

**RECURSOS PARA EL ANÁLISIS ACÚSTICO
DE LA MELODÍA DEL HABLA EN PRAAT**

DOMINGO ROMÁN MONTES DE OCA
Pontificia Universidad Católica de Chile
dromanm@uc.cl

RESUMEN

En este documento se muestran las principales técnicas de análisis para realizar investigaciones sobre la melodía del habla en Praat. Se revisan tanto las posibilidades manuales como las automáticas que se encuentran disponibles en el programa. Finalmente se presentan dos *scripts* que pueden ser de utilidad para todo investigador.

Palabras clave: *F0, frecuencia fundamental, tono, Praat, análisis acústico, melodía del habla.*

ABSTRACT

This paper presents the main analysis techniques which have been used in the research carried out on speech melody in Praat. Manual as well as automatic possibilities available in the program are revisited. Finally, two scripts which can be of use to any researcher are introduced.

Keywords: *F0, fundamental frequency, pitch, Praat, acoustic analysis, speech melody.*

1. INTRODUCCIÓN

En el presente documento se proporcionan las especificaciones básicas para obtener datos correspondientes a la frecuencia fundamental (o F0) de una señal de habla con el programa Praat (Boersma y Weenink 2008). Se presupone que el lector tiene alguna experiencia con el programa: que sabe crear objetos de tipo *Pitch* y *TextGrid* y que es capaz de abrir y ejecutar *scripts*. Por razones de espacio, no se revisarán cuestiones vinculadas con el panel de gráficos del programa.

Se pasará revista a los métodos más corrientes para obtener los valores en Hz de la frecuencia fundamental. Se mostrarán algunas maneras para hacer síntesis y manipulaciones de la F0 —técnicas especialmente interesantes para cuestiones perceptivas— y una forma para automatizar la obtención de datos.

La operación, relativamente sencilla para el ojo humano, de encontrar ciclos en un oscilograma resulta relativamente difícil para un algoritmo que funciona en un ordenador, ya que un archivo de audio consiste solamente en dos columnas numéricas (número de la muestra y valor de amplitud). En esa lista de datos el programa tiene que encontrar ciclos regulares a partir de un análisis realizado en ventanas temporales.

Por la razón expuesta, normalmente hay diferencias (mínimas, la mayoría de las veces) en los valores de F0 que, para un mismo punto de una señal, se obtienen en programas distintos. De ahí la costumbre de mantener constante el programa, e incluso a veces también la versión, durante una investigación. Una reflexión sobre este asunto se puede leer en Ladefoged (2003: 75-86) en que compara PCquirer/Macquirer¹ y Praat. A su juicio, cada uno de los algoritmos tiene ventajas y desventajas. En la práctica, declara un empate; pero hace constar que con distintos procedimientos los resultados son diferentes. Esto produce una normal inquietud en todo científico, que quiere datos lo más realistas y exactos posibles. Afortunadamente, las diferencias, como se ha mencionado, son mínimas.

El lector debe tener en cuenta que un texto como este se puede ver afectado en el tiempo por los cambios en el programa. En particular, Praat es actualizado periódicamente y, cada cierto tiempo, se introducen modificaciones que pueden obligar a reinterpretar en alguna medida lo que aquí se señala.

¹ SCICON R&D, Inc.

2. CÓMO OBTENER EL VALOR DE LA FRECUENCIA FUNDAMENTAL

2.1. En la ventana *Edit*

Si se tiene un objeto de tipo *Sound* seleccionado («objeto» no es lo mismo que «archivo»), y activamos el botón *Edit*, se abre la ventana de edición de la señal, con el oscilograma, el espectrograma y otros análisis. En esta ventana existen dos posibilidades, entre otras, para obtener los valores de la frecuencia fundamental: una manual y otra automática.

2.1.1. Método manual

Este método funciona por medición del ciclo. Esta manera de calcular la frecuencia fundamental consiste en acercarse a la señal hasta que los ciclos sean claramente identificables y seleccionar, con el cursor, uno de ellos para medir su duración. Dado que la F0 es la inversa del período, bastará con dividir $1 / \text{duración del ciclo}$ y el resultado será la F0.

Al momento de seleccionar una parte de la señal, automáticamente el programa indica, en la parte superior de la selección, la duración (en segundos); con las selecciones pequeñas el programa muestra el valor de la división de $1 / \text{la selección}$ (cualquiera que ella sea). El resultado se muestra entre paréntesis, en la parte superior de la ventana de edición. Cuando se selecciona un ciclo, el resultado de esta división es el valor de la frecuencia fundamental. Así se observa en la figura 1.

