Data Tidying and Reporting – Task 1

2024-02-18

Introduction

Handwritten digit recognition involves the conversion of human handwritten digit images into digital representations. This task might be challenging due to humans imperfections and the wide variability of handwritten forms. As well as some digits might look similar, increasing the difficulty of the task.

The dataset used for this project has been provided by the professor in the file named `qmnist_nist.RData`. It contains both the training and testing datasets, storing the images as a collections of pixel values. This report focuses on using ridge logistic regression to distinguish between handwritten digits, specifically 4 and 9, and lately this procedure has been extrapolated to the classification problems of one versus another digit.

Methodology

Ridge logistic regression is a regularization technique used for classification tasks, which involves adding a penalty term, $\lambda \sum_{j=1}^{n} \beta_j^2$, to the loss function. This term reduces model complexity and helps prevent overfitting. This technique is suited for high-dimensional data like images. Below, we detail the methodology applied in this project:

1. **Data preprocessing**. The initial step involves preparing both, training and testing sets. The data is filtered to include only the images representing the two digits of interest. Then it is formatted properly to ensure compatibility with the computational model.

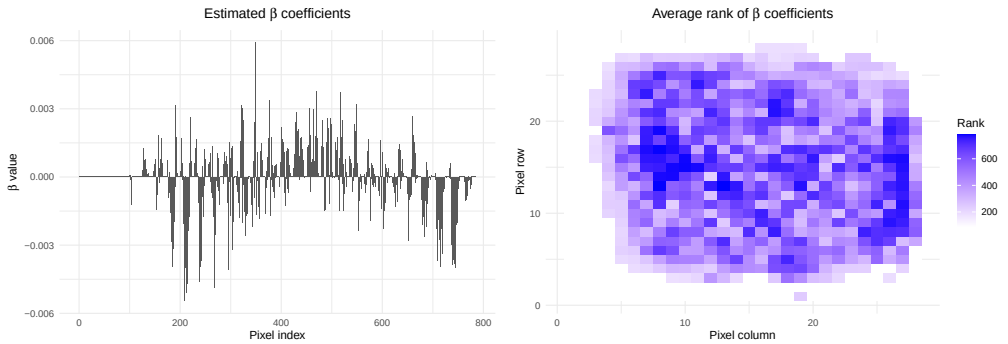
2. **Model training**. In this phase, we search for the optimal λ value which helps in making a good model (not too simple, not too complex). It is worth to mention that we do not standardize the predictors; this decision is based on the consistent scale and range of pixel values across the handwritten digit images.

3. **Model evaluation**. The performance of our trained model is assessed using a reserved test set. The evaluation metric is the accuracy, defined as the proportion of the number of correctly identified digits to the total number of digits in the test set.

4. **Model visualization**. Through two plots, we are able to understand the model's strategy in distinguishing between digits. The *Estimated β coefficient* shows the magnitude and direction of this parameter (β) over the pixel index (28x28=784 points). The higher the magnitude values, the greater their contribution is. Meanwhile, the *Average rank of β coefficients* uses a gradient from white (no relevant) to blue (high relevant) to highlights the importance of each pixel for the model's decision, shown on a 28x28 grid.

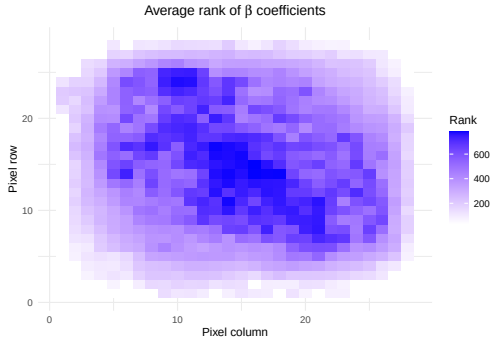
Results

4 vs 9 classifier. The model classifier showcased a remarkable ability to differentiate between digits 4 and 9, achieving an accuracy of 97.799%. Two critical regions are identified over the blue areas. The left region, in particular, captures the curve of the digit 9, contrasting with the peak characteristic of the digit 4.



General classifier. Classify one digit versus another involves the creation of 45 different models. This approach yielded an incredible average accuracy of 99.17%. Among these, the best performing was between digits 0 and 1, achieving an accuracy of 99.969%. On the other hand, even the worst model obtained a 97.181% of accuracy which was obtained by digits 3 and 8.

The visualization of pixel importance across all classifications showcases a pattern: pixels within blue regions are most located in the central area of the images, indicating their crucial role in the definition of the shape of the digits. Conversely, areas in white which hold less importance, are typically found at the edges and corners of the images.



Conclusion

The ridge logistic model has demonstrated an outstanding performance across all tasks, distinguishing greatly between pairs of digits. Therefore, it can be concluded that the chosen model is a great selection for this classification problem. Lastly, it is important to mention that this report focuses on presenting key insights and a comprehensive explanation, for those interested in the implementation details, the full code is available in the R Markdown script.

References

Bates, Douglas, Martin Maechler, and Mikael Jagan. 2023. *Matrix: Sparse and Dense Matrix Classes and Methods*. https://Matrix.R-forge.R-project.org.

Friedman, Jerome, Robert Tibshirani, and Trevor Hastie. 2010. "Regularization Paths for Generalized Linear Models via Coordinate Descent." *Journal of Statistical Software* 33 (1): 1–22. https://doi.org/10.18637/jss.v033.i01.

R Core Team. 2023. *R: A Language and Environment for Statistical Computing*. Vienna, Austria: R Foundation for Statistical Computing. https://www.R-project.org/.

Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag New York. https://ggplot2.tidyverse.org.