

# ***SCRIPTS EN PRAAT PARA LA EXTRACCIÓN DE DATOS TONALES Y CURVA ESTÁNDAR***

*Miguel Mateo Ruiz*  
Laboratori de Fonètica Aplicada  
Universitat de Barcelona

## **Resumen**

Este documento ofrece los *Scripts* para Praat presentados en nuestro artículo “Protocolo para la extracción de datos tonales y curva estándar en Análisis Melódico del Habla (AMH)” (en este mismo número): un primer *Script* para la extracción de datos y un segundo para la obtención de la curva estándar. Se presenta la manera de instalarlos y su contenido completo.

**PALABRAS CLAVE:** entonación, análisis melódico, extracción automática de datos tonales

## **Resum**

Aquest document ofereix els *Scripts* per a Praat presentats al nostre article “Protocolo para la extracción de datos tonales y curva estándar en Análisis Melódico del Habla (AMH)” (en aquest mateix número): un primer *Script* per a l’extracció de dades i un segon per a l’obtenció de la corba estàndar. Presentem la manera d’instal·lar-los i el seu contingut complet.

**PARAULES CLAU:** entonació, anàlisi melòdic, extracció automàtica de dades tonals

## **Abstract**

This document provides the Praat *Scripts* presented in our article “Protocolo para la extracción de datos tonales y curva estándar en Análisis Melódico del Habla (AMH)” (in this issue): a first *Script* to extract data and a second for obtaining standard curve. Considers how to install and its entire contents.

**KEYWORDS:** intonation, melodic analysis, tonal data automatic extraction

## 1. Script 1 para la extracción de datos tonales

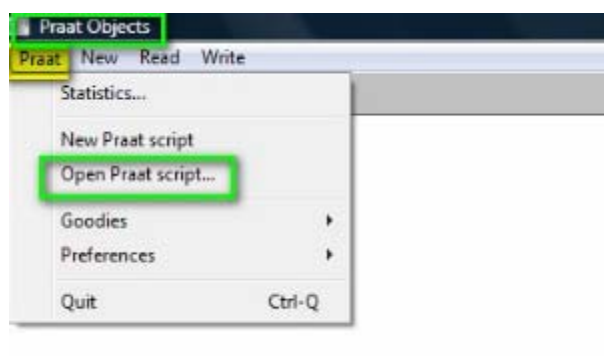
### 1.1. Instalación

En estos momentos, la instalación de los scripts es manual: nuestra intención inicial era automatizar este proceso, pero hemos encontrado problemas de ‘seguridad’ debido a las diferencias de configuración de los ordenadores de los usuarios del script. Por ello facilitamos aquí los pasos a seguir.

Tras descargar los programas y copiarlos en el directorio deseado, se ha de realizar una modificación de los directorios de trabajo dónde se ubicaran los ficheros que generan los programas, dos directorios – F0 y curva estándar– son comunes a los dos programas, el tercero, el de alertas sólo se utiliza en el primer script.

La modificación se puede realizar mediante un editor (bloc de notas, por ejemplo) o desde el propio PRAAT, opción que recomendamos ya que de esta forma en el mismo proceso realizamos la modificación y activamos en los menús de PRAAT los botones y comandos que permitan su utilización:

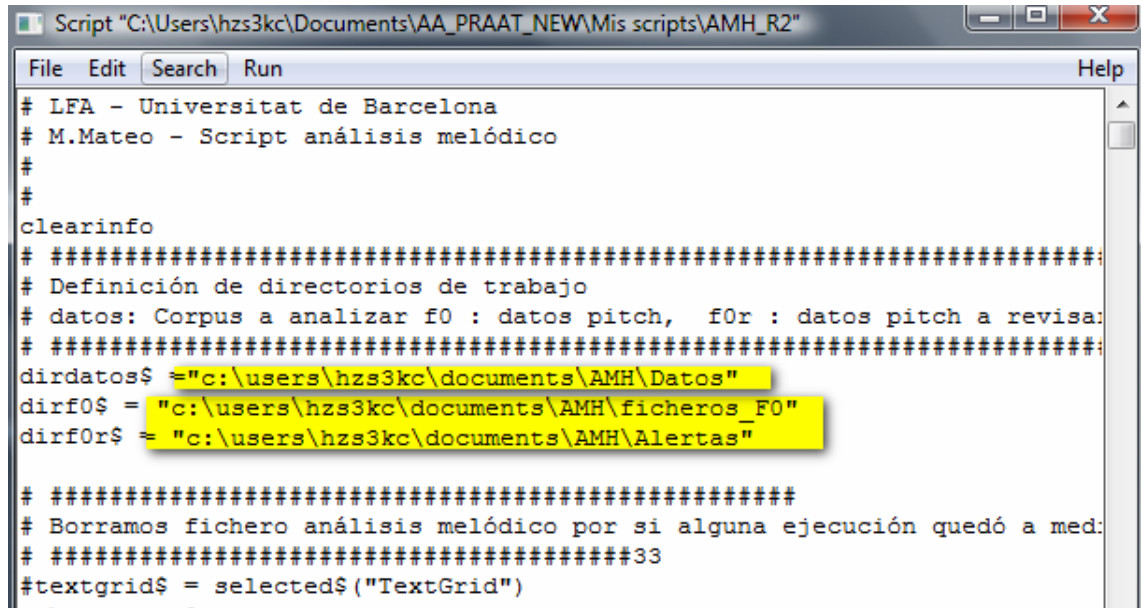
1. Ya en PRAAT, abrimos el script, desde la opción disponible en el primer menú de Praat Objects (v. figura 1)



**Figura 1** – PRAAT : instalación script 1, paso 1

2. En las líneas 10-12 (v. figura 2) indicaremos los directorios (enteros) donde queremos que se graben los dos ficheros que genera el programa (F0 y CE); estos directorios deben estar creados antes de procesarlo por primera vez. Una vez modificado, lo salvamos sin cerrar el fichero.