Para realizar esta operación con mayor precisión, es posible ajustar el inicio y el final de la selección a los puntos en que la señal cruza por el valor de cero amplitud con la opción del menú *Select > Move start of selection to nearest zero crossing* y con *Move end of selection to nearest zero crossing*.

Este método es lento pero eficaz. Muchas veces fallan los algoritmos que permiten obtener valores de manera muy rápida en señales largas. En esos casos, el método al que se recurre es este: medir la F0 en el oscilograma; después de todo, esa duración debe correlacionarse con la del período del tono laríngeo, es decir, con la frecuencia a la que vibran las cuerdas vocales.

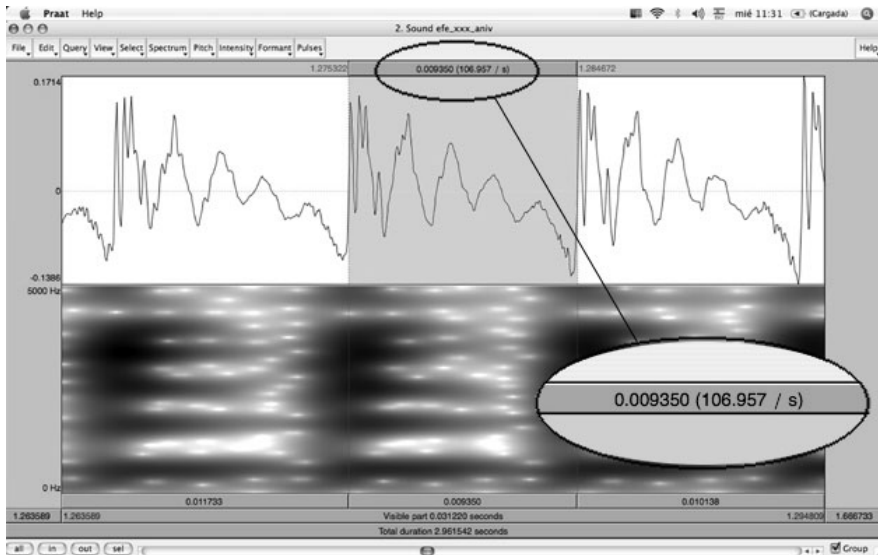


Figura 1. Selección de un ciclo en la ventana Edit; los límites están ajustados a los ceros más cercanos. Los números amplificados indican la duración y la frecuencia fundamental.

La ventana de edición también permite mostrar los valores de F0 cuando la opción *Pitch > Show pitch* está activa. Esta posibilidad se examina en el apartado 2.1.2.

2.1.2. Método automático en la ventana de edición

El método recomendado es el método de autocorrelación. Básicamente, este algoritmo funciona con una ventana de análisis que se desplaza por el archivo de audio buscando correlaciones altas entre los valores de la señal para identificar los ciclos. Para descripciones técnicas, véase Coleman (2005: 90-95), Ladefoged (1996: 148-151 y 2003: 75-83) entre otros. El método funciona bastante bien, salvo en aquellos casos en que el desajuste entre la ventana de análisis y la duración del ciclo hace que el programa encuentre dos ciclos donde en realidad hay uno o, el caso inverso, en el que el algoritmo encuentra un ciclo donde en realidad hay dos. Estos errores constituyen información distorsionada ya que el programa detecta valores del doble, o de la mitad, de lo que corresponde.

Al activar la opción *Pitch > Show pitch* del menú superior, se despliega una curva celeste con los valores de frecuencia fundamental que el método de autocorrelación detecta. En el lado derecho de la ventana se indican los valores de F0 en el punto en que está situado el cursor. Todo esto se observa en la figura 2.

