Homework - Neural networks - Part A
Interactive activation and competition (35 points)

by **Brenden Lake** and **Todd Gureckis**
Computational Cognitive Modeling
NYU class webpage: https://brendenlake.github.io/CCM-site/
email to course instructors: instructors-ccm-spring2019@nyuccl.org

This homework is due before midnight on Feb 25, 2019.

Please complete the responses to these questions as a separate written PDF document.

In this assignment, you will get hands on experience with a classic neural network model of memory known as the Interactive Activation and Competition (IAC) model. We will go through a series of exercises that will stretch your understanding of the IAC model in various ways. The exercises below examine how the mechanisms of interactive activation and competition can be used to illustrate two key properties of human memory:

- Retrieval by name and by content.
- Spontaneous generalization over a set of familiar items.

These exercises are from Chapter 2 of the online PDP Handbook by James McClelland.

You should review the slides from lecture and read Section 2.1 of the PDP Handbook before continuing. This has important background and technical details on how the IAC model works. The IAC model instantiates knowledge that someone may have from watching the 1960s musical "West Side Story," where two gangs the "Jets" and "Sharks" struggle for neighborhood control in Manhattan. The "data base" for this exercise is the Jets and Sharks data base shown in Figure 1, which has the central characters from the two gangs. You are to use the IAC model in conjunction with this data base to run illustrative simulations of these basic properties of memory.

The Jets and The Sharks

Name	Gang	Age	Edu	Mar	Occupation
Art	Jets	40's	J.H.	Sing.	Pusher
Al	Jets	30's	J.H.	Mar.	Burglar
Sam	Jets	20's	COL.	Sing.	Bookie
Clyde	Jets	40's	J.H.	Sing.	Bookie
Mike	Jets	30's	J.H.	Sing.	Bookie
Jim	Jets	20's	J.H.	Div.	Burglar
Greg	Jets	20's	H.S.	Mar.	Pusher
John	Jets	20's	J.H.	Mar.	Burglar
Doug	Jets	30's	H.S.	Sing.	Bookie
Lance	Jets	20's	J.H.	Mar.	Burglar
George	Jets	20's	J.H.	Div.	Burglar
Pete	Jets	20's	H.S.	Sing.	Bookie
Fred	Jets	20's	H.S.	Sing.	Pusher
Gene	Jets	20's	COL.	Sing.	Pusher
Ralph	Jets	30's	J.H.	Sing.	Pusher
Phil	Sharks	30's	COL.	Mar.	Pusher
Ike	Sharks	30's	J.H.	Sing.	Bookie
Nick	Sharks	30's	H.S.	Sing.	Pusher
Don	Sharks	30's	COL.	Mar.	Burglar
Ned	Sharks	30's	COL.	Mar.	Bookie
Karl	Sharks	40's	H.S.	Mar.	Bookie
Ken	Sharks	20's	H.S.	Sing.	Burglar
Earl	Sharks	40's	H.S.	Mar.	Burglar
Rick	Sharks	30's	H.S.	Div.	Burglar
Ol	Sharks	30's	COL.	Mar.	Pusher
Neal	Sharks	30's	H.S.	Sing.	Bookie
Dave	Sharks	30's	H.S.	Div.	Pusher

Figure 1: Characteristics of a number of individuals belonging to two gangs, the Jets and the Sharks. (From "Retrieving General and Specific Knowledge From Stored Knowledge of Specifics" by J. L. McClelland, 1981, *Proceedings of the Third Annual Conference of the Cognitive Science Society.*)

1 Software and architecture

We will be using IAC software from Axel Cleeremans which you can download here. Upon downloading and loading the software, you will see a display that looks like Figure 2. The units are grouped into seven pools: a pool of visible *name* units, a pool of *gang* units, a pool of *age* units, a pool of *education* units, a pool of *marital status* units, a pool of *occupation* units, and a pool of *instance* units. The *name* pool contains a unit for the name of each person; the *gang* pool contains a unit for each of the gangs the people are members of (Jets and Sharks); the *age* pool contains a unit for each age range; and so on. Finally, the *hidden* pool contains an instance unit for each individual in the set.

