COMP551 Mini Project 2

Grace Hu 260776936
Xingyu Chen 260786048
Jiahui Peng 260782511

1 Abstract

This project compares the performance of two classifiers, softmax regression and K-Nearest-Neighbour(KNN), by evaluating them on 2 datasets. The Digits dataset from Scikit-Learn is made up of images of hand-written digits, and the MFeat-morphological dataset consists of handwritten numerals (0 - 9) extracted from a collection of Dutch utility maps. We used Gradient Descent to optimize the models and selected the best hyper-parameters via 5-fold cross-validation and GridSearch. Moreover, we explored how the running time and accuracy of the models would change as we tuned the hyper-parameters. Our initial results were very promising, showing that KNN Classifier has a slightly better performance than Softmax Regression. Using our KNN Classifier, we achieved an accuracy of 98.54% on the Digits dataset and an accuracy of 72.19% on the MFeat-morphological dataset.

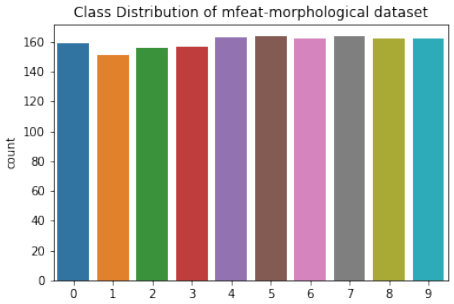
2 Introduction

In this project, we explored 2 datasets, Digits dataset from Scikit-Learn and MFeat-morphological from OpenML, and implemented multi-class logistic regression and KNN classifier on the datasets to investigate the performance of the two models. Moreover, we used Gradient Descent to optimize our models. Both of the models have many hyper-parameters, so we used 5-cross validation and grid search to find the hyper-parameter combination that gives the best performance.

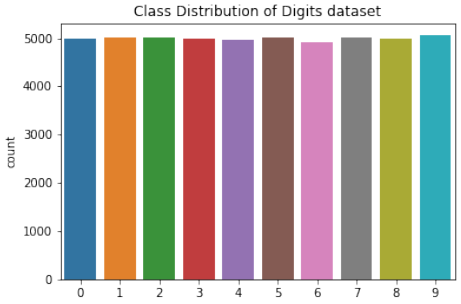
With our Softmax Regression model, we achieved an accuracy of around 97% on the Digits dataset with batch size=140, learning rate=0.45, and momentum=0.85. On the MFeat-morphological dataset, we achieved an accuracy around 71.125% on with batch size=150, learning rate=0.45 and, momentum=0.86.

With our KNN classifier, we achieved an accuracy of around 98.54% on the Digits dataset with k=3. On mfeat-morphological dataset, we achieved an accuracy around 72.19% with k=9.

3 Datasets



(a) mfeat-morphological



(b) digits

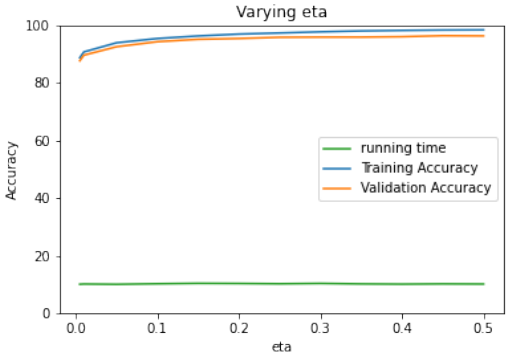
Figure 1: Class distribution

The Digits dataset is made up of 1797 8x8 images. Each image is of a hand-written digit[1]. The mfeat-morphological dataset consists of features of hand-written numerals (0-9) extracted from a collection of Dutch utility maps. 200 patterns per class (for a total of 2,000 patterns) have been digitized in binary images. These digits are represented in terms of the following six feature sets (files) [2]. We analyzed the class distribution of 2 datasets. In both datasets, each class has a similar size with the remained classes(Figure.5). Since the data does not follow a Gaussian distribution, we normalize 2 datasets to rescale them to the range [0,1]. We also reshaped the data from a 28x28 Numpy array to a 784x1 Numpy array to facilitate running the algorithm.

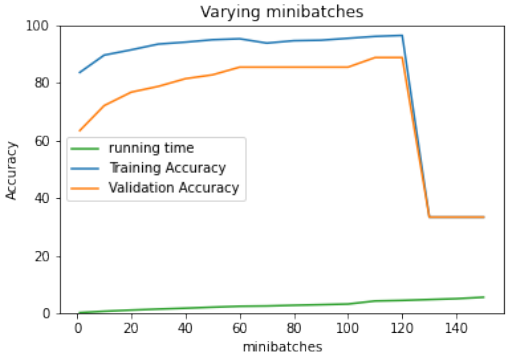
4 Results

4.1 Analysis and Early Termination

We analyzed for three datasets (OpenML MFeat-Morphological Dataset, Sklearn Digits, OpenML Iris Dataset) the effect on hyper-parameters (learning rate, momentum, and minibatch size) on convergence speed in Figures 2 and 3. Running time values were multiplied by 10 so they would be visible on the axis. As the number of minibatches increases, running time increases because there are more calculations needed per loop of Gradient Descent. Meanwhile, varying learning rate and momentum doesn't have a visible effect on running time.

