Walking through the projects

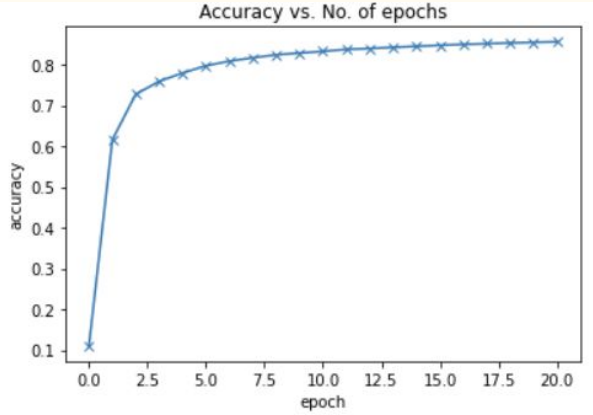
Handwritten digit recognition using Logistic regression, Feed Forward Neural Network

Smita Bhattacharya
M.Sc. in Data Science & Artificial Intelligence
Saarland University

Handwritten digit recognition

- **Goal:** Handwritten digit recognition using a **Logistic Regression Model**
- **Dataset:** MNIST - an inbuilt torchvision Image Classification Dataset
- **Labels:** 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- **Model Architecture:**
  - **Logistic Regression model** containing one fully connected linear layer
  - **Input:** Each training example is a vector, each 1x28x28 image tensor is *flattened* into a vector of size 784 (28*28) before being passed into the model
  - **Output:**The output for each image is a vector of size 10, with each element signifying the probability of a particular target label (i.e., 0 to 9). The predicted label for an image is simply the one with the highest probability. The **softmax function** is used.

Handwritten digit recognition(Cont..)



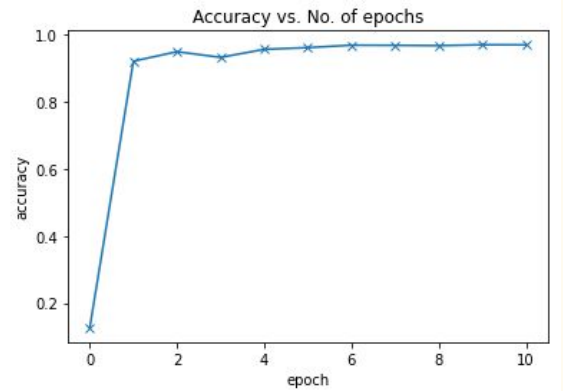
- Training and evaluation
  - Optimizer : Stochastic Gradient Descent
  - Loss Function: Cross Entropy
  - Accuracy: Computed as percentage of correct predictions

- Results and conclusions:
  - Test accuracy: At the end of 20 epochs is approximately 83%
  - Validation accuracy: Improved with each epoch
  - Conclusion: Linear model does not take into consideration the non linear relationship which can be there in the data

Handwritten digit recognition(Cont..)

- **Model Architecture:**
  - Fully connected Neural Network with following layers:
    - An Input layer
    - A hidden layer
    - An output layer.
  - **Input:** Each training example is a vector, each 1x28x28 image tensor is *flattened* into a vector of size 784 (28*28) before being passed into the model
  - **Output:**The output for each image is a vector of size 10, with each element signifying the probability of a particular target label (i.e., 0 to 9). The predicted label for an image is simply the one with the highest probability.

Handwritten digit recognition(Cont..)



- Training and evaluation
  - Optimizer : Gradient Descent
  - Loss Function: Cross Entropy
  - Accuracy: Computed as percentage of correct predictions

- Results and conclusions:
  - Test accuracy: At the end of 10 epochs is approximately 92%
  - Validation accuracy: Improved with each epoch
  - Conclusion: FFNN model does not improve beyond 92 %, we can add more hidden layers to learn more complex representation of the images, adding activation functions