Report for Lab 5

Task 1 – MPI version

- We followed the given instructions to build and run the code.
- Text 2...
- We distributed the weight vectors `W1`, `W2` and `W3` to allow the processes to work on a random portion of the samples on their own.
- Text 4...
- Every process computes the change of the weights, i.e, the variables `dWx`.
- Text 6...
- We used the given commands to figure out the number of total processes as well as the rank of the executing process.
- Text 8...
- We used the following strategy.
  - Broadcast the randomly initialized weights to all processes.

```
// Random initialization of the weights in root process
vector<float> W1;
vector<float> W2;
vector<float> W3;

if (process_id == 0) {
    W1 = random_vector(784 * 128);
    W2 = random_vector(128 * 64);
    W3 = random_vector(64 * 10);
} else {
    W1.reserve(784 * 128);
    W2.reserve(128 * 64);
    W3.reserve(64 * 10);
}

// Broadcast initial weights
MPI_Bcast((void *)W1.data(), 784 * 128, MPI_FLOAT, 0, MPI_COMM_WORLD);
MPI_Bcast((void *)W2.data(), 128 * 64, MPI_FLOAT, 0, MPI_COMM_WORLD);
MPI_Bcast((void *)W3.data(), 64 * 10, MPI_FLOAT, 0, MPI_COMM_WORLD);
```

  - When training the model, we use `MPI_Allreduce` in order to sum over all weight changes and the original weights. After the reduction, the new weights values `W1`, `W2` and `W3` are present in all processes.

```
// Training loop
for (unsigned i = 0; i < 1000; ++i) {

    ... Calculate weight changes dW1, dW2, dW3 ...

    // Allreduce the weight deltas to all processes
    const auto dW1_aggr = process_id == 0 ? W1 - lr * dW1 : -lr * dW1;
    const auto dW2_aggr = process_id == 0 ? W2 - lr * dW2 : -lr * dW2;
    const auto dW3_aggr = process_id == 0 ? W3 - lr * dW3 : -lr * dW3;

    MPI_Allreduce((const void *)dW1_aggr.data(), (void *)W1.data(),
                  784 * 128, MPI_FLOAT, MPI_SUM, MPI_COMM_WORLD);
    MPI_Allreduce((const void *)dW2_aggr.data(), (void *)W2.data(),
                  128 * 64, MPI_FLOAT, MPI_SUM, MPI_COMM_WORLD);
    MPI_Allreduce((const void *)dW3_aggr.data(), (void *)W3.data(),
                  64 * 10, MPI_FLOAT, MPI_SUM, MPI_COMM_WORLD);

    if ((process_id == 0) && (i + 1) % 100 == 0) {
        ... Logging ...
    }
}
```

- Text 10...

Task 2 – Parallel GEMM per process

Text...