Logistical Regression to Recognize Hand-Written Characters

Delaney Page, Ariel Polani, Jad Zeineddine, Daniel Bertak

***Abstract*—**

Handwritten digit recognition is both a classic and modern problem in Machine Learning literature. Classic in the sense that it is a famous problem often used to motivate courses in machine learning, and modern in the sense that progress is still being made in improving classification of handwritten digits. As such, it makes sense that there have been many different machine learning techniques applied to this problem over time. In this paper, we focus primarily on the use of logistic regression to solve the binary classification problem (a vs. b). Although we explored other methods for binary classification, we found that our logistic regression model performed the best when evaluated based on accuracy (X%). We discuss features of the training data that led to the superior performance of logistic regression. Finally, since logistic regression does not extend as naturally to the multivariate classification problem, we explore the use of Neural Network approaches among others to tackle the larger alphabet.

I. Introduction

Keeping consistency with in-class discussions about the curse of dimensionality and overfitting, a simple model for a simple problem was kept first in mind. Due to the relatively small size of the given dataset, the model would also have to efficiently maximize its use of the problem's characteristics. Some of these include the need for binary classification, sparsity (many more instances of "False" or "0" values in each observation matrix than "True" or "1" values), and shape.

The final model was chosen based on two factors: practicality and performance. We chose to submit the Logistic Regression Model instead of the SVM because we found it interesting that it performed as well as the SVM while requiring less computational power. The simplistic nature of the problem encouraged a simple solution. In the initial planning phases, our team cycled through the typical models covered in the class. We narrowed down our options by having an inner-project competition on who could come up with the best model. The winning design combined useful feature creation and a common classification model. Even when compared to probabilistic generative classifiers (PGC), neural networks (NN), and linear discriminant analysis (LDA), the logistic regression model remained dominant. The features found were found to fit best for this model: clearly separated with some overlap to prevent non-existent solutions to the maximum likelihood estimation of its coefficients [1]. In the next few paragraphs, we explain the logic behind our feature creation and define our model.

II. Implementations

Inspired by discussion in class about "projecting" data onto different planes using LDA and principal component analysis (PCA), we came up with ideas to exploit the differences in density of pixels along the x- and y-axes. In exploratory data analysis, we found that the x-axis coordinate with the highest density of pixels (the argmax of the count of pixels on a 1-dimensional projection onto the x-axis) had distinguished distributions for the different letters. In similar fashion, the difference between the range of the x-axis projection and the y-axis projection also had distinguished distributions. We found these differences by using histograms shown below.

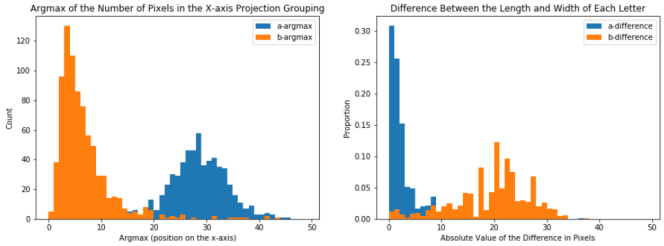

Figure 1: Histograms showing the number of pixels once projected onto the x-axis and the difference between the x-axis and y-axis projection.

It would later be discovered that around 99% of our model's accuracy would come from these two features. To create the final logistic regression model, the scikit-learn library was used to perform training and testing [2]. To determine prime performance, two versions of the logistic regression model regularization term were tested and then models using either only two or three features were tested on the best model variation that used all four features. To test accuracy, we cross-validated our model and then repeated that procedure 500 times, taking the average of the cross-validated accuracy each time to come up with a bootstrap-like technique for estimating our true accuracy. Stratified k-fold samples were taken to ensure balance in the training and testing sets over five folds and the model was trained and tested iterating over each fold.

Ultimately, we chose a logistic regression model with four features using L1 regularization and liblinear solver [3],[4]. The formal definition of this model is written below.