The units in the first six pools can be called *visible* units, since all are assumed to be accessible from outside the network. Those in the gang, age, education, marital status, and occupation pools can also be called property units. The instance units are assumed to be inaccessible, so they can be called *hidden* units.

Each unit has an inhibitory connection to every other unit in the same pool. In addition, there are two-way excitatory connections between each instance unit and the units for its properties, as illustrated in Figure 3. Note that the figure is incomplete, in that only some of the name and instance units are shown. These names are given only for the convenience of the user, of course; all actual computation in the network occurs only by way of the connections. You can also view the different connections using the IAC software by hovering your mouse over a particular unit (Figure 4).

Since everything is set up for you, you are now ready to do each of the separate parts of the exercise. Each part is accomplished by using the interactive activation and competition process to do pattern completion, given some probe that is presented to the network. For example, to retrieve an individual's properties from his name, you simply provide external input to his name unit, then allow the IAC network to propagate activation first to the name unit, then from there to the instance units, and from there to the units for the properties of the instance.

2 Exercise: Retrieving an individual from his name

To illustrate retrieval of the properties of an individual from his name, we will use Ken as our example. Make sure the simulation is paused (press SPACE) and press 'r' to reset it. Set the external input of Ken's name unit to 1 by clicking on the name unit (not the hidden unit!). The circle's background should turn bright green to represent the external input.

A unit's activity level can be visualized by the colored dot, where yellow dots are positive activation and red dots are negative activation. The larger the yellow dot, the stronger the activation. A unit's precise activity level can examined by rolling the mouse over the unit.