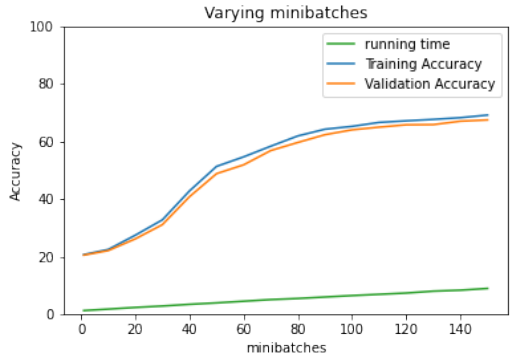


(a) Digits Dataset - Varying Learning Rate

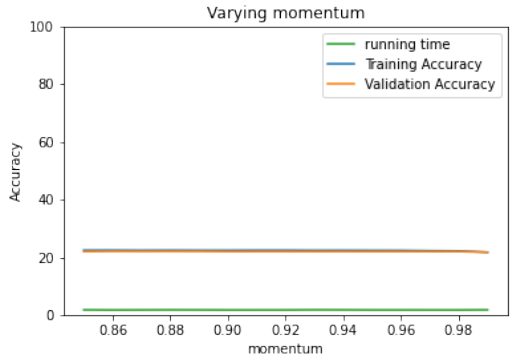


(b) Iris Dataset - Varying Minibatches

Figure 2: Effect of Varying Hyper-parameters on Running Time, Training Accuracy, Validation Accuracy



(a) MFeat-Morphological Dataset - Varying Minibatches



(b) MFeat-Morphological Dataset - Varying Momentum

Figure 3: Effect of Varying Hyper-parameters on Running Time, Training Accuracy, Validation Accuracy

	Digits	MFeat-Morphological	Iris
Learning Rate	0.45	0.45	0.45
Momentum	0.85	0.86	0.85
Minibatches	140	150	110
Max Accuracy	0.97	0.72125	0.90

Figure 4: Table of Optimal Hyper-parameters

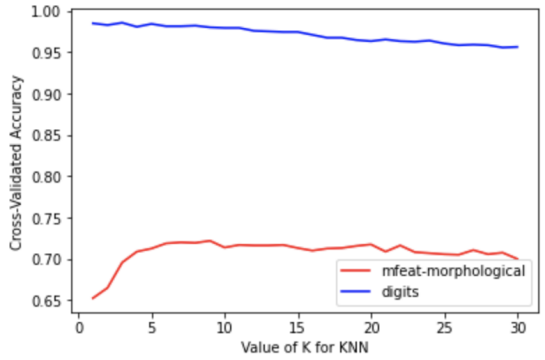
The optimal hyper-parameters we found through our implementation of GridSearch and 5-fold cross-validation that produced the highest validation accuracy were similar for all three datasets (Figure 4.1).

4.2 Comparison with KNN

For KNN classification, the hyper-parameter is the value of K. Figure 5 represents the accuracy of tuning the hyper-parameter by using the 5-fold cross-validation. We performed the analysis of 2 sets where the mfeat-morphological set is represented by the red line and the digit set is represented by the blue line. For the digit data set the best K value is 3 which yields an accuracy of 0.985, and mfeat-morphological

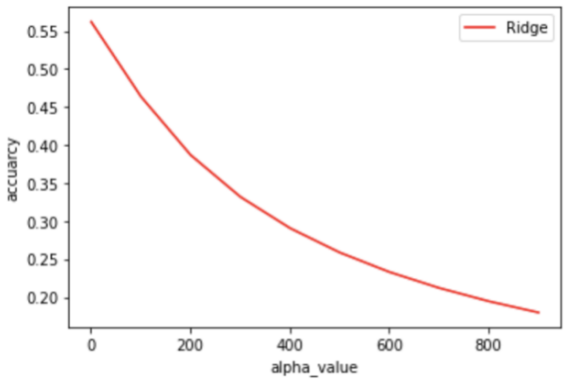
data set has the best K value of 9 with an accuracy of 0.722.

In comparison to logistic regression, the KNN has better accuracy for the digit data set. Logistic regression yields an accuracy of 0.970 with the best combination of its three hyper-parameters (ie. learning rate=0.45, momentum=0.85, and max accuracy= 140). The KNN generates a similar accuracy for the MFeat-Morphological data set. The logistic regression yields an accuracy of 0.721 with its best combination of three hyper-parameter (ie. learning rate=0.45, momentum=0.86, and max accuracy= 150).




```
best k value for mfeat-morphological 9
best accuracy for mfeat-morphological 0.721875
best k value for digits 3
best accuracy for digits 0.9853755323267517
```

(a) Digits Dataset



(b) MFEAT Dataset

Figure 5: Ridge Cross Validation

5 Creativity

As part of our creativity section, we used 5-fold cross-validation to tune the hyper-parameters of another classification model: Ridge Regression. The diagram (Figure 5.b) demonstrates that the hyper-parameter, alpha (regularization strength), yields the best accuracy when it equal to zero. This means that no regularization is needed to achieve high accuracy.

As an extension to our project, we also ran our Softmax Regression model on the OpenML Iris Dataset, and included its results in the figures above along with the MFEAT and Digits datasets as an additional comparison. The Iris dataset includes the petal and sepal lengths of three different types of iris flowers, which are used to predict which class each flower belongs to.

In terms of preprocessing for the Iris Dataset we used a LabelEncoder to map the string labels given in Iris.targets to integers. We also normalized all the data values by subtracting the data's mean from them and dividing by the data's standard deviation.

6 Conclusions

Minibatch size can have a visible effect on running time, but it also contributes to better training and validation accuracy. When choosing hyper-parameters for machine learning models in the future, it's important to find the right trade-off between accuracy and running time of the algorithm. We also discovered that the set of optimal hyper-parameters may differ between each dataset, meaning that the same model would need to be adjusted if it's used on a different dataset than the one it was trained on in order to maximize prediction accuracy.

References

[1] F. Pedregosa, G. Varoquaux, A. Gramfort, et al. "Scikit-learn: Machine Learning in Python". In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[2] Dheeru Dua and Casey Graff. *UCI Machine Learning Repository*. 2017. URL: `http://archive.ics.uci.edu/ml`.