In my opinion, the vectorized operation helps organize elements in a group that can be called simply by calling its name. It’s cleaner, easier, and faster to use vectorized operations when coding a lot of complex functions with multiple steps. The function will operate on all elements of a vector without looping through each element one by one [1]. Also, making a vector out of a data column of a dataframe runs a mathematical function as a separate set of data without impacting the original data frame, especially when it is a large data set. Writing a code with each newly made vector would be more efficient and less error-prone than calling the column of the dataframe each time repeatedly for multiple functions calls [2]. Vectorization also can be used to catch if there is a different data type within the same data set because a vector can only take the same type of data type. But, the risk of vectorizing different types of elements into a vector is that R will automatically convert them into a character type that will cause an issue when using them in mathematical functions. Another caution when performing mathematical functions with vectors containing different lengths of elements is that R will recycle the shorter vector to fulfill the function for the longer one. It’s always a good practice to check the length of a vector using length() function before performing complex mathematical functions of multiple vectors of large data sets.

Reference:

[1] https://docs.ycrc.yale.edu/r-novice-gapminder/09-vectorization/index.html

[2] https://www.noamross.net/archives/2014-04-16-vectorization-in-r-why/