**Things to look for that are special to this error correcting technique include:**

1. (5 points). The **oscillations** mentioned above. See what happens (i) when you use a constant κ value (try κ = 0.1, from 1 - epsilon) and (ii) when you use κ values that decrease with number of iterations (κ = (1/(**f**iT**f**i)-epsilon)/j, where epsilon is a small number, say 1/1000, and j represents the number of iterations you’re on). (Oscillation here means a variable, e.g., error vector length, never converges but keeps increasing and decreasing continuously)

Set adjustment\_time = 25

(b) (5 points). **Convergence.** How long it takes to converge. Use error decrease smaller than 1% (from the previous iteration) as your criterion for convergence.

(c) (5 points). **Deterioration**. How many associations can be stored before the system starts to break down? If you think about it, a system breaks down when it associates f and g vectors no better than chance, right? Or in other words, you can get an output vector close to g with an input vector other than f, hint: generate a new input vector and compute its predicted output, is this predicted output closer to g than g’ (predicted output using input vector f) is to g?

(d) (5 points). **Sequential learning.** What would happen if we presented the associations in sequence rather than randomly? What happens in the linear associator if we present associations to be learned in different sequences? Try forwards and backwards.