Robert Krajancic
1666020

CMPUT 466 - Course Project

INTRODUCTION:

This machine learning task uses the Fashion-MNIST dataset. The task is framed as a classification problem, in which, given an image as input, the machine learning program should try to predict what category the image belongs to. Each image belongs to 1 of 10 classes, where the classes represent what type of clothing is in the image (t-shirt, dress, sneaker, etc.). This dataset was selected because of its similarities to the MNIST dataset, which is commonly used as a benchmark to validate machine learning algorithms. The fashion-MNIST dataset is considered more difficult and complex; thus it is a good studying tool, and it can be used as a benchmark to further validate machine learning algorithms (Xiao Han, Kashif Rasul, and Roland Vollgraf).

Three different machine learning algorithms will be used to observe their accuracy and relative-performance in this image classification problem. These algorithms are softmax regression, one-vs-rest logistic regression, and the gaussian naive Bayes classifier. In addition, the majority guess method will be used as a trivial baseline.

PROBLEM FORMULATION:

The dataset uses grayscale images of clothing articles – each measuring 28 by 28 pixels – as input, and it uses integers between 0 and 9 (which represent different clothing classes) as the output. The dataset provides 60,000 training samples and 10,000 testing samples. For the purposes of this machine learning project, the training samples are randomly divided into 50,000 training samples and 10,000 validation samples. Doing this allows for the implementation of the training-validation-test infrastructure.

The dataset was obtained from the zalandoresearch/fashion-mnist GitHub repository, and permission for use was granted under the terms of the provided MIT license.

APPROACHES AND BASELINES:

Baseline: a majority guess method is used as a baseline for this task. When working with a 10-category classification problem, the majority guess method yields 1/10 accuracy. This was confirmed by running a quick test. Thus, an accuracy score of 0.10 will be used. Machine learning algorithms that achieve an accuracy > 0.10 in testing will be considered better than the trivial baseline.

NOTE: A random guessing algorithm was considered as a baseline. When implemented (see random-guess.py), it tended to produce results between 0.09 and 0.11. This is quite similar to the majority guess method; therefore, the more consistent majority guess baseline is used for simplicity

Softmax Regression and One-vs-Rest Logistic Regression:

Both of these algorithms use gradient descent for training. A limit of 50 epoch was chosen to achieve a reasonable runtime. Both algorithms treated the learning rate (alpha) as a hyperparameter. Both of the algorithms used gradient descent for training, so the choice of alpha has an impact on the final result. If alpha is too small, then it might not reach the best W matrix in time. Meanwhile, if alpha is too big, it may overshoot during updates, causing the algorithm to miss the best W matrix.

For hyperparameter tuning, the following alpha values were considered: 0.3, 0.1, 0.03, 0.01, 0.003, 0.001, 0.0003, and 0.0001. For each of these alpha values, the algorithm would train W using that alpha value and then perform a test on the validation data. The W matrix that performed the best on the validation data is selected as the final W matrix. Then, that W matrix is used to make predictions on the testing data.

Gaussian Naive Bayes Classifier:

This method does not use a gradient descent approach, so alpha is not an appropriate hyperparameter. Instead, the prior probabilities of the classes are treated as hyperparameters. Two sets of priors were considered. The first is set to the priors calculated based on the provided training data. The second is based on the knowledge that all classes are equally likely to occur, and so the priors for all classes are set to 0.10. Both sets of priors are tested on the validation data, and the set of priors that produces the best results is selected. Then, these priors are used to make prediction on the testing data

EVALUATION METRIC:

All three algorithms are evaluated using the 10,000 testing samples. The predictions for classes made by the algorithm are then compared to the actual classes of the testing samples. The **accuracy** of the predictions is used as a measure of success. Accuracy is simply the number of correct predictions divided by the total number of test samples. The real goal of the task is to correctly classify the images, so the accuracy of predictions is a suitable evaluation metric.

RESULTS:

Note: softmax.py, logistic-regrsession.py, and naive-bayes.py may produce slightly varying results due to random shuffling of the training and validation data and the initial weights being randomly initialized. For reference, the CourseProject.zip file contains a PDF that reports the results obtained from running each of the algorithms three times.

Softmax Regression:

Result: accuracy of 0.82

Alpha selected: 0.3

Softmax regression performed significantly better than the baseline of 0.10. It was able to correctly label 82% of the 10,000 testing images.

One-vs-Rest Logistic Regression:

Result: accuracy of 0.83

Alpha selected: 0.0003

OvR logistic regression performed significantly better than the baseline of 0.10. It was able to correctly label 83% of the 10,000 testing images

Gaussian Naive Bayes Classifier:

Result: accuracy of 0.69

Priors selected: using equal prior for all categories

The naive Bayes classifier performed better than the baseline of 0.10. It was able to correctly label 69% of the 10,000 testing images

CONCLUSION:

All three machine learning algorithms performed better than the baseline. This demonstrates the power of (supervised) machine learning and its ability to use data samples to learn the correlation between input and output.

Of the three algorithms, the naive Bayes classifier was the weakest. This method assumes that each feature is independent, and that the features follow a Gaussian distribution for each class. These are strong assumptions, and they do not strictly hold for the dataset. This results in the naive Bayes classifier having weaker performance.

Softmax regression and one-vs-all logistic regression both have similar performances. One-vs-all logistic regression tends to have slightly higher accuracy; however, more research would need to be done to confirm that it is the better algorithm for this problem. Nevertheless, both algorithms have an accuracy above 0.80 which is quite impressive. For the low stakes problem of categorizing clothing items, these results are acceptable.

Thus, this project shows the potential machine learning has in image classification problems, and it provides a strong classifier for categorizing images in the Fashion-MNIST dataset.

Robert Krajancic
1666020

Works Cited

Xiao, Han, Kashif Rasul, and Roland Vollgraf. "Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms." ArXiv, 28 Aug. 2017, arXiv:cs.LG/1708.07747.

Zalandoreaserch. “fashion-mnist.” GitHub, 11 Dec, 2023, https://github.com/zalandoresearch/fashion-mnist