Machine Learning

Homework 5

Faculty of Interdisciplinary Sciences & Technologies

University Of Tehran

Winter 2023

1) Answer the following questions in detail and include your responses in your final report: (point : 40%)

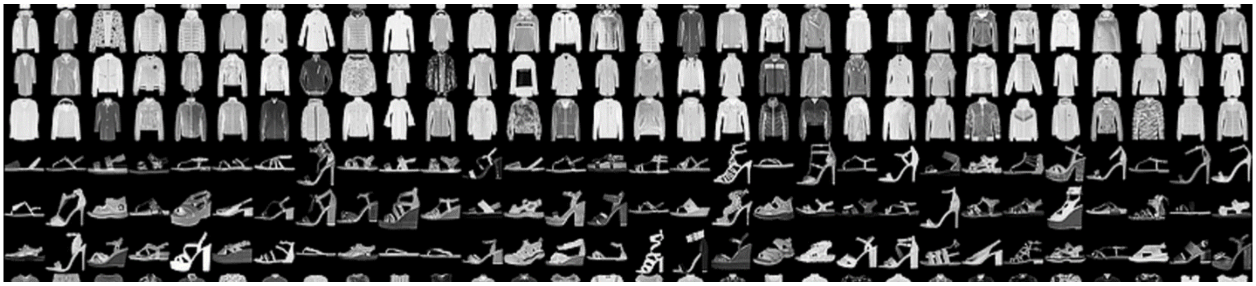
1. Describe vanishing and exploding gradients in neural networks. What is the solution to this problem? (point: 10%)

2. What is batch normalization? (point: 15%)

3. What is dropout, and what is its benefit? (point: 15%)

2) Train with RESNET-50 (CNN based model) (point: 60% + 10%)

1. Download Fashion MNIST data set. (point: 10%)



Content

Each image is 28 pixels in height and 28 pixels in width, for a total of 784 pixels in total. Each pixel has a single pixel-value associated with it, indicating the lightness or darkness of that pixel, with higher numbers meaning darker. This pixel-value is an integer between 0 and 255. The training and test data sets have 785 columns. The first column consists of the class labels (see above), and represents the article of clothing. The rest of the columns contain the pixel-values of the associated image.

To locate a pixel on the image, suppose that we have decomposed x as x = i * 28 + j, where i and j are integers between 0 and 27. The pixel is located on row i and column j of a 28 x 28 matrix.
For example, pixel31 indicates the pixel that is in the fourth column from the left, and the second row from the top, as in the ascii-diagram below.

Labels

Each training and test example is assigned to one of the following labels:

0. T-shirt/top
1. Trouser
2. Pullover
3. Dress
4. Coat
5. Sandal
6. Shirt
7. Sneaker

8. Bag
9. Ankle boot

2. You are tasked with creating a CNN for image classification using above datasets. Follow these steps: (point: 15%)

1. **Data Loading:**
   - Import the necessary libraries for deep learning (PyTorch).
   - Load above datasets.
2. **Data Preprocessing:**
   - Normalize the pixel values of the images to a range between 0 and 1.
   - Split the dataset into training and testing sets.(80 percent train and 10 percent test and 10 percent validation)
3. **Model Architecture:**
   - Design a simple CNN architecture consisting of convolutional layers, pooling layers, and fully connected layers.
   - Specify the input shape based on the image dimensions and the number of classes for the output layer.
4. **Compile the Model:**
   - Choose an appropriate loss function for a classification task.
   - Select an optimizer and a metric for model evaluation.
5. **Model Training:**
   - Train the CNN on the training dataset.
   - Monitor the training process and make adjustments if necessary.
6. **Model Evaluation:**
   - Evaluate the trained model on the testing dataset.
   - Calculate and display accuracy and any other relevant metrics.
7. **Prediction:**
   - Choose a few images from the testing set.
   - Use the trained CNN to predict the classes of these images.
8. **Discussion:**
   - Discuss the performance of the model.
   - Mention any challenges faced during training and potential improvements.

3. (Multi-Layer Perceptron) for image classification. Follow these steps: (point: 25% + 10%)

1. **Data Loading:**
   - Import the necessary libraries for deep learning.
   - Load above datasets.
2. **Data Preprocessing:**
   - Flatten the 2D images into 1D arrays.
   - Normalize the pixel values to a range between 0 and 1.

- Split the dataset into training and testing sets.

3. **Model Architecture:**
   - Design an MLP architecture with multiple hidden layers.
   - Experiment with different activation functions for hidden layers.
   - Choose an appropriate number of neurons for each layer.
   - Specify the input size based on the flattened image dimensions and the number of classes for the output layer.
4. **Compile the Model:**
   - Select a suitable loss function for a classification task.
   - Choose an optimizer and a metric for model evaluation.
5. **Model Training:**
   - Train the MLP on the training dataset.
   - Experiment with different training epochs and batch sizes.
   - Monitor the training process and adjust hyperparameters if needed.
6. **Model Evaluation:**
   - Evaluate the trained MLP on the testing dataset.
   - Calculate and display accuracy and any other relevant metrics.
7. **Prediction:**
   - Choose a few images from the testing set.
   - Use the trained MLP to predict the classes of these images.
8. **Discussion:**
   - Compare the performance of the MLP with the previously implemented CNN.
   - Discuss the challenges faced and potential improvements.

4. You have been given a dataset containing images.Perform a comprehensive EDA to gain insights into the dataset before building the MLP and CNN models. Consider the following tasks: (point: 10%)

1. **Data Distribution:**
   - Visualize the distribution of classes in the dataset.
   - Examine whether the dataset is balanced or if there is a class imbalance issue.
2. **Image Statistics:**
   - Calculate and visualize the mean and standard deviation of pixel values across all images.
   - Analyze whether there are variations in brightness and contrast among different digits.
3. **Dimensionality Analysis:**
   - Explore the dimensions of the images and analyze whether there is a need for resizing or cropping.
   - Visualize a few sample images to understand the variability in writing styles.
4. **Correlation Analysis:**
   - Investigate the correlation between pixel values within the images.

- Consider creating a heatmap to visualize correlations.

5. **Noise and Artifacts:**
    - Check for any noise or artifacts in the images that might affect model performance.
    - Identify and visualize any anomalies or irregularities.
6. **Feature Engineering Possibilities:**
    - Explore potential additional features that can be extracted from the images.
    - Consider techniques such as edge detection or texture analysis.
7. **Data Augmentation:**
    - Explore the feasibility of data augmentation techniques to increase the diversity of the training set.
    - Visualize a few augmented images to understand the impact on image variations.
8. **Discuss Implications:**
    - Discuss how your findings from EDA may impact the design of the MLP model.
    - Consider any preprocessing steps that might be beneficial based on the EDA results.

3) Attention Metrix's (bounces point: 40%)

1. What is attention Matrix in transformer architecture.(theory) (point: 10%)

2. Implement simple attention scenario with image embedding of question 2 part 4 and explain why attention is useful. (point: 30%)

Good Luck!