Praat Scripting
06 Tables

In order to create a nice output for you measured data (and ignoring the Praat Info Window), there is the possibility to generate a table object. The following script[1] associates duration measurements with a table object containing two columns: `column1` durations of lax vowels, `column2` durations of tense vowels. This script only works for a TextGrid that has been segmented for vowels only. The difference between tense and lax vowels is made by using the symbol ":".

```
name$ = selected$ ("TextGrid")
noi = Get number of intervals:  1

# creates table with two columns ('kurz' and 'lang') and one row
Create Table with column names:  name$, 1, "kurz lang"

# initiate auxiliary variable
kurz_no = 0
lang_no = 0

for n to noi
  selectObject:  "TextGrid " + name$
  label$ = Get label of interval:  1, n
  if label$ <> ""
    start = Get start point:  1, n
    end = Get end point:  1, n
    dauer = round((end-start)*1000)
    # does label contain ":"?
    is_long = index(label$, ":")
     selectObject:  "Table " + name$
    nor = Get number of rows
     # if tense vowel...
    if is_long > 0
      lang_no = lang_no+1
      if lang_no > nor
        # add row
        Append row
       endif
      # add duration
      Set numeric value:  lang_no, "lang", dauer
       # if lax vowel...
    else
      kurz_no = kurz_no+1
      if kurz_ no > nor
        Append row
       endif
      Set numeric value:  kurz_no, "kurz", dauer
     endif
  endif
endfor
```

[1] Phonetische Analysen mit Praat Ein Handbuch für Ein- und Umsteiger; http://praatpfanne.lingphon.net/downloads/script431.txt [06.06.2014], changed to new syntax accordingly, not available anymore [21.07.2015]

You can read in any file as a table file if it shows a consistent structure and includes a header. Therefore you can either read in a tab-, comma- or whitespace-separated file using the following arguments

```
Read table from tab-separated file:
Read table from comma-separated file:
Read table from whitespace-separated file:
```

You can also save your measured data (e.g. durations, formants, . . .) without creating a Table file from the beginning like we did before: create a header and assign the measured values. Later, you can read it in as a table file.

```
appendInfoLine:  "fileName", tab$, "Label", tab$, "Duration[ms]", tab$, "F1[Hz]", tab$,
"F2[Hz]"
appendInfoLine:  name$, tab$, label$, tab$, duration:2, tab$, fixed$(f1,0), tab$, fixed$(f2,0)
```

in order to create the following output

```
fileName Label Duration[ms] F1[Hz] F2[Hz]
narrative1 ?  25.91 788 1392
narrative1 aI 154.26 648 1642
narrative1 n 47.50 322 1670
narrative1 S 97.02 1821 2395
```

What you need to do now, is to save these information directly in a file (textfile .txt or table file .table) and not just print out the data in the PraatInfo window. To do that, you can use the argument `writeFileLine` and `appendFileLine`.

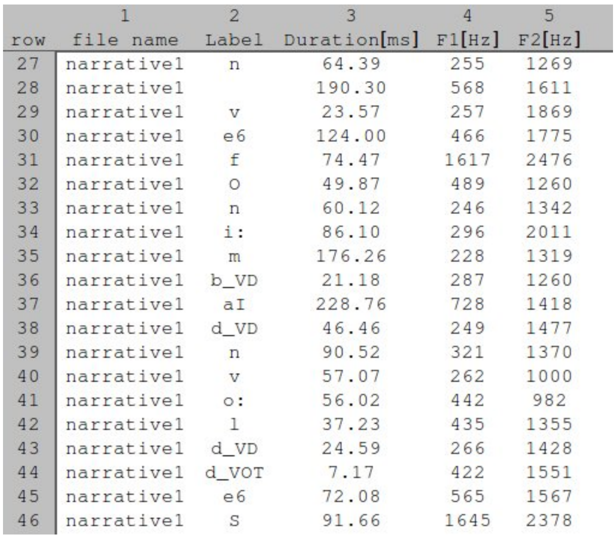
To create a new file, use

```
file$ = "formants.txt"
writeFileLine:  file$, "fileName", tab$, "Label", tab$, "Duration[ms]", tab$, "F1[Hz]",
tab$, "F2[Hz]"
```

To append text at the end of an existing file, you use

```
appendFileLine:  file$, name$, tab$, label$, tab$, duration:2, tab$, fixed$(f1,0), tab$,
fixed$(f2,0)
```

Use `writeInfo` or `appendInfo` if you don't want a newline at the end of the file.



	1	2	3	4	5
row	file name	Label	Duration[ms]	F1[Hz]	F2[Hz]
27	narrative1	n	64.39	255	1269
28	narrative1		190.30	568	1611
29	narrative1	v	23.57	257	1869
30	narrative1	e6	124.00	466	1775
31	narrative1	f	74.47	1617	2476
32	narrative1	O	49.87	489	1260
33	narrative1	n	60.12	246	1342
34	narrative1	i:	86.10	296	2011
35	narrative1	m	176.26	228	1319
36	narrative1	b_VD	21.18	287	1260
37	narrative1	aI	228.76	728	1418
38	narrative1	d_VD	46.46	249	1477
39	narrative1	n	90.52	321	1370
40	narrative1	v	57.07	262	1000
41	narrative1	o:	56.02	442	982
42	narrative1	l	37.23	435	1355
43	narrative1	d_VD	24.59	266	1428
44	narrative1	d_VOT	7.17	422	1551
45	narrative1	e6	72.08	565	1567
46	narrative1	S	91.66	1645	2378

You can run **some statistics** on your data table as well. Unfortunately that is quite laborious but possible…

Query →

List:	copies the table into the Praat Info Window
Get number of rows	prints out the number of rows
Get number of columns	prints out the number of columns
Get column label:	prints out the name for a specific column
Get column index:	prints out the index for a specific column
Get value:	prints out the content for a specific cell
Search column:	searches for a specific value in a specific column
Get quantile:	calculates out the quantile for a specific column; you can specify which quantile you want to calculate
Get minimum:	calculates out the minimum for a specific column
Get maximum:	calculates out the maximum for a specific column
Get mean:	calculates the mean for a specific column
Get group mean:	calculates the group mean for a specific sub-table an not the whole column
Get standard deviation:	calculates the standard deviation for a specific column
Report difference (Student t):	t-test for a specific column
Report group difference (Student t):	t-test for a specific sub-table and not the whole column

In order to be able to create a sub-table of your data you have to extract a part of it.