En el directorio “Datos” de nuestro ejemplo estará nuestro corpus: los sonidos y sus respectivos textgrid.



```

Script "C:\Users\hzs3kc\Documents\AA_PRAAT_NEW\Mis scripts\AMH_R2"
File Edit Search Run Help
# LFA - Universitat de Barcelona
# M.Mateo - Script análisis melódico
#
#
clearinfo
# #####
# Definición de directorios de trabajo
# datos: Corpus a analizar f0 : datos pitch, f0r : datos pitch a revisa
# #####
dirdatos$ = "c:\users\hzs3kc\documents\AMH\Datos"
dirf0$ = "c:\users\hzs3kc\documents\AMH\ficheros_F0"
dirf0r$ = "c:\users\hzs3kc\documents\AMH\Alertas"

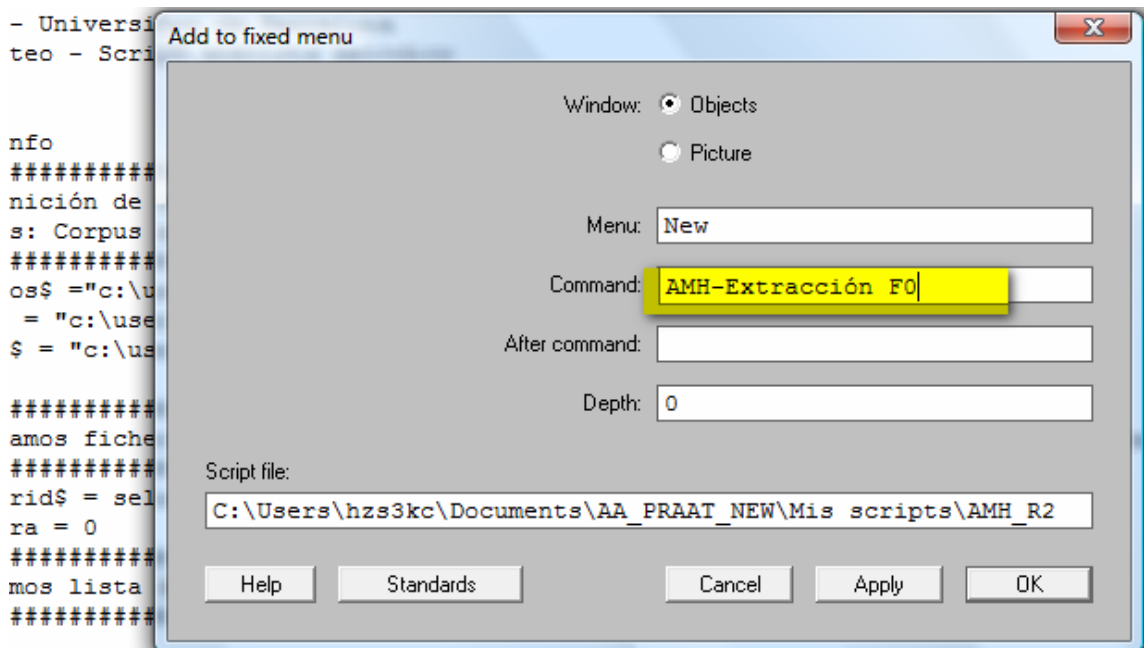
# #####
# Borramos fichero análisis melódico por si alguna ejecución quedó a med
# #####33
#textgrid$ = selected$("TextGrid")

```

**Figura 2** – PRAAT : instalación script 1, paso 2

3. El último paso es definir el menú de PRAAT dónde ha de aparecer la opción de extracción de datos y curva estándar (v. figura 3), para ello :

Lo añadimos a un menú “fijo” porque todo el proceso se realizará de forma “masiva” sobre todos los datos de nuestro corpus que estén ubicados en el directorio datos, sin tener que seleccionar ningún elemento en “Praat Objects”



**Figura 3** – PRAAT : instalación script 1, paso 3

**1.2.Contenido**

<u>Líneas</u>	<u>Descripción</u>
14-16	Definición de los directorios dónde se grabarán los ficheros que se generan en el script : los datos de frecuencia fundamental (F0), y datos a revisar por el investigador. Para la ejecución del script ya deben existir; se pueden cambiar para adaptarlos a la estructura de documentación.
22-23	Borrado de los ficheros por si algún análisis melódico anterior dejó datos en los directorios de trabajo.
25-44	Creación de la lista de ficheros a analizar. Se crea a partir del corpus (sonido + textgrid) que se ha dejado en el directorio “Datos”.
46-69	Extraemos del textgrid la información de si la voz es masculina o femenina
71-82	Generación del fichero con la extracción de los datos de pitch con parámetro de rango de hercios en función de si el enunciado es de voz masculina o femenina
84-96	Bucle principal de búsqueda de información de cada segmento tonal informado en el <i>TextGrid</i> .
97-144	Inicialización de variables de trabajo
148-200	Obtención de los distintos valores de pitch : principal, inicial, final, mínimo y máximo de cada segmento tonal; si no encontramos información asignamos un 0.
232-234	Asignación de variable de grabación en ficheros de pitch y de curva estándar cuando no tenemos información.
237-252	Control de valores extremos, mínimos y máximos, por encima de los límites definidos, para generar una alerta para su revisión manual posterior.
253-270	Asignación de límites de comparación para cada medida de pitch: 10% en el caso del castellano y catalán. Diferencias de pitch por debajo de este valor no se considerarán significativas.

