Classifying Numerical Values of Handwritten Digits

Ben Wright
Department of Mathematics
UW-Madison
Madison, WI 53703
bwright8@wisc.edu

Abstract

Greyscale data of 60,000 handwritten digits from the MNIST database were used to train classifiers to predict the numerical value of a handwritten digit from test data of 10,000 labeled digits. Classifiers were based on regression, support vector machines, and neural networks. Accuray of the classifiers ranged roughly between 80 to 98 percent.

1 The Data

The present project focused on the MNIST database of handwritten digits, which may be found at http://yann.lecun.com/exdb/mnist/. The data consists of a training set of 60,000 labeled, fixed-sized, greyscale images of handwritten digits and a test set of 10,000 handwritten digits with the same properties. The images are saved in a special file format called IDX, which can be converted to a NumPy array of the greyscale values of the pixels with the library idx2numpy found at https://pypi.org/project/idx2numpy/.

Further, the website hosting the data describes that the data from the trainnig set is comprised of 30,000 handwritten digits from employees of the Census Bureau and 30,000 digits recorded from high school students. The test data is also a fair mix of handwritten digits from these two categories. There are at least 250 different writers represented in the training data. Also, note some greyscale pixels appear due to anti-aliasing of the data.

2 The Algorithms Used

The datas are labelled according to the value of a digit the handwriter was instructed to write, making the dataset prime for analysis by supervised learning techniques. The question that was proposed for study was: Can the computer be trained to learn the (label) value of a handwritten digit from its IDX representation? The three algorithms that were used to approach this question were: 1. linear/ridge regression. 2. support vector machines. 3. neural networks. A brief overview of each mehtod is given in the following subsections.

2.1 Linear Regression

The idea of linear regression for binary classification is as follows: Given a set of training data X with labels y, interpret the features as inputs and the labels as output. Then, compute the hyperplane represented by w in the feature-label space that minimizes the sum of squares of the difference between the actual labels y and the value of the hyperplane evaluated at each point of training data. Namely, $||Xw-y||_2^2$ is minimized. Now, each side of the hyperplane in the feature space corresponds to a predicted label for any input data. The hope is that because the squared-error is minimized that there are few misclassifications.

Preprint. Under review.

However, the model is biased because it assumes that labels can be predicted by a linear function. One way to try to reduce the bias is to introduce a regularization parameter λ that tells the regresssion model to prefer w with smaller coefficients. Namely, instead of minimizing $||Xw - y||_2^2$, we seek to minimize $||Xw - y||_2^2 + \lambda||w||_2^2$. Both regularized (ridge) regression and standard linear regression have multiple algorithmic solutions, say, such as gradient descent, but they also have closed-form solutions, which were the solutions used in this project. Namely, for linear regression, the solution is $w = (X^T X)^{-1} X^T y$, and for ridge regression, we have $w = (X^T X + \lambda I)^{-1} X^T y$.

2.2 Support vectors

A support vector machine is similar to a regression model in the sense that we are looking for a hyperplane to separate data in the feature-label space, only now, we consider a different measure of inaccuracy of the model, that is, a different "loss" function. In what follows, the following loss function was used: $\max(0, 1 - y_i(w^T X_i))^2$ where y_i is 1 or -1 if the datapoint X_i respectively is or is not a certain digit, and so the expression the SVM algoritm tries to minimize is

$$\frac{1}{60000} \sum_{i=1}^{60000} \max(0, 1 - y_i(w^T X_i))^2 + \lambda||w||_2^2$$

where $\lambda > 0$ is some regularization parameter.

When the data is linearly seperable, w can be understood at the hyperplane that separates the data that maximizes the margin between the hyperplane and the data. When the data is not seperable, a hyperplane is penalized by the above loss function for making a misclassification.

This algorithm that was implemented in this project to build the SVM classifier was the sklearn.svm.LinearSVC method. Basically, the algorithm is iterative; it makes an initial guess for the classifier, and then updates the guess based on a (subgradient) descent-style computation on the loss function. The algorithm terminates when losses of sequential guesses are within a certain distance threshold of each other. Details can be found in `https://www.csie.ntu.edu.tw/~cjlin/papers/liblinear.pdf`.

2.3 Neural Networks

A neural network is an organized collection of nodes arranged layers, the first layer being the input layer that takes the data to be classified. Data from one layer is then weighted and passed into each node in the next layer, where a certain nonlinearity is applied. Eventually, the final (output) layer outputs the predicted class of the data.