Extract →

Extract rows where column (number):	extracts all rows with a specific numeric value
Extract rows where column (text):	extracts all rows with a specific string value
Collapse rows:	here you have the possibility to to calculate mean and median for several variables (e.g. F1, F2, F3) for a given independent variable (e.g. Label); you will get an additional table with the results

To change specific parts of your table you want to use one of the following arguments

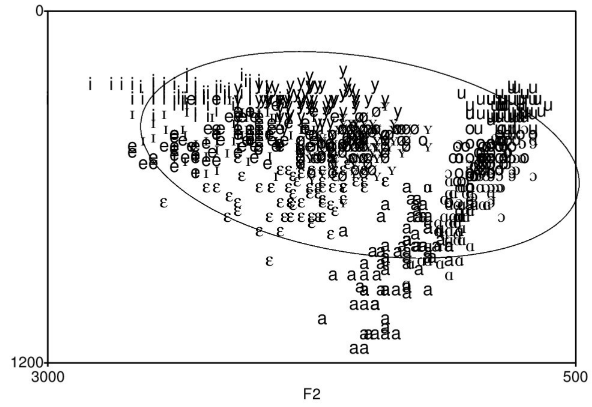
Modify →

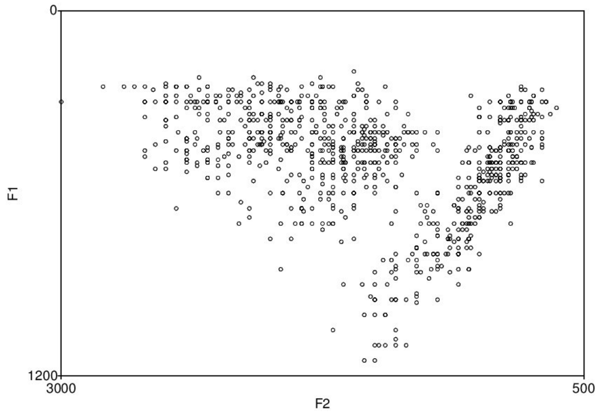
Set string value:	sets string value for a specific row and column
Set numeric value:	sets numeric value for a specific row and column
Formula:	uses a formula for a specific column
Formula (column range):	uses a formula for more than one specific column
Sort rows:	sorts rows for a specific column
Randomize rows	randomizes rows for all columns (no specific information about randomization type)
Append row	appends a row for all columns
Append column:	appends a new column
Append sum column:	appends a new column that adds up the values of two columns
Append difference column:	appends a new column that subtracts the values of two columns
Append product column:	appends a new column that multiplies the values of two columns
Append quotient column:	appends a new column that divides the values of two columns
Remove row:	removes a specific row
Remove column:	removes a specific column
Insert row:	inserts a row at a specific position
Insert column:	inserts a column at a specific position
Set column label (index):	names a specific column
Set column label (label):	renames a specific column

And last but not least some nice draw functions:

Draw →

Scatter plot:	creates a scatter plot for specific variables (e.g. F1-F2 plot of Label)
Scatter plot (marks):	creates a scatter plot for specific variables (e.g. F1-F2 plot of Label) using (+ x o .) instead of e.g. IPA symbols
Draw ellipse (standard deviation):	draws an ellipse around all data points within two standard deviations of the mean (~95% of the data)





Exercise 07

Create an already existing table in Praat which contains formant data (`Create formant table (Pols & Van Nierop 1973)`). Write a script that draws a F1-F2 scatter plot for only one vowel including an ellipse (2 standard deviations). The script should also give the user the possibility to

a) choose the name of the Table object

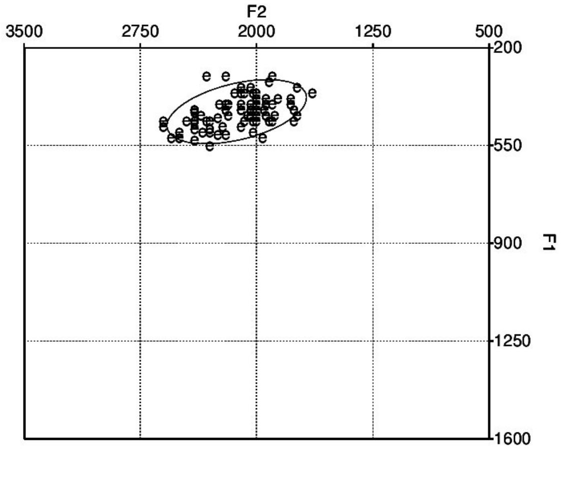
b) choose the column name that contains the vowel label

c) choose the vowel that should later be drawn in the scatter plot

d) choose colour of the drawn vowel

e) change information about: font size, lower and upper F1, lower and upper F2 limit, standard deviation for drawn ellipse

Captions and values for F1 and F2 should be right and top, respectively (see below).


F2
3500
2750
2000
1250
500
200
550
900
1250
1600
F1