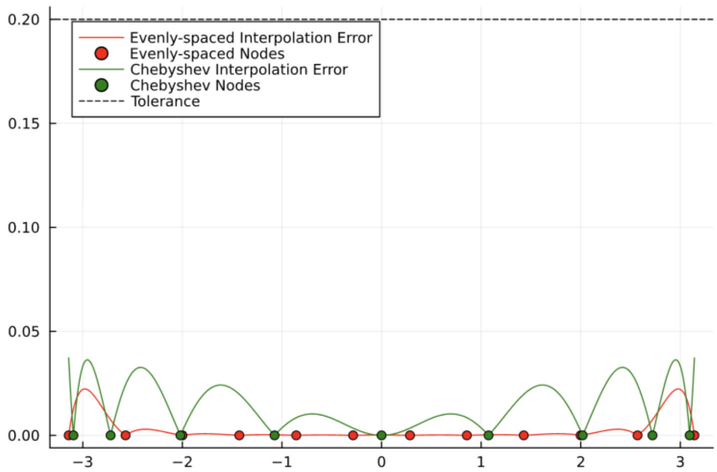
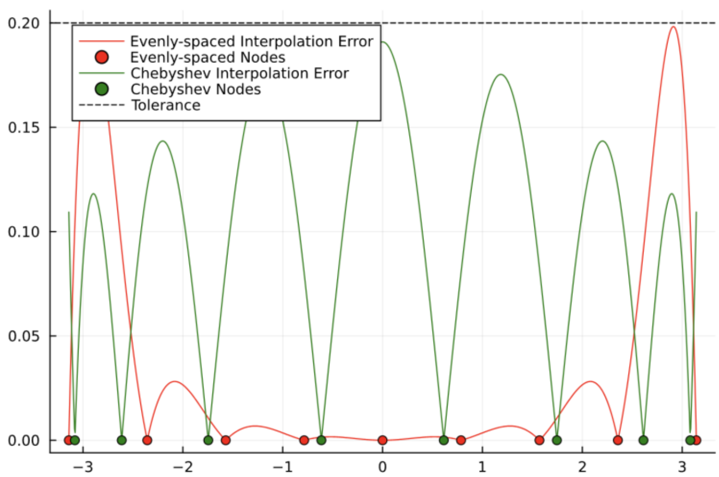
Cameron Beneteau (400274092)
beneteac@mcmaster.ca

Problem 5 (10 points): Run assignment2_plot.jl in the same folder as your completed assignment2_handout.jl file. Comment on the two strategies for selecting interpolation points xi: do they both satisfy the absolute error bound? Which requires more points to satisfy this bound? Give a qualitative description of the Chebyshev nodes' distribution.



Both strategies satisfy the absolute error bound. As seen in the graph, the interpolation errors of both methods over the interval [-π, π] stay significantly below the tolerance of 0.20. To achieve this, the subdivide method requires 12 nodes while the Chebyshev method requires 9 nodes. This calculation was done in assignment2_handout.jl. It is important to note these are not the exact minimum number of nodes required, due to the analytical method used by the functions. Rather, they are conservative estimates of the number of nodes required. Going above and beyond, the minimum number of nodes for each method is 9 and 8, respectively. This can be achieved with an iterative approach by creating the interpolating polynomial for n points and ensuring the error of all x values over the interval when plugged into the polynomial are less than the tolerance. If any value is greater, the process is repeated process with n + 1 nodes until a valid interpolating polynomial is found. The following image shows this result.



Cameron Beneteau (400274092)
beneteac@mcmaster.ca

As seen from the above charts, the green Chebyshev nodes are not equally spaced. They tend to be closer near the endpoints of the [-π, π] interval and sparser towards the middle. One of the problems with interpolating using evenly spaced nodes is the Runge phenomenon: large oscillations near the endpoints of the interval. Analysing the charts above, it is clear the error of the evenly spaced nodes interpolation grows when close to the endpoints, shown in red. Conversely, Chebyshev nodes, shown in green, are specifically chosen to be dense at endpoints, and sparse in the middle to minimize the maximum error over an interval. This, in turn, helps mitigate the Runge phenomenon and requires the same fewer nodes to achieve the same tolerance.