ME 532 Project
Machine Learning Algorithms for Fashion MNIST Data

Table of Contents

INTRODUCTION 1
DATA VISUALIZATION 1
LOGISTIC REGRESSION 2
PCA 3
NEURAL NETWORK 4
SUMMARY 5

Github link: https://github.com/agiwal/ME532-Project.git

Introduction

The dataset used for this project is the **Fashion-MNIST** data from Zalando's article images. The data consists of image examples each of which is a 28x28 pixels greyscale image. Thus, there are total 784 pixels in an image. Each pixel holds a value from 0 to 255 denoting the brightness of the pixel. The location of each pixel in the image is defined by the equation $x=28i+j$, where the x^{th} pixel lies in the i^{th} row and j^{th} column of a 28x28 matrix. Each article can be associated to one of the 10 labels mentioned later. There are 60,000 examples in the training data and another 10,000 in the test data. The goal of this project is to use machine learning algorithms for accurate classification and analyses of dataset.

The labels are as followed:

0 - T-shirt/top
1 - Trouser
2 - Pullover
3 - Dress
4 - Coat
5 - Sandal
6 - Shirt
7 - Sneaker
8 - Bag
9 - Ankle booot

Data Visualization

The first step in analyzing this dataset was to visualize the training data. Here is a summary and figures describing the training data:

a) Data dimensions: 1000 rows and 785 columns
b) Data labels:

Fig 1: Example of various labels in the MNIST dataset

c) Here is how the training data looks like:

	label	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6	pixel7	pixel8	pixel9	...	pixel775	pixel776	pixel777	pixel778	pixel779	pixel780	pixel781	pixel782	pi
0	2	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	
1	9	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	
2	6	0	0	0	0	0	0	0	5	0	...	0	0	0	30	43	0	0	0	
3	0	0	0	0	1	2	0	0	0	0	...	3	0	0	0	0	1	0	0	
4	3	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	

5 rows × 785 columns

Fig 2: Brief outlook of training data

Github link: https://github.com/agiwal/ME532-Project.git

d) To understand the distribution and correlation between various input pixels as well as the labels, distribution plots and correlation heat maps were evaluated:

