1. My implementation here depends on A. I take A as an input and find n from A. The implementation is the same as the algrithm in 17-2 on note with simplification excluding all 0 entries in A.

The weight $w = \frac{2}{3}$, the eigenvalues are $1 - \frac{4}{3}\sin^2\left[\frac{k\pi}{2(n+1)}\right]$. For $\frac{n}{2} \le k \le n$, these lie in the interval $[-\frac{1}{3}, \frac{1}{3}]$, which is the closest to 0 among $w \in [0,1]$.

The implementation is in the Matlab files.

2. The reason it is a good test is because the numbers are random and there would be extreme cases with enough tests, Also, the actual answer is easy to calculate ($x = 0$).

The converging ratio $R = \frac{\|b - Ax^{k+1}\|}{\|b - Ax^{k}\|}$ seems to converge to a number $\alpha <= 1$, and α seems related to n. This might be explained by the change of eigan values with the growth in size of matrix. When $n \to \infty$, $\alpha \to 1$. In general, the speed of convergent is quite related to n. $n \downarrow$ speed $\downarrow$.

3. The ratio $R = \frac{\|b - Ax^{k+1}\|}{\|b - Ax^{k}\|}$ seems to be related to both k and n. $k \uparrow$ $R \downarrow$; $n \uparrow$ $R \downarrow$ (the jumps should be ignored, it might reach the min the computer could relognized).

What's more for weighted Jacobi 2/3 converges most quickly.