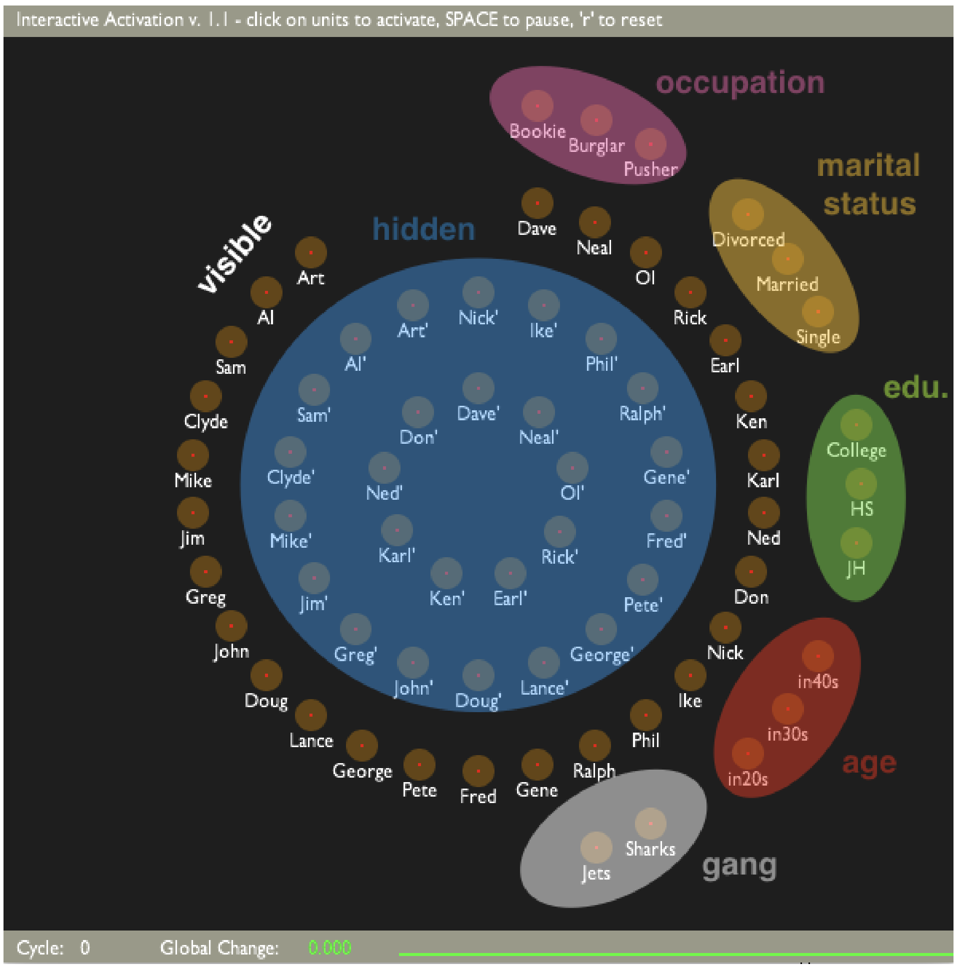

Figure 2: Screen shot from Cleeremans' IAC software. Units are organized into 7 groups. For illustration here, all groups have a different color background, while the group of visible name units have no background.

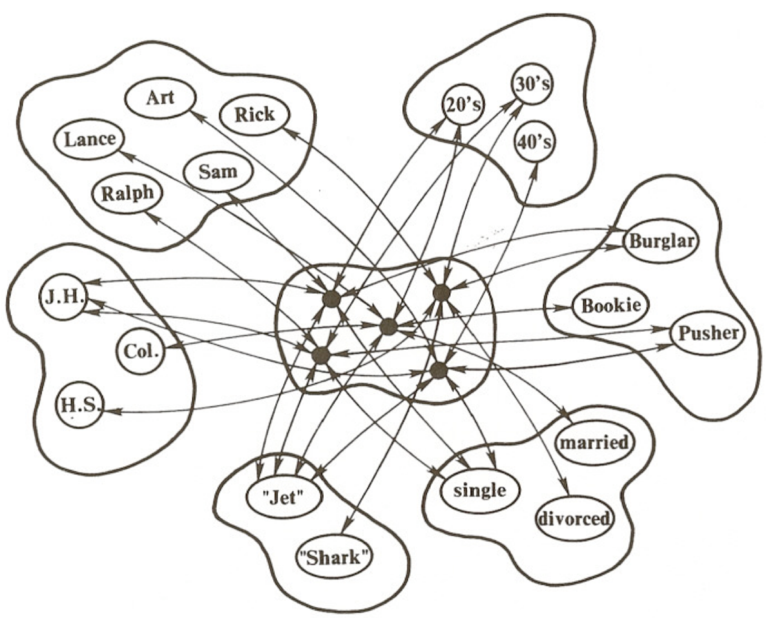

Figure 3: The units and connections for some of the individuals in Figure 1. The arrows represent excitatory connections. The outlined groups of units have mutually inhibitory connections (not shown). (From "Retrieving General and Specific Knowledge From Stored Knowledge of Specifics" by J. L. McClelland, 1981, *Proceedings of the Third Annual Conference of the Cognitive Science Society*.)

