

Problem set 2

For this problem set, you will use Python and PsychoPy to create an experiment that measures a person's ability to identify faint visual signals in white noise.

1. Create an experiment with 200 trials. On each trial, a Gabor stimulus appears at the centre of the screen, rotated 5° to the left or right of vertical. For the Gabor, use an image size of 128×128 pixels, a wavelength of 16 pixels, and a scale constant (the σ parameter of the Gaussian window) of 10 pixels. The Gabor's contrast is randomly chosen from the values 0.01, 0.02, 0.04, 0.06, and 0.08. (You can adjust this range if the task turns out to be too hard or too easy.) The Gabor is shown in pixelwise white noise with a contrast standard deviation of 0.2. The stimulus duration is 500 ms. After the stimulus disappears, the subject presses one of two keys to indicate whether the Gabor was tilted left or right. A high or low beep provides auditory feedback on whether they were correct. After a short pause, the next trial begins. The subject can also press a third key to quit the experiment early.

On each trial, save the following information on a new line in a text file: trial number, signal contrast, signal orientation ($\pm 5^\circ$), whether the signal orientation was left or right of vertical (0 or 1), whether the subject indicated that the orientation was left or right of vertical (0 or 1), whether the response was correct (0 or 1), and the reaction time.

Create a separate module called 'tools' where you define a function `gabor()` that takes arguments for wavelength, orientation, image size, etc., and returns the Gabor as a numpy array. Import this module into your experiment script, and use it to create the Gabors.

In the same tools module, define a function called `beep()` that plays a short beep, using the `sounddevice` module that we covered as an alternative to PsychoPy's sound module. The function should take one or more arguments that allow you to play a high or low frequency beep.

To show the stimulus, use a `visual.ImageStim()` object. You can read PsychoPy's help to learn how to use this object. You will need to use the 'win', 'image', and 'size' arguments.

2. In a separate script, write an analysis routine that loads the data and plots proportion correct as a function of signal contrast. Fit a smooth curve to the data, e.g., a normal cumulative distribution function, and plot the curve as well. Save the figure as PDF file. From the fit of the curve, find and print the subject's 75% threshold, i.e., the signal level required to achieve 75% correct.

Email your solutions to me (`rfm@yorku.ca`) in a single `.zip` or `.tar` file named with your last name in lowercase, e.g., `murray.zip`. Be sure to include sufficient help text and comments in your code.

Due December 12, 2025