Computational Models of Language and Beyond

Assignment IV

A connectionist vowel detector

Please hand in the completed assignment the latest on Friday the 26th at 6pm, preferably by e-mail to s.l.frank@uva.nl.

Vowels and formants

Some languages have only a few vowels. For instance, Arabic has just three: /u/, /a/, and /i/. Polynesian languages and Greek have five: /u/, /a/, /i/, /o/, and /e/.

Different vowels can be characterized (and identified) by the central frequencies of the first two formants, which are usually denoted by F_1 and F_2 (if this is new to you, see http://en.wikipedia.org/wiki/Formant). The approximate formant frequencies for the five Polynesian/Greek vowels are plotted in Figure 1.

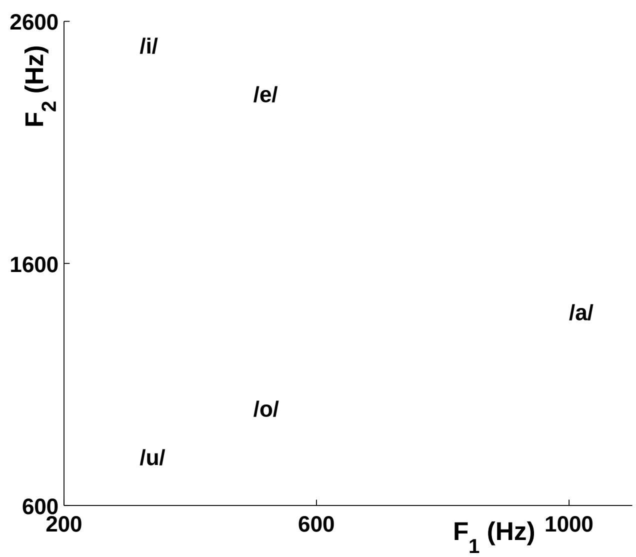

Figure 1: Approximate 1st and 2nd formant frequencies for five vowels.

Detecting vowels

The objective of this assignment is to construct a neural network that recognizes different vowels. That is, it should take as input the values of F_1 and F_2, and activate *only* the output unit that corresponds to the correct vowel. For simplicity, it is assumed that formant frequencies are never outside the scope of Figure 1, so F_1 is always between 200 Hz and 1200 Hz, and F_2 is always between 600 and 2600 Hz.

We start with a very simple network that merely indicates whether or not the vowel /u/ is detected. This network, shown in Figure 2, consists of just one unit, which is labeled /u/ (because it's supposed to be an /u/-detector).

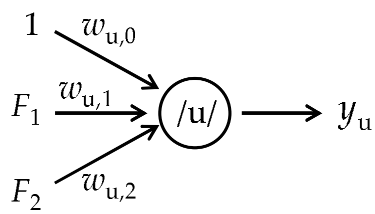

Figure 2: A one-unit neural 'network' for detecting /u/.

Unit /u/ receives a total input that equals:

$$x_u = w_{u,0} + w_{u,1}F_1 + w_{u,2}F_2 \tag{1}$$

The unit's activation function is a step function, so its output is:

$$y_u = \begin{cases} 0 & \text{if } x_u < 0 \\ 1 & \text{if } x_u \geq 0 \end{cases} \tag{2}$$

If $y_u = 1$, unit /u/ has detected the occurrence of vowel /u/. If $y_u = 0$, the vowel has not been detected. The 'borderline' between the two cases occurs when the total input equals zero:

$$\begin{aligned} w_{u,0} + w_{u,1}F_1 + w_{u,2}F_2 &= 0 & \iff \\ w_{u,2}F_2 &= -w_{u,0} - w_{u,1}F_1 & \iff \\ F_2 &= -\frac{w_{u,0}}{w_{u,2}} - \frac{w_{u,1}}{w_{u,2}}F_1 \end{aligned} \tag{3}$$

Make sure that you understand this before reading on.

Assignment

In Figure 1, draw one straight line that separates /u/from the other four vowels (you don't need to hand this in). If the inputs F_1 and F_2 are on the '/u/-side' of this line, then /u/ should be detected, meaning that $y_u = 1$. Otherwise, $y_u = 0$ because /u/ is not detected. So, we need connection weights $w_{u,0}, w_{u,1}$, and $w_{u,2}$ that make sure the network will behave like that.

Question 1 Looking at Equation 1, should the values of $w_{\mathrm{u},1}$ and $w_{\mathrm{u},2}$ be positive or negative? Why?

Actually, only two of the three weights need to be specifically chosen (bonus question: why is that?) so we can make things much easier by setting $w_{\mathrm{u},2} = +1$ or $w_{\mathrm{u},2} = -1$, depending on your answer to Question 1.

The line you just drew (like any straight line) can be described by an equation of the form:

$$F_2 = b + mF_1 \tag{4}$$

Question 2 Compare Equations 3 and 4, keeping in mind that $w_{\mathrm{u},2} = \pm 1$. Choose connection weights $w_{\mathrm{u},0}$ and $w_{\mathrm{u},1}$ such that the network becomes an /u/-detector.[1]

Question 3 So far, the network just tells us whether or not /u/ has been detected. We want to extend it to also detect the other two Arabic vowels (/a/ and /i/). This means that two output units are added, resulting in six new connections. Choose a set of connection weights that turn the network into an 'Arabic' vowel-detector. Note that *only* the output that best matches the input should be activated, but it's okay if *no* output is active when the input is ambiguous or is unlike any of the three vowels.

Question 4 Next, we want a 'Greek' vowel-detector that correctly classifies inputs into /u/, /a/, /i/, /o/, and /e/. Again, at most one output should be active at any time. Explain why a single-layer network cannot perform this task.

Question 5 To detect the vowel /o/ (as distinct from the other four vowels) a *two*-layer network is needed. Such a network has an /o/-unit as output, which is active only when the input is more like /o/ than like anything else. Also, there are at least two hidden units, which receive the input and send activation to the /o/-unit. Assume that these hidden units have the step function (Equation 2) as activation function.

Have another look at Figure 1 and explain why a network with hidden units is able to separate /o/ from the other vowels[2] and how it can do so. Why can a network with 3 hidden units act as a more selective /o/-detector than a network with only 2 hidden units?

---

[1] Solving this problem requires some basic high-school math. If you need to refresh your memory, check out http://mathworld.wolfram.com/Line.html

[2] The simple answer that two-layer networks can compute *any* function is not sufficient.