

Lab session 4

Summary

Today we will measure formants for a further example from the Boston Radio News corpus (BRN). We will do this manually first, then completely automatically by way of a praat script. Finally (and scientifically the most reasonable way) we will create a TextGrid to mark the places where we want to obtain the formants, then run a script to extract the formant values at the specified locations automatically.

If you are done and have time left, there is an optional part where you measure formants the way I would recommend it for "serious" student projects.

Measuring formants manually

Open speech signal and spectrogram

Open praat and load f2bs01p2.wav and f2bs01p2.TextGrid from

```
/mount/studenten/MethodsCL/2019/Speech/Lab4
```

Select both objects together, then click View & Edit.

Make the spectrogram visible by checking Spectrum → Show Spectrogram in the menu, and formants and pitch invisible by unchecking Formant → Show Formants and Pitch → Show pitch.

Measure visually in spectrogram

Work through the first few vowels in the TextGrid. (Note that the TextGrid was created automatically and may contain incorrect or inaccurate labels!!) Try to estimate the formant values by looking at the spectrogram. **Usually, formants should be measured in the middle of the stable part of the vowel, i. e. in the middle of the part where the bars are as parallel as possible.**

Click into the spectrogram at the time where you think the vowel is most stable and at the frequency where you think the formant is located. Read the frequency off the left margin of the window. Write down the values.

Measure visually in spectrum

Click into the spectrogram at the time where you think the vowel is most stable (i.e. no region marked, but a dotted line indicating the location of the cursor). Select `Spectrum → View spectral slice` from the menu. This will create a new object in the object window, which you can view on its own by `View & Edit`.

Determine the formants by clicking at the peaks. Do the values match with the ones you wrote down before?

Have praat measure the formants

Since it is sometimes not easy to decide which peaks are actually formants, it may help to look at spectra of the surrounding sounds. This is what praat does when you check `Formant → Show formants` from the menu: it makes decisions about formant locations by taking into account the candidate formants of the surrounding frames, considering continuity constraints (i.e., praat exploits the knowledge that in human speech, formants can not have too abrupt jumps, since the vowel tract shape cannot change too abruptly).

You can either query individual formant values by selecting `Formant → Get first formant` from the menu (or the second and third formants). Or you can list all formants and their bandwidths at once by selecting `Formant → Get formant listing`.

Try it out and compare to your values from above.

Measuring formants automatically

Measuring everything step by step and manually is time-consuming. We try the other extreme and use a praat script to determine the formant values at the middle of each vowel.

Open the script `get.formants.for.intervals.praat` by choosing `Praat → Open praat script...` from the object window menu. (Beware, not the `Open → Read from file...` command we used to open sound and textgrid!) The script is located in the same Lab4 folder as above, and it is also attached to today's session in ILIAS. If you are running praat on a Mac, you will find the `Open` menu not in the Objects window but in the Praat menu at the top of your screen.

Select sound and textgrid object in the object window. Then go to the script window, and run the script using `Run → Run`. It will prompt you for a regular expression to describe the labels for those vowels that you are interested in, suggesting an expression that should cover all vowels in our example except for `ih`, which I removed because it is so often reduced or colored by neighboring sounds due to coarticulation that it would pop up all over the place

and obscure the other vowels in the plot (but put it in if you are interested). The dialog box also asks you in which tier in the textgrid to look for the vowels; in our example, they are in tier 1.

The script should produce a listing of the first three formants for all intervals corresponding to the regular expression. It will always measure exactly in the middle of the labeled interval. It then plots F1 and F2 values into a formant plot.

Do the vowels appear in the expected places? Remember in the lecture we claimed that F1 and F2 in vowels correspond to vowel height and frontness. Compare to the diagram from the lecture.

You should find some outliers, for instance two ae vowels which are not in the place where most other vowels of these types are. You can track them down: when you ran the script, you should have gotten a listing of all measured vowel labels, the time points where they occurred, and the first two formants, all in a separate window, the Praat Info window. You can search for vowel names in this window using <Strg>-f (for the first occurrence) and <Strg>-g (to continue the search).

Identify the outlier you want to track down, and note the approximate outlying value. For instance, one [ae] in the plot seems to have an F1 of around 420 Hz, while other [ae] vowels have an F1 of around 600 Hz or above. Search through your listing for occurrences of [ae] until you find one with this suspicious value (there should be one at 2.48 seconds). Using this time point from the listing, find the correct time in the textgrid. You will find that this particular vowel came from the word "at", and the way the speaker pronounced it, it is indeed closer to an ih than to an ae: this is due to vowel reduction, which we often find in function words such as as. Similarly, there is an [eh] vowel very low in the plot, with an F1 of around 690, and an F2 of around 1250. You can track it down in the list and find it at 6.72 seconds. If you check the TextGrid with the sound it turns out that in the [eh] at that point you can already hear the upcoming "dark" l in the r – and that impression is caused by a more back pronunciation. However, you can also observe that the boundary for the [eh] is actually too far into the l. If you move the boundary so that you don't have parts of the [l] sound in the interval of the [eh] label, then rerun the script, you will find that the [eh] has moved more to the front, because now we are measuring at a more appropriate point.

In other cases, you might find that vowels might be reduced to schwa, or to [ih].

This script will also work on the TextGrid you created in the previous lab session. So you can now validate your annotation from last time, by loading the TextGrid and the Sound, selecting both in the object window, and then running this script. Look at the formant plot so find out if your vowel labels seem to correspond to the formant values.

Tips and tricks for the future (Theses, seminar projects!!)

Note that you can save the content of the info window in a .txt file, and the content of the graphics window in a PDF file. You can then run statistics on the values reported in the .txt file, and use the PDF for illustrating graphs.

If you do something like this for a thesis or project, you should also look at the procedure of measuring formants suggested in the following optional part of this lab session.

Optional: Measuring formants semi-automatically

So the problem with the script above is that the textgrid is inaccurate at some places, or has wrong labels. Also, in some cases, the point exactly in the temporal middle is not necessarily where the formants are most stable. A better procedure for such cases is to manually specify the places where praat should measure formants in a special tier, then do the measurement automatically.

Create a tier to indicate time points for measuring

Go to the spectrogram window. Insert an additional tier into the textgrid. The script expects a point tier, so use Tier → Add point tier from the menu.

Add point labels at the places where you want the formants measured. (This works as before: click into the spectrogram, and then into the little circle that appears in your new tier.) You have to give these points names; these will displayed later in the vowel space, so it makes sense to just give them the name of the vowel in which they occur. If you want to be able to identify which data point corresponds to which vowel, you could add indices to your vowel names.

Please don't forget to save the textgrid.

Run script for measuring

When you have a good number of vowels marked this way, open the script `get.formants.for.points.praat` (Remember the command for opening scripts is in the Praat menu, not in the Read menu). The script is again in the Lab4 folder, an in today's ILIAS session. Select sound and textgrid together, then run the script. This time you only have to specify the number of the tier in which you labeled the points for measuring. It is not necessary to specify a regexp for vowels because it will measure formants for any point that was labeled in the tier.

Does the formant plot have less outliers now?