Memory capacity of RNN in reservoir computing

Background: One of the essential topics in computational neuroscience is to borrow the properties of brain to develop so-called brain-inspired or neuromorphic computations, which is regarded as a potential road to next generation of artificial intelligence. On the other hand, these types of work might, in return, provide answers towards the capability of general intelligence of biological brains. The Echo-State Network, introduced by Jaeger et al (2004), provides a suitable framework for brain-inspired computation, which is also known as Reservoir Computing (RC). It's a special types of recurrent neural network (RNN), which is commonly regarded as an easy-trainable and relatively robust universal approximator of dynamical system. In this project, we ask: what advantages does the brain connectome have to for their learning and memory capability? Here, we mainly tackle on the memory capacity of the reservoir network as functions of their architectures. Does it help by introducing biological connectome? And how?

Project setups: We provide the basic framework of reservoir computing, and biological brain connectome for tests. [Jupyter Notebook demo]

Project map: The project core is covered by 1-5; subsequent questions can be taken in any order.

Q1. Build up a random echo state network (ESN) and train it with sequence-recall tasks described as follows. Train the ESN to predict the input identity at time $(t-\tau)$ from the network activity at time t. Your should use a separate ESN for each τ. You can plot the decoding performance as a function of τ. What is the memory capacity of the network you build? (defined as the decoding performance as τ increases)

Q2. Try varying the spectrum radius of the network, and how does the performance change?

Q3. Try to change the architecture of connectivity, such as small-world, scale-free networks. Does it make the performance better or worse?

Q4. Next, try turning the input magnitude, or the number of cells in the reservoir to receive input.

Q5. Replace connectivity of reservoir network with biological brain connectome. What does the memory capacity changes? Is it better than random networks?

Q6. Consider the functional modular segregation of brain connectome. Assign inputs and outputs to different different modulars (similar as Q4). Study the performance change. Is the modular segregation necessary in the perspective of computational capability?

Q7. A linear decoder (readout) might only limited stories. Try to replace linear decoder with polynomial basis, as $x_i^2, x_i^3, \cdots$. Does the performance improve?

Q8. Try adding heterogeneity to the time constant of neuronal in the reservoir network. The time constant can be either trainable or not. Does it improve the performance?

Q9. What if we use different types of training algorithm, e.g. FORCE? How does the memory capacity change?

Q10. What if we use unsupervised plasticity mechanism for training, such as Hebbian/anti-hebbian/asymmetric synaptic plasticity? Is this trainable? If so, what is the memory capacity?