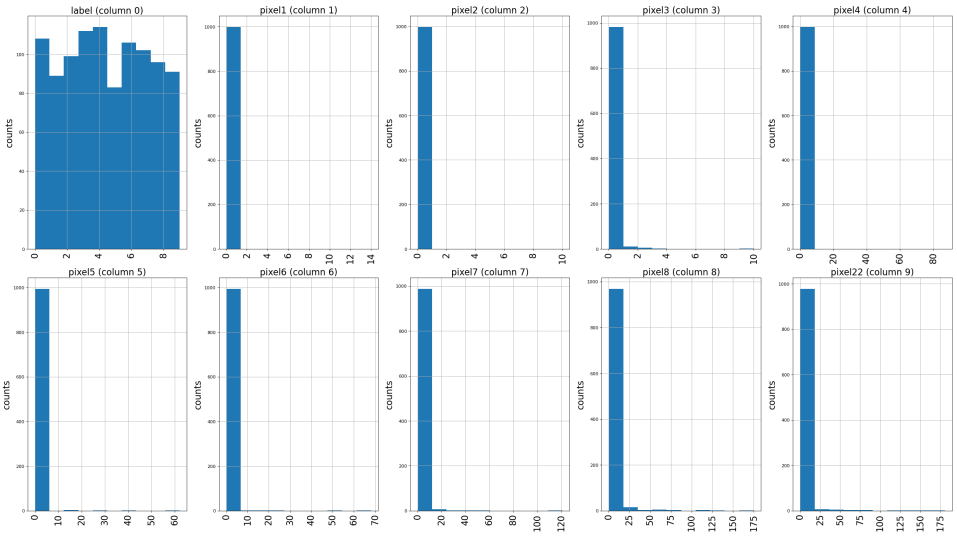

Fig 3: Distribution graphs

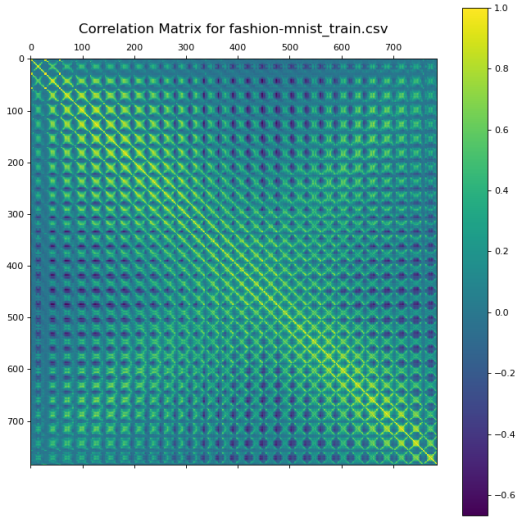

Fig 4: Pixel correlation matrix for training data

Similar distributions and correlations were seen for test data. From the data visualization and analyses three machine learning algorithms were chosen for this problem:

1. Logistic regression
2. Principal component analyses
3. Neural networks

Logistic regression

Since the dataset is a pixel-based image, logistic regression seemed a good start for classification. The algorithm to apply logistic regression is as follows:

a) Store pixels as input variable, x and labels as output variable, y
b) Normalize dataset by 255 to have data in range [0,1]
c) Using stochastic gradient descent to minimize error and optimize weights for each label
d) Using one vs all multiclass approach to generate new labels for each class

Github link: https://github.com/agiwal/ME532-Project.git

e) Using sigmoid function, the probability of each image with a certain weight to belong to each class
f) The highest probability is chosen as the assigned label for each image
g) The assigned label and actual label are compared error in regression algorithm was calculated.

Fig 5: Regression analyses accuracy as a function of iterations

The regression analyses gives a strong outlook of using multi-class methods for classification problems. However, due to poor correlation amongst input variables the model accuracy suffers.

PCA

The dataset is a 28x28 pixel image, PCA has been used to visualize if dimension reduction can be applied and the complexity of the dataset can be reduced. The preliminary results have been shown here. For instance, with a 6 component PCA analyses on 4 random objects is as follows, the variance from original images has also been calculated and mentioned.

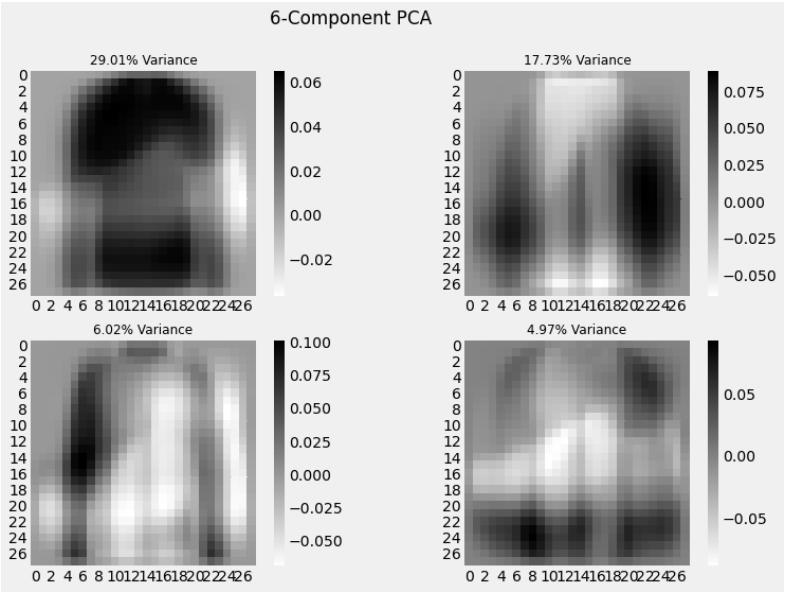

Fig 6: 6- Component PCA layout

It can be seen that a shoe can be identified in these images and thus the classification can be made to a certain assurance. To find out a good cut-off for number of components for PCA analyses the variance was plotted vs components. As can be seen through this plot a 2-component PCA analyses can be used as a cut-off since the variance flattens beyond that.

