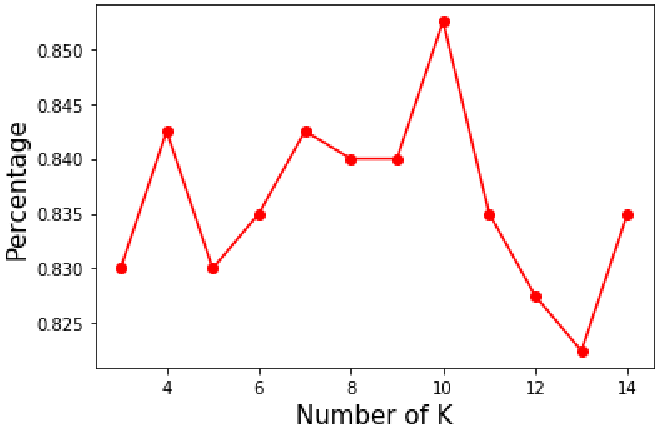
CS 532 Project Update

Name: Haotian Shi

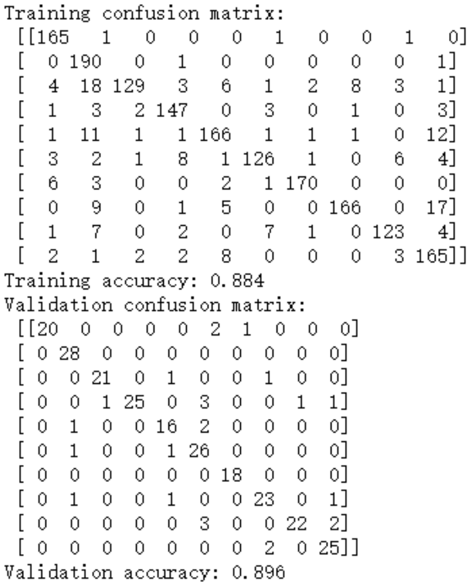
1. Current Progress

For the original dataset (MINIST), the dimension of the training data 'X' is 60000 x 784, where each sample corresponds to a label from numbers 0-9. The dimension for the test data 'X' is 10000 x 784. Due to the huge amount of data which leads to demanding computation, the first 2000 samples of the original dataset are used as the training set in this project, and the first 400 samples of the test set are used as the testing set in this project. The dataset was normalized after loading process to improve the training performance.

Currently, I have implemented KNN algorithm and Logistic algorithm for my experiment. For KNN algorithm, three works have been done. Firstly, I did a basic case experiment, which directly uses the KNN algorithm with 10 clusters (0-9 numbers) for classification and use the model to evaluate the test dataset. The testing accuracy is 85.25%. Then, I did an experiment which tunes the number of clusters K. I chose the K ranging from 3 to 14. The results show that when K = 10, the testing accuracy reaches maximum 85.25%, which shows below.



Thirdly, I did cross-validation in KNN algorithm based on 10 number of clusters. The training dataset was split into 8 folds. The 7 folds of data was used for training, and the rest dataset was used for validation. After 8 iterative training process, the model that shows highest validation accuracy was used for testing process. I also printed the training confusion matrix and validation confusion matrix for the best KNN model, which is shown below:



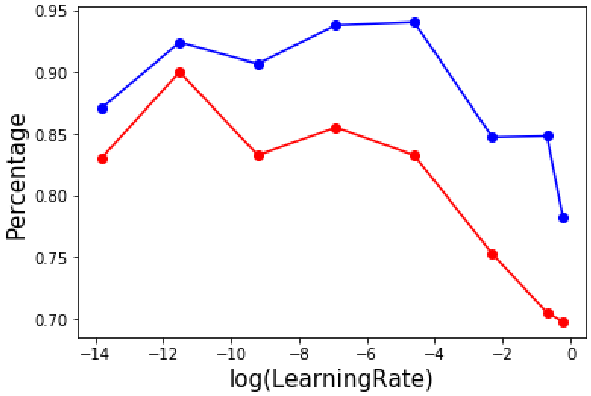
```
Training confusion matrix:
 [[165   1   0   0   0   1   0   0   1   0]
 [  0 190   0   1   0   0   0   0   0   1]
 [  4  18 129   3   6   1   2   8   3   1]
 [  1   3   2 147   0   3   0   1   0   3]
 [  1  11   1   1 166   1   1   1   0  12]
 [  3   2   1   8   1 126   1   0   6   4]
 [  6   3   0   0   2   1 170   0   0   0]
 [  0   9   0   1   5   0   0 166   0  17]
 [  1   7   0   2   0   7   1   0 123   4]
 [  2   1   2   2   8   0   0   0   3 165]]
Training accuracy: 0.884
Validation confusion matrix:
 [[20  0  0  0  0  2  1  0  0  0]
 [ 0 28  0  0  0  0  0  0  0  0]
 [ 0  0 21  0  1  0  0  1  0  0]
 [ 0  0  1 25  0  3  0  0  1  1]
 [ 0  1  0  0 16  2  0  0  0  0]
 [ 0  1  0  0  1 26  0  0  0  0]
 [ 0  0  0  0  0  0 18  0  0  0]
 [ 0  1  0  0  1  0  0 23  0  1]
 [ 0  0  0  0  0  3  0  0 22  2]
 [ 0  0  0  0  0  0  0  2  0 25]]
Validation accuracy: 0.896
```

The accuracy of the best knn model after cross-validation is 0.8575. Although the highest validation accuracy has been improved to 0.896. The accuracy for the test dataset still remain 0.8575.