The advantage of neural networks in a learning problem such as this one is that neural networks, given enough nodes on a bounded number of layers, can learn any function, meaning, they can have highly nonlinear decision boundaries.

For this project, the weights of the networks were trained by doing a stochastic-gradient-descent-type algorithm called "Adam." The essential difference between Adam and standard stochastic gradient descent through backpropogation - which updates weights in the network by utilizing the chain rule of calculus to compute the gradient of the loss function - is that Adam uses different learning rates for each weight in the network and updates the weights at each iteration of the descent based on how quickly the network is learning the classifier. See, for example, `https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/` for a further explaination and see also the sklearn documentation for multilayered neural networks at `https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html`.

3 Results

3.1 Linear Regression

A binary, linear classifier was trained on all training data for each digit between 0 and 9. A digit selector then takes an input data and checks which of the ten classifiers assigns the data the highest output norm and predicts the corresponding digit. This process is 82.38 percent accurate on the test data of 10,000 handwritten digits. One might wonder which digits are the hardest to distinguish, or if regularizing the digit classifiers improves performance. Well, it turns out the digit 5 was the easiest to classify with over 91 percent accuracy, and 1 was the hardest to classify with just over 88 percent accuracy.

3.1.1 Headings: third level

Third-level headings should be in 10-point type.

Paragraphs There is also a `\paragraph` command available, which sets the heading in bold, flush left, and inline with the text, with the heading followed by 1 em of space.

4 Citations, figures, tables, references

These instructions apply to everyone.

4.1 Citations within the text

The `natbib` package will be loaded for you by default. Citations may be author/year or numeric, as long as you maintain internal consistency. As to the format of the references themselves, any style is acceptable as long as it is used consistently.

The documentation for `natbib` may be found at

    http://mirrors.ctan.org/macros/latex/contrib/natbib/natnotes.pdf

Of note is the command `\citet`, which produces citations appropriate for use in inline text. For example,

```
\citet{hasselmo} investigated\dots
```

produces

> Hasselmo, et al. (1995) investigated...

If you wish to load the `natbib` package with options, you may add the following before loading the `neurips_2020` package:

```
\PassOptionsToPackage{options}{natbib}
```

If `natbib` clashes with another package you load, you can add the optional argument `nonatbib` when loading the style file:

```
\usepackage[nonatbib]{neurips_2020}
```

As submission is double blind, refer to your own published work in the third person. That is, use "In the previous work of Jones et al. [4]," not "In our previous work [4]." If you cite your other papers that are not widely available (e.g., a journal paper under review), use anonymous author names in the citation, e.g., an author of the form "A. Anonymous."

4.2 Footnotes

Footnotes should be used sparingly. If you do require a footnote, indicate footnotes with a number[1] in the text. Place the footnotes at the bottom of the page on which they appear. Precede the footnote with a horizontal rule of 2 inches (12 picas).

Note that footnotes are properly typeset *after* punctuation marks.[2]

Figure 1: Sample figure caption.

Table 1: Sample table title

Part		
Name	Description	Size (μm)
Dendrite	Input terminal	$\sim$100
Axon	Output terminal	$\sim$10
Soma	Cell body	up to 10^6

4.3 Figures

All artwork must be neat, clean, and legible. Lines should be dark enough for purposes of reproduction. The figure number and caption always appear after the figure. Place one line space before the figure caption and one line space after the figure. The figure caption should be lower case (except for first word and proper nouns); figures are numbered consecutively.

You may use color figures. However, it is best for the figure captions and the paper body to be legible if the paper is printed in either black/white or in color.

4.4 Tables

All tables must be centered, neat, clean and legible. The table number and title always appear before the table. See Table 1.

Place one line space before the table title, one line space after the table title, and one line space after the table. The table title must be lower case (except for first word and proper nouns); tables are numbered consecutively.

Note that publication-quality tables *do not contain vertical rules.* We strongly suggest the use of the `booktabs` package, which allows for typesetting high-quality, professional tables:

`https://www.ctan.org/pkg/booktabs`

This package was used to typeset Table 1.

5 Final instructions

Do not change any aspects of the formatting parameters in the style files. In particular, do not modify the width or length of the rectangle the text should fit into, and do not change font sizes (except perhaps in the **References** section; see below). Please note that pages should be numbered.

[1] Sample of the first footnote.
[2] As in this example.