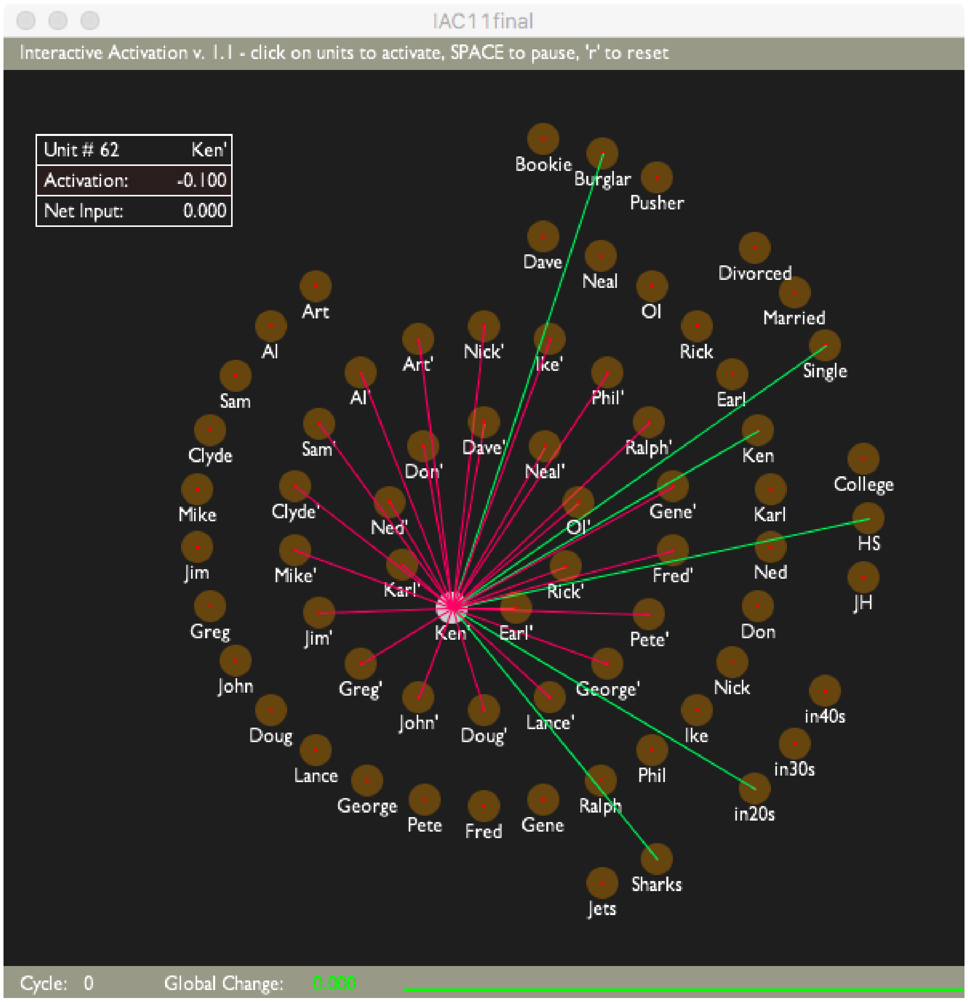

Figure 4: You can view the connections to a unit by placing your mouse over it. Green connections are excitatory and red connections are inhibitory.

Press SPACE to unpause and allow the network to run for approximately 150 cycles (the cycle counter is in the bottom left of the panel). The simulation runs quickly, so be sure to pause at about 150 cycles exactly! A picture of the screen after 150 cycles is shown in Figure 5. At this point, you can check to see that the model has indeed retrieved the pattern for Ken correctly. There are also several other things going on that are worth understanding. Answer all of the following questions below regarding the network at this state (you'll have to refer to the properties of the individuals, as given in Figure 1).

Problem 1 (10 points)

None of the visible name units other than Ken were activated, yet a few other hidden instance units are active (i.e., their activation is greater than 0). Explain why these units are active. Please turn in the response to this question (and the following) as a separate PDF document. Keep your response short (about 3 sentences).

Problem 2 (10 points)

Some of Ken's properties are activated more strongly than others. Why? Keep your response short (about 3 sentences).

3 Retrieval from a partial description

Next, we will use the IAC software to illustrate how it can retrieve an instance from a partial description of its properties. We will continue to use Ken, who, as it happens, can be uniquely described by two properties, *Shark* and *in20s*. Reset the network ('r') and make sure everything is paused and that all units have input of 0. Click to set the external input of the *Sharks* unit and the *in20s* unit to 1.00. Run a total of 150 cycles again, and take a look at the state of the network.

Of all of the visible name units, Ken's name should be the most active. Compare the state of the network's with the a screen shot of the previous network state when activating Ken's name directly, such as that in Figure 5.

Problem 3 (10 points)

Explain why the occupation units show partial activations of units other than Ken's occupation, which is Burglar. While being succinct, try to get to the bottom of this, and contrast the current case with the previous case. Keep your response short (about 3 sentences).

