**Overview**

This tutorial shows you how to construct from start to finish a simple model of an common experimental task from Cognitive Psychology -- the CPT-AX task [1]. This one tutorial touches on all major aspects of the system. The tutorial can be read/worked through from your usual web browser, but these pages are actually intended to be accessed from within the ax_tutorial.proj project itself, under the docs section of the Navigator tree in the left panel of the GUI. For the latter, launch emergent, open the ax_tutorial.proj project. It can be downloaded from the emergent wiki here: tutorials/ax_tutorial (preferred method since it should always be the latest version, not requiring your updating Emergent to the latest source code); or ax_tutorial.proj can be found in the your 'Emergent' folder under demo/leabra/ax_tutorial.proj. From within ax_tutorial.proj select the ProjectDoc tab which is pinned to the top of the middle Editor panel (should be showing when you open the project, or accessed from the docs/ProjectDoc section of the left Navigator panel).

Since there are many links in the tutorial that refer to the objects within the emergent project itself, most users find it most productive to take the within ax_tutorial.proj approach, although that will result in these documentation pages "going away" frequently since many things using the middle Editor panel where these pages are displayed. This includes the program editor itself where we will be spending the greatest amount of our time. Therefore, if taking that approach you should familiarize yourself with finding all of these wiki documents again when you need them by clicking on their links near the top of the left Navigator tree under the docs section. Other aids some have found useful are: 1) access these pages in your usual web browser and print them out so you can have them alongside as you make your way through the project; and, 2) Find the taDoc menu at top left of the Editor panel with this documented opened there. In the drop down menu revealed by clicking on the little arrow to the left of taDoc select Open in Window to create a separate window that will stay around all the time (although it still gets obscured a lot).

**Building a Complete Model**

This project provides step-by-step directions for constructing a working neural network simulation from the ground up, including programming a simple psychological task (target detection), which we extend through several stages to ultimately simulate the more complex CPT-AX task used in working memory studies.

Some basic terminology:

- **Left Navigator panel** is the left portion of the window with a "tree" of objects in the simulation (including the network, and the input/output data, etc).
- **Middle Editor panel** is where you are currently reading -- it can display different things depending on the selected tabs at the top, and what is currently selected in the left Navigator panel. The left-most tab in this middle Editor panel usually shows what is selected in the left Navigator panel, and the other "pinned" tabs are locked in place and contain this document and the Wizard, which we will be making heavy use of. The right-most tab in the Editor panel represents the configuration information for the current 3D display shown in the right-most Visualizer panel. The current "Frame1" tab is empty, because the corresponding "Frame1" view panel is empty.
- **Right Visualizer panel** shows 2D & 3D displays of various simulation objects, including the network, input/output patterns, and graphs of results, etc. The current view displayed is called "Frame1" and is empty.
Basic Task: Target Detection

The basic task we'll start with is a toy version of the CPT-AX that simply involves presenting letters one at a time to the network, and having it distinguish "targets" from "non-targets". The targets are the letters 'A' and 'X', and the non-targets are 'B', 'C', 'Y', and 'Z'.

The network will have six input units representing each of these letters, and two output units, one for "target" and the other for "non-target". It will have a hidden layer to learn the mapping (though this task is initially so trivial that it doesn't even require a hidden layer -- we'll make it harder later).

Chapters

Here are the steps we'll go through, organized as separate document chapters:
1. AXTut BuildNet -- build the network
2. AXTut InputData -- make basic patterns (data) to present to the network
3. AXTut Programs -- create and control the programs that perform the simulation
4. AXTut OutputData -- monitor and analyze the performance of the model
5. AXTut TaskProgram -- write a program to construct the task input patterns, including more complex tasks
6. Extras: elaborations that go all the way to the full CPT-AX task
   1. AXTut CPTAX_Program -- extend our basic program to the full CPT-AX task
   2. AXTut PfcBg -- add a prefrontal cortex/basal ganglia to the model to handle the full CPT-AX task

If You Get Off Track...

In the same directory where you loaded this project is a ax_tutorial_final.proj project file, which has the full project that will result from following these directions (not the extras). You can load this project and refer to it to see what things are supposed to look like.

Let's begin...

→ Go to BuildNet

AX Tutorial Archive

AX_Tutorial_v5.0.2.0

References

Article Sources and Contributors