- 274-293 Verificación de la relación de valores de cada segmento tonal : mínimo, máximo, inicial, final y su relación de variación respecto al 10% para decidir qué valores de pitch deberemos considerar para generar la curva estándar.
- 330-333 Llamada al procedimiento de grabación del fichero intermedio con los datos de pitch de cada segmento tonal del enunciado.
- 340 Fin bucle de búsqueda de información de cada segmento tonal
- 349-455 Procedimiento de grabación de los ficheros con el pitch y las alertas , si procede, de cada uno de los segmentos tonales. Los ficheros se graban siempre con el nombre del enunciado.
- 457-482 Procedimiento para buscar el primer valor de pitch informado en el segmento tonal si el inicial no lo estaba.
- 483-510 Procedimiento para buscar el último valor de pitch informado en el segmento tonal si el final no lo estaba.

```

01 # LFA - Universitat de Barcelona
02 # M.Mateo - Script análisis melódico
03 #
04 #
05 clearinfo
06 #
07 #####
08 #####
09 # Definición de directorios de trabajo
10 # datos: Corpus a analizar f0 : datos pitch, f0r : datos pitch a revisar
11 #
12 #####
13 #####
14 dirdatos$ ="c:\users\hzs3kc\documents\AM\Master\Datos"
15 dirf0$ = "c:\users\hzs3kc\documents\AM\Master\ficheros_F0"
16 dirf0r$ = "c:\users\hzs3kc\documents\AM\Master\ficheros_F0_Revisar"
17
18 # #####
19 # Borramos fichero análisis melódico por si alguna ejecución quedó a
20 # medias y grabamos cabecera
21 # #####33
22 #textgrid$ = selected$("TextGrid")
23 cabecera = 0
24 # #####
25 # Creamos lista de ficheros a procesar
26 # #####
27 #
28 Create Strings as file list... list 'dirdatos$'\*.TextGrid
29 numberOfFiles = Get number of strings
30 #
31 for ifile to numberOfFiles
32   select Strings list
33   sonido$ = Get string... ifile
34   Read from file... 'dirdatos$'\sonido$'
35 # objetotg$ = selected$("TextGrid")
36 textgrid$= selected$("TextGrid")
37 fichero$ = selected$("TextGrid")
38 call cabecera 'fichero$' 'dirf0$' 'dirf0r$'
39 # Generamos el fichero de pitch a partir de fichero original
40 # en función de si se ha informado que la voz era masculina (m) o
41 # femenina (f)
42 cabecera= cabecera + 1
43 select TextGrid 'textgrid$'
44 n = Get number of intervals... 1
45 # #####
46 # Buscamos información de si la voz es femenina o masculina
47 # #####

```

```

48 voz$ = ""
49 for i to n
50   silaba$ = Get label of interval... 1 i
51   if i = 1 or i = n
52     if silaba$ = "f" or silaba$ = "F"
53       voz$ = "f"
54     else
55       if silaba$ = "m" or silaba$ = "M"
56         voz$ = "m"
57       endif
58     endif
59   endif
60 endfor
61 #
62 #####
63 #####
64 # Validamos que está informado tipo de voz
65 #
66 #####
67 #####
68
69 if voz$=""
70   exit Textgrid no correcto, falta informar si la voz es masculina (m) o
71   femenina (f)
72 endif
73 # #####
74 # Creación fichero pitch
75 # #####
76 Read from file... 'dirdatos$\textgrid$.wav
77 select Sound 'textgrid$'
78 if voz$ = "f"
79   To Pitch (ac)... 0.02 90 15 yes 0.03 0.25 0.01 0.35 0.14 600
80 else
81   To Pitch (ac)... 0.02 40 15 yes 0.03 0.25 0.01 0.35 0.14 350
82 endif
83 pitch = selected ("Pitch")
84 select TextGrid 'textgrid$'
85 n = Get number of intervals... 1
86 # #####
87 # Bucle principal para cada sílaba informada (*)
88 # #####
89 for i to n
90   silaba$ = Get label of interval... 1 i
91   if (silaba$ <> "" and silaba$ <> "f" and silaba$ <> "m" and silaba$ <>
92   "F" and silaba$ <> "M")
93     ti = Get starting point... 1 i
94     ti$ = fixed$ (ti, 8)
95     tf = Get end point... 1 i

```

```

96     tf$ = fixed$ (tf, 8)
97     select pitch
98
99     # #####
100    # Inicializamos las variables
101    # #####
102     i$ = fixed$ (i, 0)
103     grabo= 0
104     grabo$ = fixed$ (grabo, 0)
105     control = 0
106     control$ = fixed$ (control, 0)
107     f0_i = 0
108     f0_i$ = fixed$ (f0_i, 0)
109     f0_ib = 0
110     f0_ib$ = fixed$ (f0_ib, 0)
111     f0_ibus = 0
112     f0_ibus$ = fixed$ (f0_ibus, 0)
113     f0_f = 0
114     f0_f$ = fixed$ (f0_f, 0)
115     f0_fb = 0
116     f0_fb$ = fixed$ (f0_fb, 0)
117     f0_fbus = 0
118     f0_fbus$ = fixed$ (f0_fbus, 0)
119     f0 = 0
120     f0$ = fixed$ (f0, 0)
121     min_f0 = 0
122     min_f0$ = fixed$ (min_f0, 0)
123     max_f0 = 0
124     max_f0$ = fixed$ (max_f0, 0)
125     fi090 = 0
126     fi090$ = fixed$ (fi090, 0)
127     fi110 = 0
128     fi110$ = fixed$ (fi110, 0)
129     ff090 = 0
130     ff090$ = fixed$ (fi090, 0)
131     ff110 = 0
132     ff110$ = fixed$ (ff110, 0)
133     fmin090 = 0
134     fmin090$ = fixed$ (fmin090, 0)
135     fmin110 = 0
136     fmin110$ = fixed$ (fmin110, 0)
137     fmax090 = 0
138     fmax090$ = fixed$ (fmax090, 0)
139     fmax110 = 0
140     fmax110$ = fixed$ (fmax110, 0)
141    # #####
142    # Variables para información de seguimiento (puts)
143    # en ejecución normal : N