(equation 1 goes here)
Define coefficients

III. Experiments

Differences between the models were extremely small. Going from the L1 regularization to the L2 regularization produced a difference of around -0.0001 percentage points. Dropping one feature produced a difference of -0.005 accuracy percentage points and dropping two produced a difference of -0.003 accuracy percentage points from four features on the L1 regularization logistic regression model. The results of some of these tests are shown visually in the diagrams below.

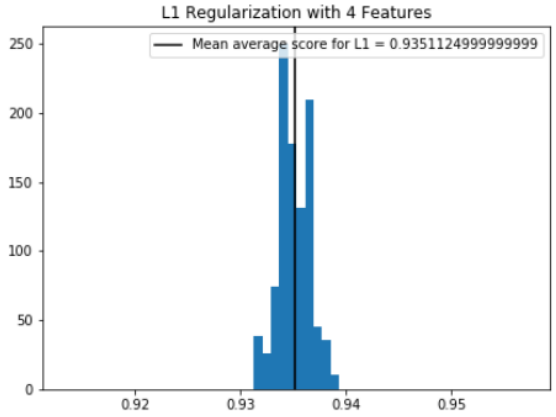

Figure 2: Accuracy score for Logistic Regression when using L1 Norm

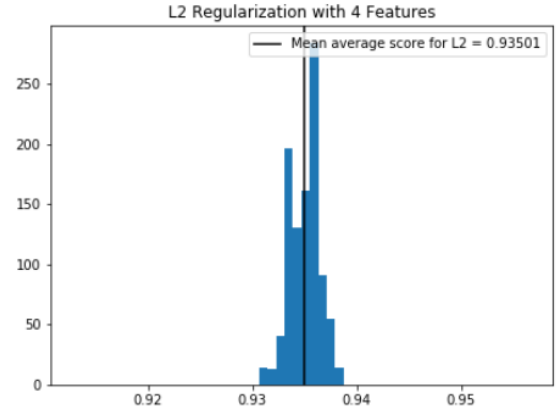

Figure 3: Accuracy score for Logistic Regression when using L1 Norm

Figure 4: Accuracy score for Logistic Regression as the number of features vary

IV. Conclusion

The task of correctly classifying images has been a popular topic in machine learning for the past few decades. More specifically, the classification of letters has been a crucial advancement for its impact on applications in industries like the mail delivery systems. In this paper, we presented an approach to classifying letters using a X-valued histogram into a logistic regression classifier. Our model outperformed the accuracy of some of the state-of-the-art methods like PCA into SVM. The high accuracy score discriminating based on the X-valued histogram empirically shows that there exists a discriminatory basis in the way most hand-written “a”s and “b”s are distributed on the X-axis.

Although this method is an improvement compared to some traditional state-of-the-art machine learning methods like PCA into SVM, it is still heavily outperformed by Convolutional Neural Networks and other Deep Learning developments [5].

V. References

[1] Mansournia, M., Geroldinger, A., Greenland, S. and Heinze, G. (2017). Separation in Logistic Regression: Causes, Consequences, and Control. *American Journal of Epidemiology*, [online] 187(4), pp.864-870. Available at: https://academic.oup.com/aje/article/187/4/864/4084405.

[2] Scikit-learn: Machine Learning in Python, Pedregosa *et al.*, JMLR 12, pp. 2825-2830, 2011.

[3] Scikit-learn.org. (2019). *1.1.11. Logistic regression — scikit-learn 0.21.3 documentation*. [online] Available at:

[4] https://scikitlearn.org/stable/modules/linear_model.html#logistic-regression [Accessed 1 Dec. 2019].

[5] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin. LIBLINEAR: A library for large linear classification *Journal of Machine Learning Research* 9(2008), 1871-1874

[6] Kowsari, K., Heidarysafa, M., E. Brown, D., Jafari Meimandi, K. and E. Barnes, L. (2018). *RMDL: Random Multimodel Deep Learning for Classification*.