Github link: https://github.com/agiwal/ME532-Project.git

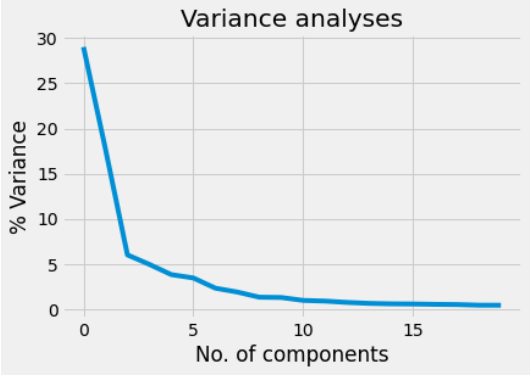

Fig 7: Variance analyses for PCA

Neural network

A fully connected neural network has been designed that performs classification over a given number of classes, hidden layer dimensions and input dimensions taken from the training dataset. A ReLU function has been used across the layers. The steps are as follows:

a) The model is initialized with weights for both layers along with biases
b) Using backpropagation loss has been calculated with a regularization variable in place
c) The neural network is trained using stochastic descent
d) Using the trained weights, the two layer neural network is used to predict the scores for an image across all classes
e) The class corresponding to the highest score is assigned to the image
f) Loss is calculated across multiple iterations to identify model accuracy

Below are the results of loss vs number of iterations for multiple hidden layer combinations. Fig 8 shows that the accuracy of neural network saturates after 10 hidden layers. Thus, for further analyses a 10 hidden layer neural network was chosen. Within this network the loss variation with iterations was plotted to evaluate the optimized iteration number. From Fig 9 we can see that the loss saturates beyond 150 iterations of the algorithm.

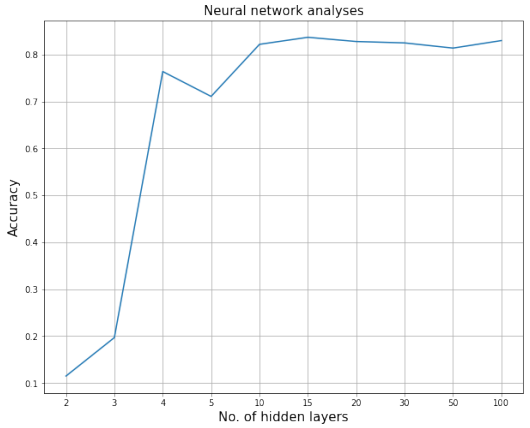

Fig 8: Accuracy of neural network with iterations

Github link: https://github.com/agiwal/ME532-Project.git

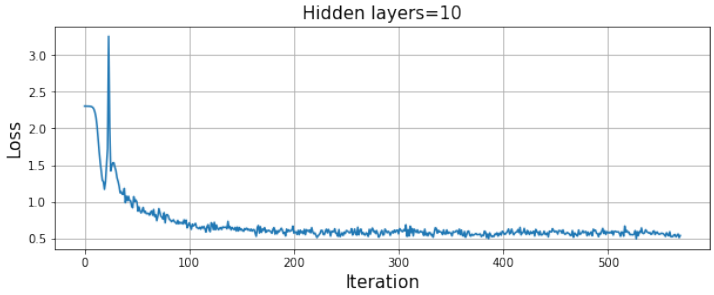

Fig 9: Loss variation for a 10 layer network

Summary

The fashion MNIST dataset has been used in this study to for implementation of machine learning algorithms for data evaluation and classification. The results of the analyses are summarized below:

- Data analyses and visualization showed that the pixels are poorly correlated and are evenly distributed across all images.
- Logistic regression analysis was performed on the dataset using stochastic gradient loss to optimize weights. An accuracy of 0.8 (80%) was achieved beyond 50000 iterations of the algorithm.
- PCA was performed to explore dimensionality reduction for the dataset. 2-component PCA was chosen as the cut-off using variance values.
- Neural networks were built for classification. The accuracy of algorithm saturated beyond 10 hidden layers and loss saturated at 150 iterations.

Github link: https://github.com/agiwal/ME532-Project.git