```



```

144 # #####
145     imprimir$ = "N"
146     imprimirtodo$ = "N"
147     sigo$ = "N"
148 #
149 #
150 #
151 # Asignamos valor 0, si sistema no ha podido asignar inicial y final.
152 # Intentamos buscar el primero informado, si lo encontramos, lo
153 informamos.
154 # Convertimos los valores a enteros.
155 # Los valores se informarán manualmente.
156 #
157 #
158     f0 = Get mean... ti tf Hertz
159     f0$ = fixed$ (f0, 0)
160     if f0$ = "--undefined--"
161         f0$ = "0"
162     endif
163     min_f0 = Get minimum... ti tf Hertz Parabolic
164     min_f0$ = fixed$ (min_f0, 0)
165     if min_f0$ = "--undefined--"
166         min_f0$ = "0"
167     endif
168     max_f0 = Get maximum... ti tf Hertz Parabolic
169     max_f0$ = fixed$ (max_f0, 0)
170     if max_f0$ = "--undefined--"
171         max_f0$ = "0"
172     endif
173 # #####
174 # Si no hay información de pitch en el segmento tonal,
175 # no realizamos la búsqueda
176 # #####
177     if f0$ = "0" and min_f0$ = "0" and max_f0$ = "0"
178         f0_i$ = "0"
179         f0_f$ = "0"
180     else
181         f0_i = Get value at time... ti Hertz Linear
182         f0_i$ = fixed$ (f0_i, 0)
183         if f0_i$ = "--undefined--"
184             f0_i$ = "0"
185         call primero 'ti$' 'tf$' 'f0_i$'
186             f0_i = f0_ib
187             f0_i$ = fixed$ (f0_i, 0)
188             if f0_i$ = "--undefined--"
189                 f0_i$ = "0"
190             endif
191     endif

```

```

192     f0_f = Get value at time... tf Hertz Linear
193     f0_f$ = fixed$ (f0_f, 0)
194     if f0_f$ = "--undefined--"
195         f0_f$ = "0"
196         call ultimo 'ti$' 'tf$' 'f0_f$'
197         f0_f = f0_fb
198         f0_f$ = fixed$ (f0_f, 0)
199         if f0_f$ = "--undefined--"
200             f0_f$ = "0"
201         endif
202     endif
203 endif
204 #
205 # #####
206 # Valores para seguimiento, según variable imprimir, se pueden añadir
207 más condiciones
208 # #####
209 #
210 if imprimir$ = "S" and i=30
211     echo Valores :
212     printline grabacion : 'grabo$'
213     printline i : 'i$'
214     printline sílaba : 'silaba$'
215     printline inicio : 'ti$'
216     printline fin : 'tf$'
217     printline f0_i: 'f0_i$'
218     printline f0_ibuscado : 'f0_ibus$'
219     printline f0_fbuscado : 'f0_fbus$'
220     printline f0_f : 'f0_f$'
221     printline f0 : 'f0$'
222     printline min_f0 : 'min_f0$'
223     printline max_f0 : 'max_f0$'
224     printline fi090 : 'fi090$'
225     printline fi110 : 'fi110$'
226     printline ff090 : 'ff090$'
227     printline ff110 : 'ff110$'
228     printline fmin090 : 'fmin090$'
229     printline fmin110 : 'fmin110$'
230     printline fmax090 : 'fmax090$'
231     printline fmax110 : 'fmax110$'
232     endif
233 #
234 # Si alguno de los valores está a 0 (el sistema no ha podido
235 calcular)imprimiremos directamente.
236 # Los valores se informarán manualmente en el fichero resultado (***)
237 #
238     if f0_i$ = "0" or f0_f$ = "0" or f0$ = "0" or min_f0$ = "0" or max_f0$
239     = "0"

```

```

240     grabo$ = "8"
241     else
242     #
243     # (1) Validación de valores extremos
244     # 90 y 550 --> female
245     # 60 y 350 --> male
246     # para generar alerta en fichero de revisión
247     #
248     if (voz$ = "f" and ((f0_i > 550 or f0_f > 550 or f0 > 550 or min_f0 >
249     550 or max_f0 > 550) or (f0_i < 90 or f0_f < 90 or f0 < 90 or min_f0 < 90
250     or max_f0 < 90)))
251         control$ = "1"
252     endif
253     if (voz$ = "m" and ((f0_i > 350 or f0_f > 350 or f0 > 350 or min_f0 >
254     350 or max_f0 > 350) or (f0_i < 60 or f0_f < 60 or f0 < 60 or min_f0 < 60
255     or max_f0 < 60)))
256         control$ = "1"
257     endif
258     #
259     #
260     #
261     # (2) Asignamos los márgenes inferior y superior -actualmente 10%-
262     #
263         fi090 = f0_i * 0.90
264         fi090$ = fixed$ (fi090, 0)
265         fi110 = f0_i * 1.10
266         fi110$ = fixed$ (fi110, 0)
267         ff090 = f0_f * 0.90
268         ff090$ = fixed$ (ff090, 0)
269         ff110 = f0_f * 1.10
270         ff110$ = fixed$ (ff110, 0)
271         fmin090 = min_f0 * 0.90
272         fmin090$ = fixed$ (fmin090, 0)
273         fmin110 = min_f0 * 1.10
274         fmin110$ = fixed$ (fmin110, 0)
275         fmax090 = max_f0 * 0.90
276         fmax090$ = fixed$ (fmax090, 0)
277         fmax110 = max_f0 * 1.10
278         fmax110$ = fixed$ (fmax110, 0)
279     #
280     # (3) Realizamos las verificaciones para decidir qué valores grabaremos.
281     #
282     #
283
284         if (('f0_f$' >= 'fi090$' and 'f0_f$' <= 'fi110$') and (('max_f0$' >=
285     'fi090$') and ('max_f0$' <= 'fi110$')) and (('min_f0$' >= 'fi090$') and
286     ('min_f0$' <= 'fi110$'))
287             grabo$ = "1"

