1.3   MATLAB: Variables and elementary operations

MATLAB is case-sensitive, so `a` and `A` denote different objects. Variable names must begin with a letter and may contain digits and underscores; only the first 31 characters are relevant.

In MATLAB, variables are not formally declared. The default variable type is the *array*—of whatever kind of data (integers, complex numbers, strings, . . . ; kind and dimension of an array are set automatically and depend on the data attributed to it[4]). In accordance with that, by default, the mathematical operations refer to arrays, i.e. they are *vector* or *matrix operations* (as they are called in <u>MATLAB</u>).

To indicate that an operation shall refer to the individual elements of an array instead *(element-by-element operation),* it must be preceded by a `.` (period). Note that in MATLAB this is called an *array operation.*[5]

As an example, for an array `x` the operation `x^2` is undefined and results in an error message, whereas `x.^2` is well-defined and equals `x.*x`, i.e. it results in an array with the same dimension as `x`, containing all the elements of `x` squared. The scalar product of two vectors `x` and `y` of the same dimension is obtained by `x'*y`, where the prime denotes the conjugate transpose. The expression `x*y'` results in a matrix; this is called the *dyadic product* of `x` and `y`. Obviously, `x'.*y` is undefined and results in an error message.[6]

There are some pre-defined variables in MATLAB, the most important ones are these: `ans` is the default name for results when they are not explicitly attributed to any variable, `pi` gives the value of π, `eps` gives the machine precision (to be explained later), and both `i` and `j` represent the imaginary unit (positive square root of -1). Note that these really are variables, not constants, i.e. they can be overwritten.

---

[4]If in a calculation the elements of an array are not specified by a single assignment but are generated one after the other, the dimension of the array will automatically be adjusted. Internally, a new variable is created then, with the contents of the old variable being copied to and the new data included. Having this in a loop can slow down code execution considerably—which can be avoided by an initialization of the target variable, e.g. by assigning a final-size vector of zeros to it.

[5]Of course, an array operation also works on matrices of identical dimension. Be aware, therefore, that for square matrices `A` and `B`, also `A.*B` is well-defined but does *not* result in the matrix product—which is `A*B`.

[6]Note that it is worthwhile to use the appropriate operation symbol right from the beginning. For instance, in a program first written for scalar numbers the distinction between vector operation and element-by-element operation is irrelevant, therefore the dot preceding the operation symbol would not be necessary. However, any program that works for scalar numbers works also elementwise for arrays—but only if the dotted operator symbols are used.