we also see a slower convergence of the weights shown in the second row of the of fig 3, where the validation cost converges towards the asymptote more slowly as α decreases. We also saw an increase in the average validation cost as α decreases (last row of fig 3). As we altered β while keeping the other hyperparameter fixed, we observe that as β increased , the average validation accuracy decreased (first row of fig 4), there was a slower convergence towards the asymptotic validation cost (second row of fig 4) , and the average validation cost increased (last row fig 4).For lambda, we observed that the average validation accuracy was higher as lambda got smaller. We also observed a steeper convergence towards the asymptotic validation cost as lambda increased, although the average validation cost was higher as lambda increased. Finally, when we altered the size of the validation set, we found that the average validation accuracy decreased as the value of the validation size increased. Although we only displayed the effect of varying our hyperparameters on the digits dataset, the effect of varying the hyperparameters α, β, λ and the batch size were the same for the letter datasets. For termination conditions, we decided to stop the gradient descent algorithm when the difference between two consecutive iterations (epsilon) is less than e^{-8}, when the validation cost did not decrease for 20 consecutive iterations and when the number of iterations exceeds 1000 for the digits dataset and 2000 for the letter dataset. We observed that the run time of cross validation and test were faster as we decreased epsilon.

3.3 Comparison with KNN

We ran the multiclass KNN classification model from the Scikit-Learn package with neighbours K ranging from 1 to 100 to find the optimal hyperparameter for both datasets. As shown in fig 5, the KNN model that achieved the highest validation accuracy for both the digits and letter datasets was KNN with K = 1. It is important to note that the hyperparameter tuning of the KNN model was easier and faster than for our softmax regression model, as there is only one hyperparameter in the KNN model and the running time of KNN was overall smaller than the running time of our model.

Compared to our multi-class logistic regression with minibatch gradient descent, momentum and L_2 regularization with 5-fold cross-validation, the KNN classification model 5-fold cross-validation performed better on both the digits and the letter dataset and had a faster run time. Indeed the optimal KNN model had a validation accuracy of 97% and test accuracy of 95%

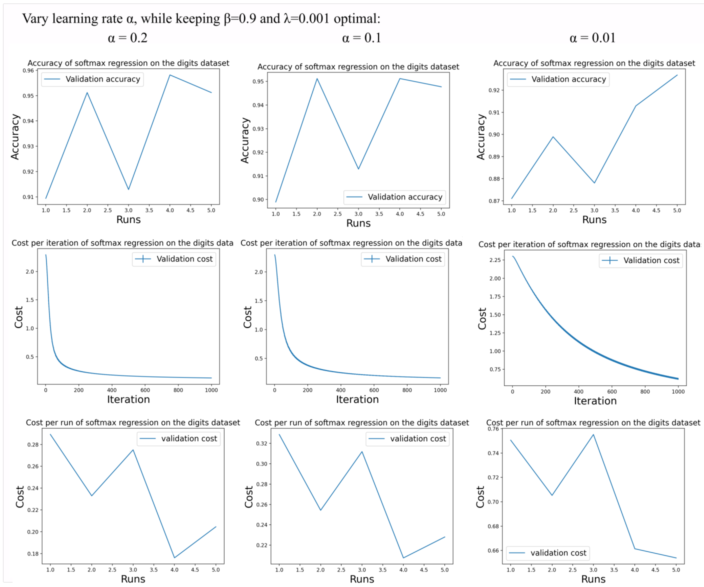

Figure 3: Effect of varying the **learning rate** α on the validation accuracy (first row), convergence of the validation cost (second row) and final validation cost (third row) of the 5-fold cross-validation softmax regression model on the digits dataset. (The optimal combination of hyperparameters is displayed in the first column).