```

```

288         endif
289         if ('min_f0$' <= 'fi090$') and ('f0_f$' >= 'fmin090$') and ('f0_f$'
290 <= 'fmin110$')
291             grabo$ = "2"
292         endif
293         if ('max_f0$' >= 'fi110$') and ('f0_f$' >= 'fmax090$') and ('f0_f$' <=
294 'fmax110$')
295             grabo$ = "3"
296         endif
297         if (('f0_f$' <= 'fmin090$') or ('f0_f$' >= 'fmin110$')) and ('min_f0$'
298 <= 'fi090$')
299             grabo$ = "4"
300             control$="1"
301         endif
302         if ('max_f0$' >= 'fi110$') and (('f0_f$' <= 'fmax090$') or ('f0_f$' >=
303 'fmax110$'))
304             grabo$ = "5"
305             control$ = "1"
306         endif
307 #
308 #
309 # #####
310 # Valores para seguimiento, según variable imprimir, se pueden añadir
311 más condiciones
312 # #####
313 if imprimirtodo$ = "S" and i=10
314     echo Valores :
315     printline grabacion : 'grabo$'
316     printline control : 'control$'
317     printline i : 'i$'
318     printline sílaba : 'silaba$'
319     printline inicio : 'ti$'
320     printline fin : 'tf$'
321     printline f0_i : 'f0_i$'
322     printline f0_ibuscado : 'f0_ibus$'
323     printline f0_fbuscado : 'f0_fbus$'
324     printline f0_f : 'f0_f$'
325     printline f0 : 'f0$'
326     printline min_f0 : 'min_f0$'
327     printline max_f0 : 'max_f0$'
328     printline fi090 : 'fi090$'
329     printline fi110 : 'fi110$'
330     printline ff090 : 'ff090$'
331     printline ff110 : 'ff110$'
332     printline fmin090 : 'fmin090$'
333     printline fmin110 : 'fmin110$'
334     printline fmax090 : 'fmax090$'
335     printline fmax110 : 'fmax110$'

```

```

336     endif
337     # #####
338     # Fin verificación de lo que tenemos que grabar (***)
339     # #####
340     #
341     endif
342     #
343     # #####
344     # Llamada al procedimiento de grabación de cada sílaba en el fichero,
345     # después volvemos a seleccionar textgrid, porque se
346     # deselecciona en el proceso
347     # #####
348     #
349     call grabacion 'silaba$' 'f0_i$' 'f0_f$' 'f0$' 'min_f0$' 'max_f0$' 'grabo$'
350     'fichero$' 'dirf0$' 'dirf0r$'
351     select TextGrid 'textgrid$'
352     # #####
353     # Fin bucle de cada sílaba (***)
354     # #####
355     #
356     endif
357 endfor
358 endfor
359 #
360 #
361 #####
362 #####
363 #
364 # Grabación del fichero con todos los datos informados
365 #
366 # Valor 8 es cuando no se ha podido obtener algún dato y se graba a 0
367 # Además de grabar el fichero normal con el dato a 0, lo grabamos en el
368 directorio "Revisar",
369 # en el que quedarán todos los enunciados con análisis incompleto.
370 #
371 #
372 #####
373 #####
374 #
375 procedure grabacion .silaba$ .f0_i$ .f0_f$ .f0$ .min_f0$ .max_f0$ .grabo$
376 .fichero$ .dirf0$ .dirf0r$
377 #
378 #
379 #
380     if grabo$="8" or grabo$="0"
381         fileappend "'dirf0$\fichero$.txt"
382         ... 'silaba$',22,0,0'newline$'
383         fileappend "'dirf0r$\fichero$.txt"

