The pattern I see is that the larger the array gets the insert function gets slower. However, the append function which uses the push() method fares better.
For the most part push() will be faster and have a better time complexity than unshift().
Unshift is slower than push because it also needs to unshift all the elements to the left once the first element is added. In contrast, push() adds the element at the end of the array. You can see in the results below that the doublerAppend function that uses the push() method runs quicker, specifically when larger arrays are passed in. Push() default time complexity is constant O(1). As seen in the results below it doesn't matter how many elements are in the array, because the number of operations being performed won't change. Because Unshift() adds to the beginning the new element will have an index of 0 and all other indexes will change and be incremented, meaning it has linear time complexity.

Results for the extraLargeArray
insert 780.022375 ms
append 1.876417 ms

Results for the largeArray
insert 6.958917 ms
append 421.792 µs

Results for the mediumArray
insert 151.458 µs
append 104.375 µs

Results for the smallArray
insert 39.167 µs
append 69.875 µs

Results for the tinyArray
insert 27 µs
append 81.416 µs