Numerical methods in Biomedical Engineering

Tutorial II

August 22, 2019

1 Functions and Data Types in Matlab

1.1 Functions

A function is a block of code which is intended to be used multiple times. It specifies a set of input and output variables. A set operations is done on the specified inputs and is returned to the output variables.

function output variables = *function name*(input variables)
Set of operations on input variables.
Return Output variables
end

You can refer to an existing function using an @ symbol or a local function can also be created using @, which can only be used in the current script.

2 Data Types

There are many data types present in Matlab. The ones which are frequently used are Numeric arrays, characters and strings, tables, structures, and cell arrays; data type conversion. We will look at tables, structures and cell data types, as these are most often used to store data efficiently.

2.1 Tables

Tables consists of column oriented data storage spreadsheet. Each column or row can be accessed by using the '.' operator. Data from a table can be retrieved by using () or {} brackets. () brackets are used to retrieve the data in table data type whereas {} brackets are used to access the data in its original data type. The original data type may be in double, string or character datatype. Every element in the table can be accessed using index notation or using row names or column names.

2.2 Structures

A structure array is a data type that groups related data using data containers called fields. Each field can contain any type of data. Access data in a field using dot notation of the form *structName.fieldName*.

2.3 Cells

A cell array is a data type with indexed data containers called cells, where each cell can contain any type of data. Cell arrays commonly contain either lists of text, combinations of text and numbers, or numeric arrays of different sizes. Refer to sets of cells by enclosing indices in smooth parentheses, (). Access the contents of cells by indexing with curly braces, {}.