```

```

384     ... 'silaba$',0,newline$'
385     endif
386     if grabo$="1" and control$ = "0"
387     fileappend "'dirf0$\'fichero$.txt"
388     ... 'silaba$',f0:0',0,0,newline$'
389     endif
390     if grabo$="2" and control$ = "0"
391     fileappend "'dirf0$\'fichero$.txt"
392     ... 'silaba$',f0_i$:0',0,0,newline$'
393     fileappend "'dirf0$\'fichero$.txt"
394     ... 'silaba$*',min_f0$:0',0,0,newline$'
395     endif
396     if grabo$="3" and control$ = "0"
397     fileappend "'dirf0$\'fichero$.txt"
398     ... 'silaba$',f0_i$:0',0,0,newline$'
399     fileappend "'dirf0$\'fichero$.txt"
400     ... 'silaba$*',max_f0$:0',0,0,newline$'
401     endif
402     if grabo$="4" and control$ = "0"
403     fileappend "'dirf0$\'fichero$.txt"
404     ... 'silaba$',f0_i$:0',0,0,newline$'
405     fileappend "'dirf0$\'fichero$.txt"
406     ... 'silaba$*',min_f0$:0',0,0,newline$'
407     fileappend "'dirf0$\'fichero$.txt"
408     ... 'silaba$**',f0_f$:0',0,0,newline$'
409     endif
410     if grabo$="5" and control$ = "0"
411     fileappend "'dirf0$\'fichero$.txt"
412     ... 'silaba$',f0_i$:0',0,0,newline$'
413     fileappend "'dirf0$\'fichero$.txt"
414     ... 'silaba$*',max_f0$:0',0,0,newline$'
415     fileappend "'dirf0$\'fichero$.txt"
416     ... 'silaba$**',f0_f$:0',0,0,newline$'
417     endif
418 #
419 #####
420 #####
421 # Grabamos fila en el fichero de datos a revisar si está activada la variable
422 # de control de valores extremos.
423 #
424 #####
425 #####
426     if grabo$="1" and control$ = "1"
427     fileappend "'dirf0r$\'fichero$.txt"
428     ... 'silaba$',999,newline$'
429     fileappend "'dirf0$\'fichero$.txt"
430     ... 'silaba$',f0:0',0,0,newline$'
431     endif

```

```

432   if grabo$="2" and control$ = "1"
433     fileappend "'dirf0r$\'fichero$.txt"
434     ... 'silaba$',999'newline$'
435     fileappend "'dirf0$\'fichero$.txt"
436     ... 'silaba$',f0_i$:0',0,0'newline$'
437     fileappend "'dirf0$\'fichero$.txt"
438     ... 'silaba$*',min_f0$:0',0,0'newline$'
439     endif
440   if grabo$="3" and control$ = "1"
441     fileappend "'dirf0r$\'fichero$.txt"
442     ... 'silaba$',999'newline$'
443     fileappend "'dirf0$\'fichero$.txt"
444     ... 'silaba$',f0_i$:0',0,0'newline$'
445     fileappend "'dirf0$\'fichero$.txt"
446     ... 'silaba$*',max_f0$:0',0,0'newline$'
447     endif
448   if grabo$="4" and control$ = "1"
449     fileappend "'dirf0r$\'fichero$.txt"
450     ... 'silaba$',3'newline$'
451     fileappend "'dirf0$\'fichero$.txt"
452     ... 'silaba$',f0_i$:0',0,0'newline$'
453     fileappend "'dirf0$\'fichero$.txt"
454     ... 'silaba$*',min_f0$:0',0,0'newline$'
455     fileappend "'dirf0$\'fichero$.txt"
456     ... 'silaba$***,f0_f$:0',0,0'newline$'
457     endif
458   if grabo$="5" and control$ = "1"
459     fileappend "'dirf0r$\'fichero$.txt"
460     ... 'silaba$',3'newline$'
461     fileappend "'dirf0$\'fichero$.txt"
462     ... 'silaba$',f0_i$:0',0,0'newline$'
463     fileappend "'dirf0$\'fichero$.txt"
464     ... 'silaba$*',max_f0$:0',0,0'newline$'
465     fileappend "'dirf0$\'fichero$.txt"
466     ... 'silaba$***,f0_f$:0',0,0'newline$'
467     endif
468
469   endproc
470   #
471   # #####
472   # Buscamos primer valor informado del segmento tonal
473   # #####
474   procedure primero .ti$ .tf$ .f0_i$
475     if sigo$ = "S"
476       echo Valores
477       printline entro búsqueda primero
478     endif
479   #

```

```

480 #
481 #
482   numberOfTimeSteps = ('tf$' - 'ti$') / 0.001
483   step = 1
484   repeat
485     tmin = 'ti$' + (step - 1) * 0.001
486     tmax = tmin + 0.001
487     f0_ib= Get mean... tmin tmax Hertz
488     f0_ibus$ = fixed$ (f0_ib, 0)
489     if f0_ibus$ = "--undefined--"
490       f0_ibus$ = "0"
491     endif
492     step = step + 1
493     until ('f0_ibus$' > 0) or (step = numberOfTimeSteps)
494 #
495 #
496 endproc
497 #
498 #####
499 # Buscamos último valor informado del segmento tonal
500 #
501 #####
502 #
503 procedure ultimo .ti$ .tf$ .f0_f$
504 if sigo$ = "Z"
505 echo Valores
506 printline entro búsqueda último
507 endif
508
509   numberOfTimeSteps = ('tf$' - 'ti$') / 0.001
510   step = 1
511   repeat
512     tmin = 'tf$' - (step * 0.001)
513     tmax = tmin + 0.001
514     f0_fb= Get mean... tmin tmax Hertz
515     f0_fbus$ = fixed$ (f0_fb, 0)
516     if f0_fbus$ = "--undefined--"
517       f0_fbus$ = "0"
518     endif
519     step = step + 1
520     until ('f0_fbus$' > 0) or (step = numberOfTimeSteps)
521   endproc
522 procedure cabecera .fichero$ .dirf0$ .dirf0r$
523 deleteFile ("dirf0$\fichero$.txt")
524 deleteFile ("dirf0r$\fichero$.txt")
525   fileappend "dirf0$\fichero$.txt"
526     ... Segmento,HZ,Perc,CE'newline$'
527 Endproc

