Assignment 1 Report

Statistical Library

Members:

Đinh Duy Kha 1411675

Nguyễn Xuân Danh 1652736

Cù Huy Chiến 1111111

## Introduction

### Tasks

This assignment asks our group to build a statistical library that support data type **char**, **int** and **double**, having the following functions

1. Calculates sum
2. Calculate mean
3. Calculate standard deviation
4. Calculate variance
5. Calculate Q1, median and Q3
6. Calculate mode
7. Calculate and visualize histogram
8. Calculate and visualize noise

### Challenges

Statistics and probabilities is a rather new topic for us because we have not taken the course on this subject. To implement some functions of this project, we had to research on fundamental knowledge of statistic and probabilities.

### Requirements

1. Build a user interface support loading dataset of 3 types: **char**, **int** and **double**.
2. Write functions that are required for the library.

## Tool used

### Code::Block

Code::Block is the basic IDE of taught in the course. We decided to use it in this project because it is simple and easy to use.

### Git and Github

Git is a version control system that is simple and fast. A version control system helps us synchronize our code across different computers. This helps us speed up the development process tremendously.

Github[[1]](#footnote-1) is a free-to-use Git repository hosting service. It is used by more than 31 million developers across many fields, with more than 100 million repositories hosted[[2]](#footnote-2).

The Github for our project is at: <https://github.com/dalo2903/statistical-lib>

## Implementation

### Classes

To increase the modularity of the code, we decided to split the program into 3 classes that do different tasks of the program. Those classes are:

#### statistical\_lib

This class contains the main operations to perform calculations on data. It supports up to 3 types of data: char, int and double by using the defined *data* union type. The class contains a vector of data as and contains an int variable *type* to store the current type of the dataset

The class contains the functions:

1. double sum() : Calculate sum of dataset.
2. double mean() : Calculate mean of dataset.
3. double standard\_deviation() : Calculate standard deviance of dataset.
4. double variance() : Calculate variance of dataset.
5. double Q1() : Calculate Q1 of dataset.
6. double median() : Calculate median of dataset.
7. double Q3() : Calculate Q3 of dataset.
8. void histogram() : Visualize histogram of dataset
9. void noise() : Visualize noise of dataset

#### data\_loader

This class implement methods to load data from files into memory. It mainly utilize **fopen()** and **fscanf()** functions of <*stdio.h*>. This class has the functions:

1. bool open\_file(string file\_name)

: Open a file using fopen(). Return true if success, false otherwise.

1. void load\_char\_data(string file\_name, vector<data> &d)

: Load data of type char from a file to a vector d.

1. void load\_int\_data(string file\_name, vector<data> &d)

: Load data of type int from a file to a vector d.

1. void load\_double\_data(string file\_name, vector<data> &d)

: Load data of type double from a file to a vector d.

#### menu\_printer

This class mainly contains functions to print out the user interface of the program. Those function are implemented mainly using *<iostream>* library for printing , <*iomanip*> and <*windows.h*> libraries for formatting. This class contains the following functions:

1. void print\_title() : Print the header of the program’s interface.
2. void print\_main\_menu\_no\_data() : Print main interface when there is no data loaded.
3. void print\_main\_menu() : Print main interface when there is no data loaded.
4. void print\_load\_data\_menu() : Print the interface of data loading process.

### Data types

#### union type

The *union* is used to store 3 type of data type: **int**, **char** and **double** inside a single variable.

### Data structures

#### std::vector

Vector is a type of data structure containing a list of data. Its implementation is similar to dynamic array in C.

#### std::map

Map stores element in a pair of key and value. It is useful when constructing a frequency table.

### Functions

#### Sum

This function is implemented iterating over the entire dataset and add them to a sum.

#### Mean

This function is implemented by calculating the sum, then divide it by the size of the dataset.

#### Standard deviation

Standard deviation is calculated the same way as variance, just without the last part with is taking square root.

#### Variance

Variance is calculated by iterating over the dataset, first calculating the difference of each element and the mean of the dataset, square it, then add it to a sum. The sum is then divided by the size of the dataset and square rooted, resulting in variance.

## Conclusion

## Teamwork activities

## References

<http://www.cplusplus.com/reference/>

1. <https://github.com> [↑](#footnote-ref-1)
2. <https://github.com/about> [↑](#footnote-ref-2)