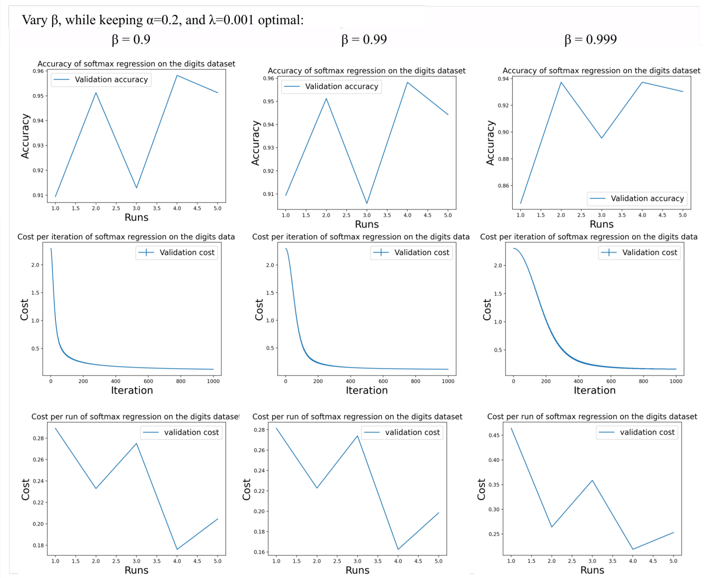

Figure 4: Effect of varying the **momentum** β on the validation accuracy (first row), convergence of the validation cost (second row) and final validation cost (third row) of the 5-fold cross-validation softmax regression model on the digits dataset. (The optimal combination of hyperparameters is displayed in the first column).

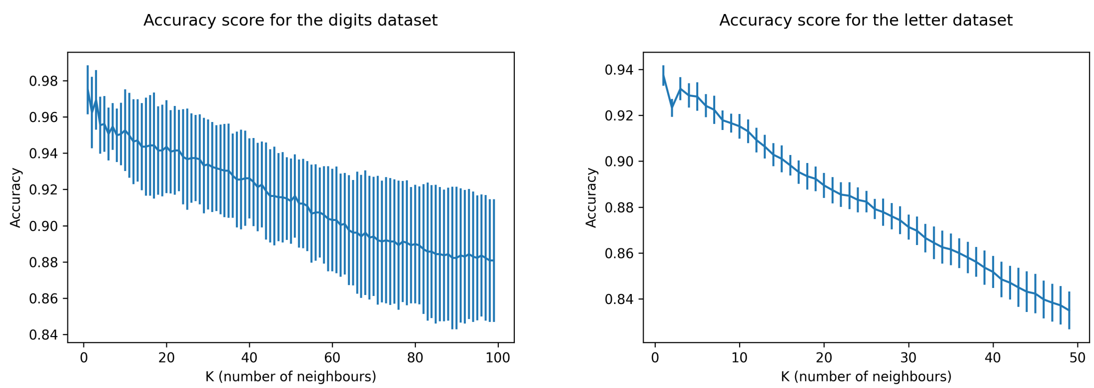

Figure 5: Accuracy score 0f 5-fold cross-validation KNN model when varying the number of neighbours K for the digits dataset (left) and letter dataset (right)

Our model performed better on the digits dataset with validation and test accuracy of 94% and 90%, respectively. However, the optimal KNN model performed significantly better on the letter dataset with validation and test accuracy of 94%. In contrast, our model achieved validation and test accuracy of 71% on the letter dataset.

4 Discussion and Conclusion

In summary, our implementation of the multi-class logistic regression model (softmax regression model) with mini-batch gradient descent, momentum and L_2 regularization with 5-fold cross-validation had a slightly lower test and validation accuracy on the Scikit-Learn digits dataset than the KNN model with $k = 1$ imported from the Scikit-Learn package. However, the KNN model with $k = 1$ from the Scikit-Learn package achieved a better test and validation accuracy on the letter dataset from Openml than our softmax regression model. Having had more time, we would have tried more implementation of the gradient descent optimizer, like Adaptive Gradient (Adagrad) or Adaptive moment estimation (Adam). We would also have conducted a more extensive grid-search, as it took a lot of running time.

5 Statement of Contributions

Krystal contributed to data exploration, the accuracy score set up and the abstract and datasets section of this report. Corinne worked on the introduction, part of the results section and conclusion, as well as the gradient descent optimizer and softmax regression model. Nghi contributed a part of the abstract and the hyperparameter tuning and termination condition section.