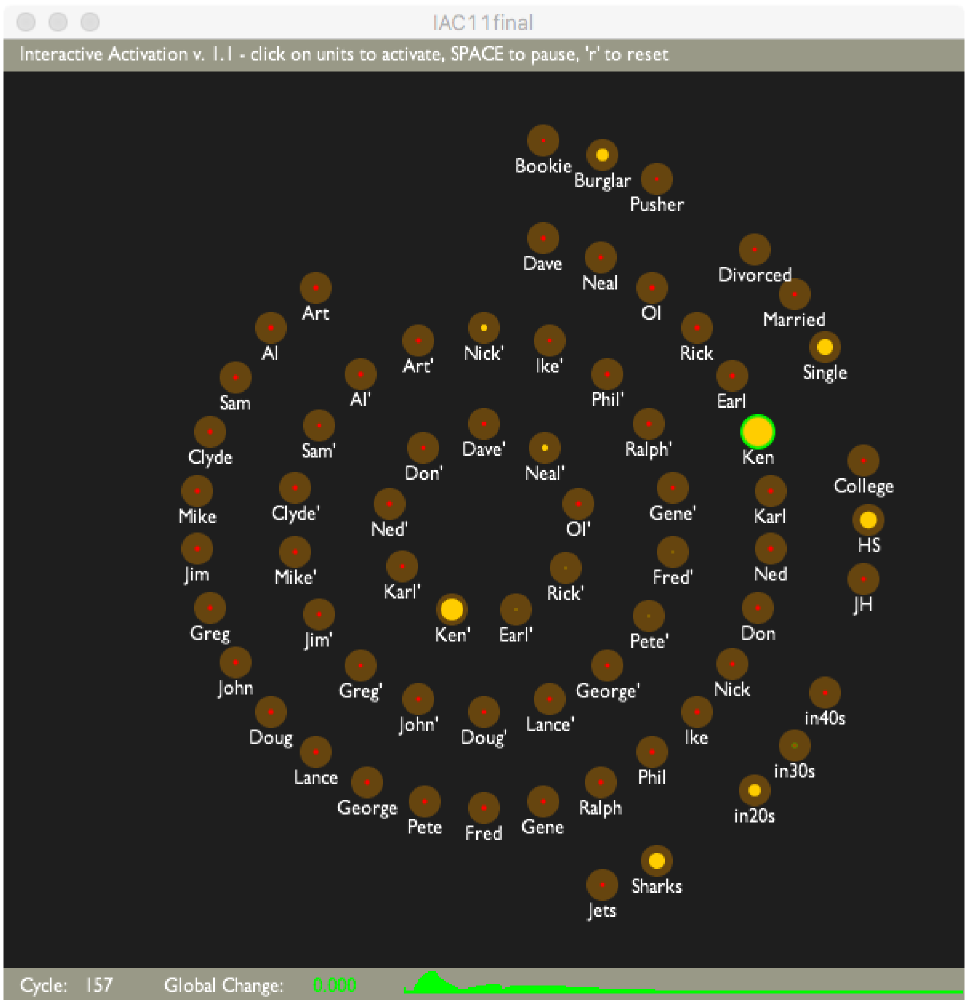

Figure 5: The display screen after about 150 cycles with external input to the name unit for Ken.

4 Spontaneous generalization

Now we consider the network's ability to retrieve appropriate generalizations over sets of individuals—that is, its ability to answer questions like "What are Jets like?" or "What are people who are in their 20s and have only a junior high education like?" Reset ('r') the network. Make sure all units have input of 0 and none are highlighted green.

Set the external input of Jets to 1.00 by clicking on it. Run the network for 150 cycles and observe what happens.

Problem 4 (5 points)

Given the network's state, what can you infer about a typical Jet? (1-2 sentences is plenty).

Again, please complete the responses to these questions as a separate written PDF document.

There are more parts of this homework which are Jupyter notebooks.

When all is complete, please submit via NYU classes