```



## 2. Script 2 para la estandarización de los datos

### 2.1. Instalación

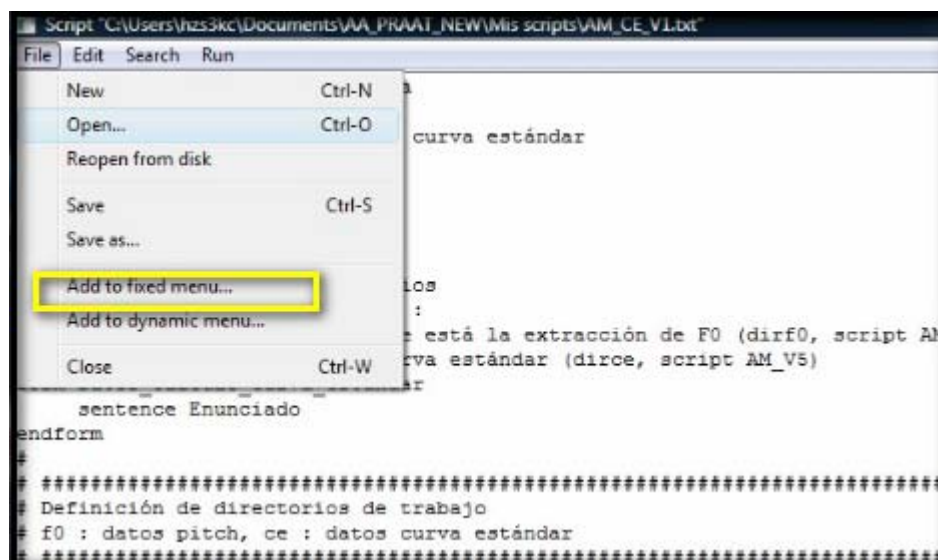
Con el segundo script, seguiremos los pasos (1) y (2) descritos anteriormente, en el segundo paso, sólo son dos los directorios a modificar (v. figuras 4 a 6).

```
# Miguel Mateo
# Análisis melódico - obtención curva estándar
#
#
clearinfo
#
# *****:
# Definición de directorios de trabajo
# f0 : datos pitch, ce : datos curva estándar
# *****:
dirf0$ = "c:\users\hzs3kc\Documents\AMH\ficheros_F0"
dirce$ = "c:\users\hzs3kc\Documents\AMH\Curva Estándar"
tipof$ = "txt"
#
```

**Figura 4** – PRAAT : instalación script 2, paso 2

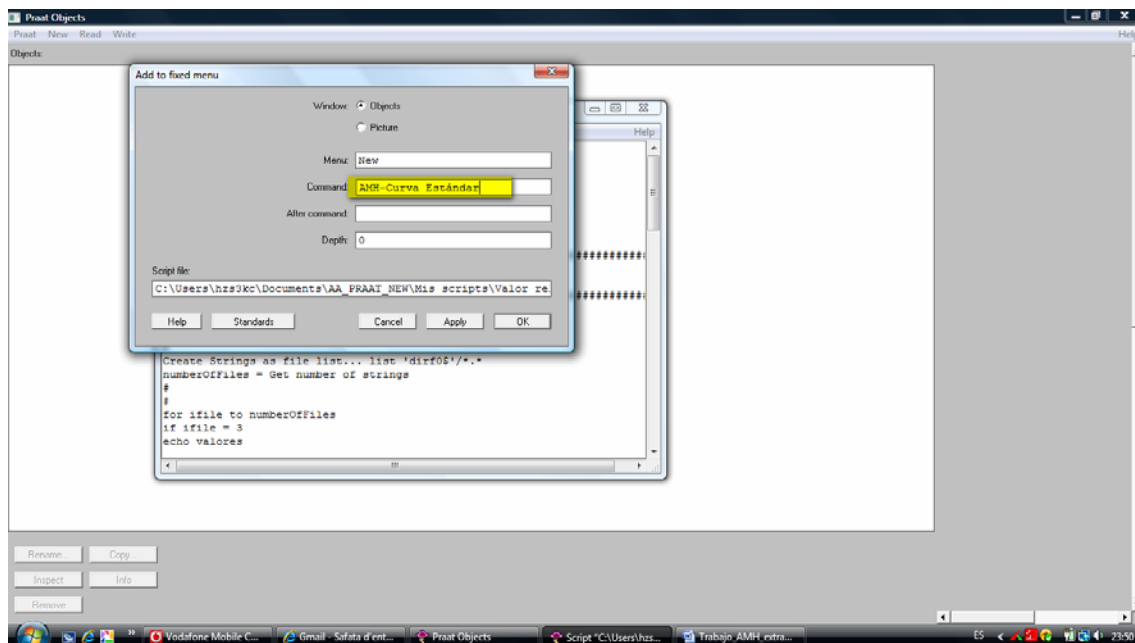
El primer directorio ya lo definimos en el primer *script*, ha de coincidir ya que contiene los datos de extracción de F0, en el segundo se dejarán los ficheros con los valores estandarizados.