6 Preparing PDF files

Please prepare submission files with paper size "US Letter," and not, for example, "A4."

Fonts were the main cause of problems in the past years. Your PDF file must only contain Type 1 or Embedded TrueType fonts. Here are a few instructions to achieve this.

- You should directly generate PDF files using `pdflatex`.
- You can check which fonts a PDF files uses. In Acrobat Reader, select the menu Files>Document Properties>Fonts and select Show All Fonts. You can also use the program `pdffonts` which comes with `xpdf` and is available out-of-the-box on most Linux machines.
- The IEEE has recommendations for generating PDF files whose fonts are also acceptable for NeurIPS. Please see `http://www.emfield.org/icuwb2010/downloads/IEEE-PDF-SpecV32.pdf`
- `xfig` "patterned" shapes are implemented with bitmap fonts. Use "solid" shapes instead.
- The `\bbold` package almost always uses bitmap fonts. You should use the equivalent AMS Fonts:

  ```
  \usepackage{amsfonts}
  ```

  followed by, e.g., `\mathbb{R}`, `\mathbb{N}`, or `\mathbb{C}` for $\mathbb{R}$, $\mathbb{N}$ or $\mathbb{C}$. You can also use the following workaround for reals, natural and complex:

  ```
  \newcommand{\RR}{I\!\!R} %real numbers
  \newcommand{\Nat}{I\!\!N} %natural numbers
  \newcommand{\CC}{I\!\!\!\!C} %complex numbers
  ```

  Note that `amsfonts` is automatically loaded by the `amssymb` package.

If your file contains type 3 fonts or non embedded TrueType fonts, we will ask you to fix it.

6.1 Margins in LaTeX

Most of the margin problems come from figures positioned by hand using `\special` or other commands. We suggest using the command `\includegraphics` from the `graphicx` package. Always specify the figure width as a multiple of the line width as in the example below:

```
\usepackage[pdftex]{graphicx} ...
\includegraphics[width=0.8\linewidth]{myfile.pdf}
```

See Section 4.4 in the graphics bundle documentation (`http://mirrors.ctan.org/macros/latex/required/graphics/grfguide.pdf`)

A number of width problems arise when LaTeX cannot properly hyphenate a line. Please give LaTeX hyphenation hints using the `\-` command when necessary.

Broader Impact

Authors are required to include a statement of the broader impact of their work, including its ethical aspects and future societal consequences. Authors should discuss both positive and negative outcomes, if any. For instance, authors should discuss a) who may benefit from this research, b) who may be put at disadvantage from this research, c) what are the consequences of failure of the system, and d) whether the task/method leverages biases in the data. If authors believe this is not applicable to them, authors can simply state this.

Use unnumbered first level headings for this section, which should go at the end of the paper. **Note that this section does not count towards the eight pages of content that are allowed.**

Acknowledgments and Disclosure of Funding

Use unnumbered first level headings for the acknowledgments. All acknowledgments go at the end of the paper before the list of references. Moreover, you are required to declare funding (financial activities supporting the submitted work) and competing interests (related financial activities outside the submitted work). More information about this disclosure can be found at: `https://neurips.cc/Conferences/2020/PaperInformation/FundingDisclosure`.

Do **not** include this section in the anonymized submission, only in the final paper. You can use the `ack` environment provided in the style file to autmoatically hide this section in the anonymized submission.

References

References follow the acknowledgments. Use unnumbered first-level heading for the references. Any choice of citation style is acceptable as long as you are consistent. It is permissible to reduce the font size to `small` (9 point) when listing the references. **Note that the Reference section does not count towards the eight pages of content that are allowed.**

[1] Alexander, J.A. & Mozer, M.C. (1995) Template-based algorithms for connectionist rule extraction. In G. Tesauro, D.S. Touretzky and T.K. Leen (eds.), *Advances in Neural Information Processing Systems 7*, pp. 609–616. Cambridge, MA: MIT Press.

[2] Bower, J.M. & Beeman, D. (1995) *The Book of GENESIS: Exploring Realistic Neural Models with the GEneral NEural SImulation System.* New York: TELOS/Springer–Verlag.

[3] Hasselmo, M.E., Schnell, E. & Barkai, E. (1995) Dynamics of learning and recall at excitatory recurrent synapses and cholinergic modulation in rat hippocampal region CA3. *Journal of Neuroscience* **15**(7):5249-5262.