Indian Institute of Technology, Kharagpur
Department of Computer Science and Engineering
Software Engineering (CS 20202), Spring 2024

Assignment 3 – C++ Programming

Total marks: 100

Grading guidelines:

1. *Zero marks for a submission if it does not pass the plagiarism test.*
2. *Break-up of Credits will be as follows:*
   (a) *Percentage of features implemented: 70%*
   (b) *Code understanding – code clarity, comments: 10%*
   (c) *Whether reasonably able to answer questions: 20%*

In this assignment, you are asked to implement a vector data processing library in C++ from ground up and use it to implement a nearest neighbour search algorithm. The key component of this library is an abstract data type called *DataVector*, which implements the mathematical notion of a vector. The *DataVector* class you will implement should have a dimension, which can be zero if no data is stored. Following is a partial prototype of the class:

```
class DataVector {
    vector<double> v;
    public:
    DataVector(int dimension=0);
    ~DataVector();
    DataVector(const DataVector& other);
    DataVector & operator=(const DataVector &other);
    void setDimension(int dimension=0);
    DataVector operator+(const DataVector &other);
    DataVector operator-(const DataVector &other);
    double operator*(const DataVector &other);
}
```

You have to implement the following functionalities:

1. Constructor, destructor and *setDimension* function which also removes the existing data and creates a new data with the provided dimension. [**10 marks**]
2. Copy constructor and copy assignment operator with the usual functionality. [**10 marks**]
3. Implement the operators + and – for vector addition and substraction. [**10 marks**]
4. Operator * for computing the dot product between two vectors. Using this operator implement the `norm` and `dist` member functions which calculate the length of a vector and the distance between two vectors. [**10 marks**]

Submit an implementation of the above library in C++. All class defintions should be in a header file named `DataVector.h`, and the library function definitions should be in a c++ file called `DataVector.cpp`. A comment in the beginning of the header file should clearly explain the

role of each class and function in program. **10% of the implementation marks** will be given based on this comment.

Nearest Neighbor Search

The above implemented library for vectors should be used to define a class for storing a dataset, called `VectorDataset`, and then used to implement a program for nearest neighbour search. Nearest neighbour search is a basic operation used in many Machine Learning and Information Retrieval applications. The problem of **approximate nearest neighbour search (ANN)** is: given a test vector (also called **datapoint**) *v* and a vector dataset *D*, quickly find other vectors *v' in D* which are closest to *v*. The following website provides benchmark datasets for evaluating ANN algorithms:
https://ann-benchmarks.com/index.html
You have to download the Fashion MNIST dataset from the following link:
https://github.com/zalandoresearch/fashion-mnist
Implement a simple nearest neighbour search which sequentially traverses a given dataset and finds the nearest neighbour vectors to a given test vector. The following functionalities should be implemented:

5. The `VectorDataset` class with constructors, destructors and data access functions. [**10 marks**]
6. ReadDataset member function which reads a dataset from a file downloaded from the above link and stores in a `VectorDataset` object. [**10 marks]**
7. An implementation of the `knearestneighbor` function, that takes as input a VectorDataset and a DataVector and returns a new VectorDataset with the top k nearest neighbors. This function can be called in a main function to calculate nearest neighbors for all DataVectors in a test dataset. [**10 marks**]

The above implemented library should be used to perform nearest neighbour search on the fashion-MNIST train dataset and the total time taken should be reported. The `VectorDataset` class and related functions should be implemented in `VectorDataset.h` and ANN functionality should be implemented in `nearestneighbor.cpp`.

Submit all the files to moodle.

Implementation note:
After downloading the hdf5 file from the above mentioned link, you can convert the train and test datasets to csv format for reading easily in C++. You can use the following python code for data conversion:

```
import h5py
import numpy as np
import pandas as pd

# Open the HDF5 file
```

```
with h5py.File('fmnist.hdf5', 'r') as hf:
    # Get the dataset
    Datasetnames=hf.keys()
    print(Datasetnames)

    # Get the dataset
    dataset = hf['test']

    # Convert the dataset to a NumPy array
    data = np.array(dataset)

    # Create a Pandas DataFrame from the NumPy array
    df = pd.DataFrame(data)

    # Save the DataFrame to a CSV file
    df.to_csv('fmnist-test.csv', index=False)
```