A continuación seleccionamos en “File” la opción de menú fijo:



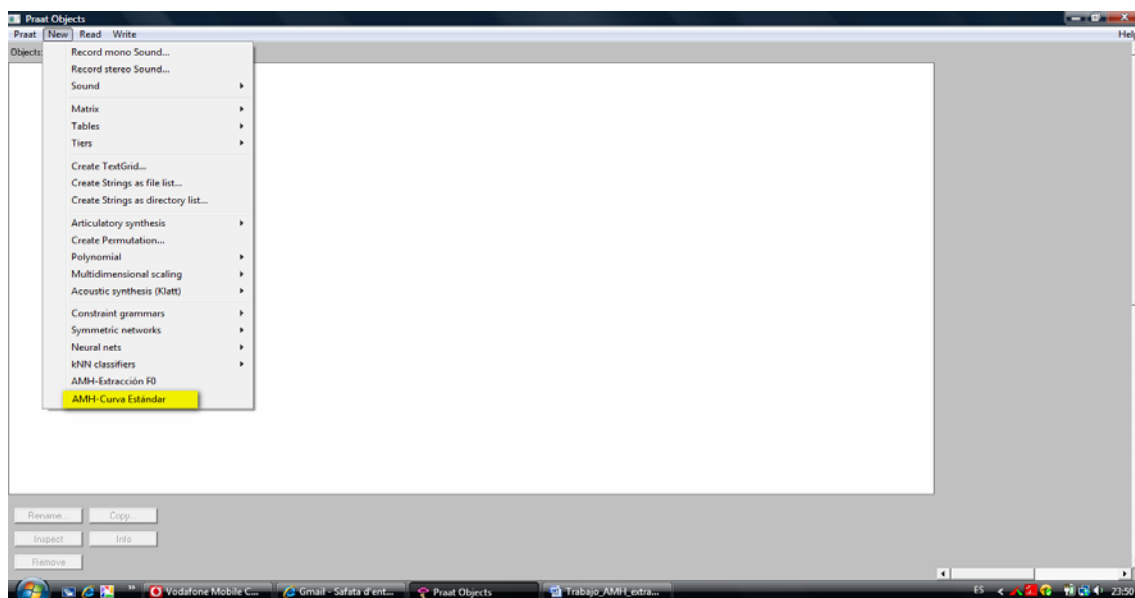
**Figura 5** – PRAAT : instalación script 2, paso3

Que nos mostrará esta pantalla, dejaremos las opciones que aparecen marcadas por defecto y en el apartado “Command” indicaremos un nombre para la opción que estamos añadiendo, por ejemplo “Curva estándar”.



**Figura 6** – PRAAT : instalación script 2, paso3

Una vez pulsamos el botón “OK”, ya nos sale la opción en el menú que elegimos (“New”), ya tenemos en el menú las opciones para procesar nuestros scripts v. (figura 7)



**Figura 7** – PRAAT : menú con las opciones de proceso del script

## 2.2. Contenido

Este segundo script realiza el proceso de estandarización de los datos tonales, según el procedimiento AMH<sup>1</sup>.

Asigna 100 a los primeros valores, tanto relativo como de curva estándar, y después aplica una sencilla regla de tres para calcular los valores estandarizados del resto de segmentos tonales de cada enunciado.

```

01 # LFA - Universitat de Barcelona
02 # Miguel Mateo
03 # Análisis melódico - obtención curva estándar
04 #
05 #
06 #
07 clearinfo
08 #
09 #
10 #####
11 #####
12 # Definición de directorios de trabajo
13 # f0 : datos pitch, ce : datos curva estándar
14 #
15 #####
16 #####
17 dirf0$ = "c:\users\hzs3kc\Documents\AM\Master\ficheros_F0"
18 dirce$ = "c:\users\hzs3kc\Documents\AM\Master\curva_F0"
19 tipof$ = "txt"
20 #
21 Create Strings as file list... list 'dirf0$'/*.*
22 numberOfFiles = Get number of strings
23 #
24 #
25 for ifile to numberOfFiles
26 if ifile = 3
27 echo valores
28 printline fichero entrada : 'fichero$'
29 printline fichero salida : 'fichero_sal$'
30 printline dirce : 'dirce$'
31 endif
32 # #####
33 # Inicialización de variables
34 i = 0
35 hz = 0
36 perc = 0
37 percent = 0

```

<sup>1</sup> Se puede encontrar una explicación en el artículo “Protocolo para la extracción de datos tonales y curva estándar en Análisis Melódico del Habla (AMH)” en este mismo volumen.

```

38 ce = 0
39 ceant = 0
40 #
41 select Strings list
42 fichero$ = Get string... ifile
43 Read Table from comma-separated file... 'dirf0$\'fichero$'
44 numberOfRows = Get number of rows
45 call calculo 'tipof$' 'dirce$' 'dirf0$' 'fichero$'
46 #
47 #
48 #
49 endfor
50 #
51 procedure calculo .tipof$ .dirce$ .dirf0$ .fichero$
52 # Bucle de cálculo de % y curva estándar
53 #
54 for i to numberOfRows
55 hz = Get value... i HZ
56 hz$ = fixed$(hz, 0)
57 perc = Get value... i Perc
58 perc$ = fixed$(perc, 1)
59 ce = Get value... i CE
60 ce$ = fixed$(ce, 0)
61 if i = 1
62 perc = 100
63 perc$ = fixed$(perc, 1)
64 ce = 100
65 ce$ = fixed$(ce, 0)
66 Set numeric value... i Perc 'perc$'
67 Set numeric value... i CE 'ce$'
68 percent$ = fixed$(perc, 1)
69 ceant$ = fixed$(ce, 0)
70 hzant$ = fixed$(hz, 0)
71 else
72 perc = (('hz$'/'hzant$') * 100) - (100)
73 perc$ = fixed$(perc, 0)
74 ce = ('perc$'*'ceant$'/100) + 'ceant$'
75 ce$ = fixed$(ce, 0)
76 Set numeric value... i Perc 'perc$'
77 Set numeric value... i CE 'ce$'
78
79 percent$ = fixed$(perc, 1)
80 ceant$ = fixed$(ce, 0)
81 hzant$ = fixed$(hz, 0)
82
83
84 endif
85 endfor

```

86	fichero_sal\$ = selected\$("Table")
87	Write to table file... 'dirce\$\fichero_sal\$.txt
88	Endproc