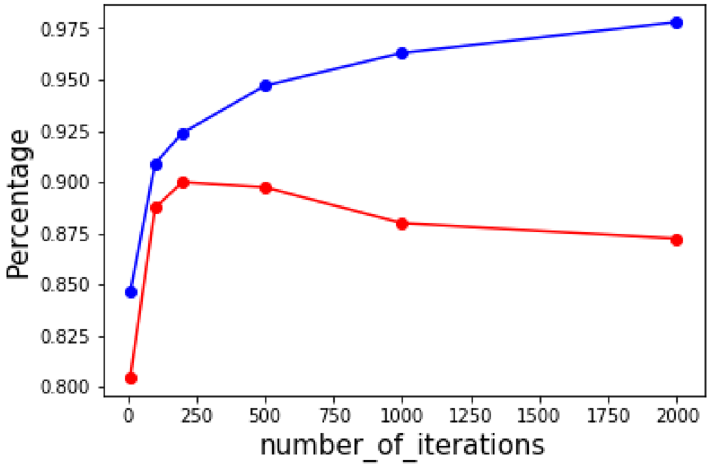
For the logistic algorithm, three works have been done. The task is decomposed into 10 binary classification tasks with 10 classifiers. Each classifier predicts a confidence level for the target number. For each sample, the category with the highest confidence is the category that the sample belongs to.

Firstly, I did a basic case experiment, which directly uses the logistic algorithm with 0.5 learning rate and 200 number of iterations for gradient descent. During the training process, each classifier shows the training accuracy and testing accuracy, which demonstrates the performance of logistic regression algorithm on classifying different numbers. The 1th classifier shows highest test accuracy, which reaches 98.75%. This is because the features of number 1 are easiest to distinguish from other numbers. The training accuracy for the entire dataset is 84.8%, while the testing accuracy is only 70.5%.

Then, I did an experiment which tunes the number of learning rate. The learning rate ranges from 0.00001 to 0.8. The results show that the best learning rate is 0.00001, which leads to the highest testing accuracy 90%. The figure below shows the results.



Thirdly, I did an experiment which tunes the number of number of iterations for gradient descent based on the optimal learning rate. The number of iterations range from 10 to 2000. The results show that when the number of iterations is 200, the testing accuracy reaches the highest 0.9. Specifically, the training accuracy increases as number of iterations increase. However, as the number of iterations increase, the testing accuracy reaches the peak at first, and then decreases. This is because too many iterations can cause the overfitting problem, which reduce the generalization capability of the model. The figure below demonstrates the result.



2. Plan for the next step

Next, I will implement neuron networks for the experiment. If the progress goes well, SVM algorithm will also be implemented for the experiment. I am generally on track based on the timeline.

The github link below shows the details for the project.

Github link: https://github.com/HaotianShi/CS532

Reference: Project Proposal

Project Datasets

The data processed in this project is MNIST dataset, which is a widely used dataset for classification algorithm from National Institute of Standards and Technology (NIST). The source link is http://yann.lecun.com/exdb/mnist/. The training set consists of handwritten numbers from 250 different people, of which 50% are high school students and 50% are from the Census Bureau staff. The test set is also handwritten digital data in the same proportion. The training data set contains 60,000 samples, and the test data set contains 10,000 samples. Each picture in the MNIST data set consists of 28 x 28 pixels, and each pixel is represented by a gray value. The 28 x 28 pixels are expanded into a one-dimensional row vector with 784 values. These rows are the feature rows in the first array of an image. The second array (labels) contains the corresponding target variable, which is the class label of the handwritten number (integer 0-9).

Thus, the dimension of the training data 'X' is 60000 x 784, where each sample corresponds to a label from numbers 0-9. Similarly, the dimension for the test data 'X' is 10000 x 784. The classification problem is defined as: which number from 0 to 9 is most likely to be given a feature vector with 784 features.

Investigated Algorithms

The following three algorithms will be applied on the dataset, and corresponding parameters will be analyzed.

1. k-nearest neighbor
   parameters for experiment: number of neighbors K; weights (all points in each neighborhood are weighted equally or weight points by the inverse of their distance); algorithm (which algorithm is used to search for the nearest k points when building the KNN model), etc.

2. neural networks
   parameters for experiment: number of hidden layers, learning rate, momentum, activation function; optimization algorithm, regularization, etc.

3. SVM
   parameters for experiment: kernel, degree, gamma, decision function shape, etc.

Validation task

The dataset will be divided into multiple subsets for cross-validation, and the validation results are averaged over the rounds to better estimate the model's predictive performance.

Confusion matrix, precision and recall will be used to evaluate the model performance. The performance of each algorithm will be compared and the basic principle behind the algorithm will be analyzed to discuss the evaluation results.

Schedule

Week	**Tasks**
10/19-10/25	Understanding the basic principles of the three algorithms
10/26-11/01	Dataset processing, developing training environment
11/02-11/08	Learning algorithm investigation for k-nearest neighbor and neural networks
11/09-11/15	Learning algorithm design for SVM
11/16-11/22	Evaluation and validation, results discussion
11/23-11/29	Write final report
11/30-12/08	Revise final report

Github link: https://github.com/HaotianShi/CS532