

Simulations of speech perception in users of cochlear implants

Introduction

Cochlear implants (CIs) substitute for a damaged inner ear by electrically stimulating residual auditory nerve fibres directly. An important determinant of performance with CIs, which varies widely from person to person, depends on the extent to which distinct frequency channels of information are transmitted to users. In this laboratory session, you will perform a speech identification task with sentences that have been processed to simulate something like what a CI user would hear (using *noise-excited vocoding*).

Materials

A computer-based program will present the sentences, gather the responses, and keep track of your levels of performance. On every trial, you will hear a sentence of the type 'Show the [animal] where the [colour] [digit] is' where 6 animals, 6 colours and 8 digits are possible (e.g., *Show the dog where the red 6 is*). You will then click on one of 48 response boxes. Feedback will be given in the form of a smiley or sad face. You only get the smiley face when both the colour and the digit are correct. The target signals will vary in the number of independent channels of information they contain, with 1, 2, 4 and 8 channels.

Method

The test will be run in Matlab, a '*high-level language and interactive environment for numerical computation, visualization, and programming*' that is used in many areas of science. To start Matlab, click on the **bell** icon on the desktop, and navigate to the folder [\\bell\\btemp\\stuart\\CCRM](#). Click on the file **CCRM.m**, and this will start Matlab.

First, put your cursor in the *Command Window* at the bottom middle of the page, and type **MyOrder**. This will give you a random order of the 4 conditions, except that everyone will start with the easiest condition, 8 channels, in order to familiarise yourself with the task and the unusual sounds. Write these into the table overleaf. Once this is done, type **CCRM** into the *Command Window* which will start the program.

You should then see a GUI like the one at left. You will pick the number of channels in the sentences with the drop-down menu at top left labelled *Targets File* (one of **1channel.txt**, **2channels.txt**, **4channels.txt** or **8channels.txt**).

Also make sure to enter your initials for the Listener Code at the bottom left. Change no other value!

Run yourself through the 4 conditions (there are 20 trials in each). There is no point in trying to think too hard about the response when you are not sure. Go with your initial impulse.

Observations

Your results files can be found under 'Documents\AUDL4007Results'. There are two files generated for each test, both of them being .csv files which can be opened in Excel. One of them has the trial-by-trial record of the test, and the other some summary values. You probably only need to look at the summary files (indicated by '_sum' in the name). The file names indicate which targets were used, and the time of the test, for example:

SR_GUI_spk1_01_12-Oct-2013_14-26-12_sum.csv

The crucial outcome measures are at the far right hand side of the spreadsheet (colourCorrect, colourPC, digitCorrect, digitPC, PC). 'PC' stands for *proportion correct*; the other numbers are simple counts of the total number correct.

target type	% correct colour	% correct digit	% correct overall
8-channels			
-channels			
-channels			
-channels			

You will plot your results along with the rest of the class on the white board. Use an 'x' if you are a non-native speaker of English, and 'o' if you are a native speaker.

Things to think about

1. Why do you think the digit '7' isn't used in the test?
2. Were colours or digits correctly identified more often? Why might this be?
3. Did all the conditions sound something like speech?
4. What is the chance level of performance on this task? Did every condition lead to levels of performance significantly higher than chance? What kinds of cues were available in the single-channel case?
5. If you have time, you could look at the error patterns, especially in the 1-channel condition where you probably made most errors. Think about how you might construct a *confusion matrix* which shows not only which trials you got correct, but also shows which colours or numbers were confused with others.