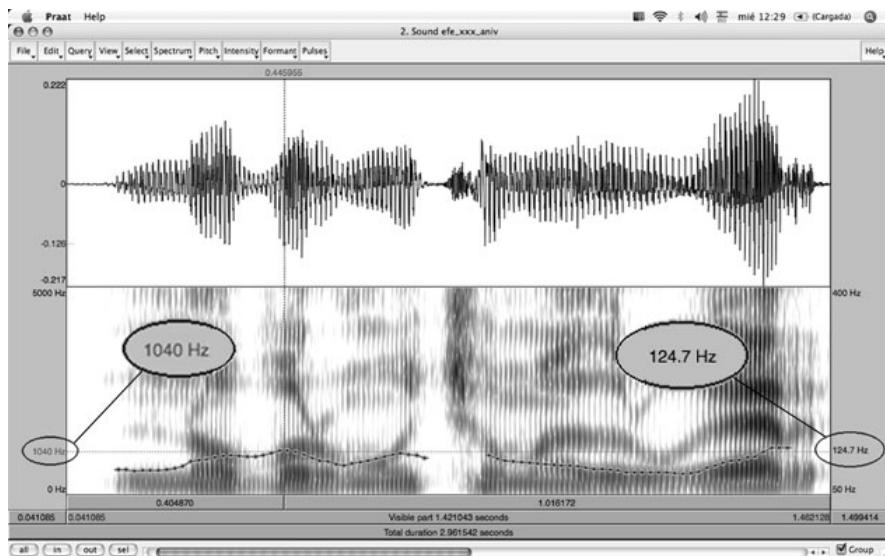


Figura 2. Ventana de edición en la que se ve la curva de pitch superpuesta al espectrograma. El valor que se muestra a la derecha corresponde a F0 en el punto señalado por el cursor. Los valores de la izquierda corresponden a la frecuencia del espectrograma.

En el menú superior *Pitch*, está disponible la opción *Pitch setting*. Con esta, se despliega un formulario que permite establecer la gama de Hz entre los cuales el algoritmo encuentra los valores mínimos y máximos (entre 75 Hz y 500 Hz, por defecto); las unidades (Hz, st, mel, ERB); el método de análisis (autocorrelación, por defecto) y la forma usada para dibujar la curva. Todo esto afecta a la información de *pitch* que parecerá en la ventana de edición. Cuando el cursor esté situado en un punto en que el algoritmo detecte periodicidad, se desplegará, al lado derecho de la ventana, el valor de la frecuencia fundamental correspondiente. La relevancia de establecer parámetros adecuados es que minimiza los errores posibles, como los mencionados anteriormente.

2.2. EL OBJETO PITCH

Para cada objeto de sonido, se puede crear un objeto de tipo *Pitch* con el botón *Periodicity- > To Pitch...* Con esto se despliega un formulario con tres campos: *Time step (s)*, *Pitch floor (Hz)* y *Pitch ceiling (Hz)*. De estos, los dos últimos son los que sirven para determinar los valores mínimos y máximos posibles de la frecuencia fundamental en esa señal. Fijados estos parámetros, con *Ok* se crea el objeto y este aparecerá en el panel de objetos. El objeto recién creado se puede editar, con *Edit*, en una ventana similar a la de la figura 3.

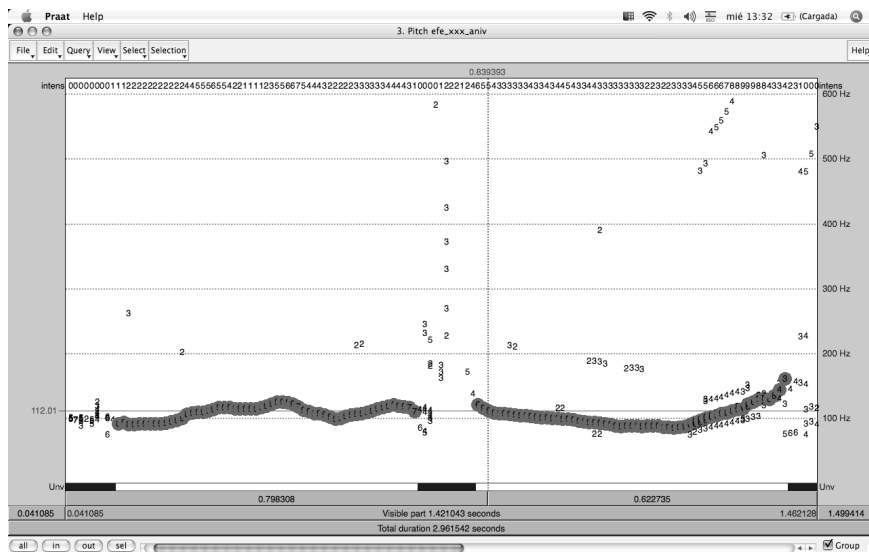


Figura 3. Ventana de edición del objeto Pitch.

En esta ventana se observan en color magenta los valores de F0 considerados por el algoritmo como los más probables. Los números distribuidos por la zona blanca de la pantalla muestran otros «valores candidatos» posibles. De hecho, el investigador puede situar el cursor sobre cualquiera de esos números y al clickear se transforman en puntos de color. También aparece una franja con sectores azules en la parte inferior, que indican zonas de aperiodicidad. Al marcar con el cursor en una zona blanca (es decir, con periodicidad) de la misma franja, esta se convierte

en zona azul (es decir, de aperiodicidad) o sea, sin valores de F0 (simultáneamente, se borran los puntos de *pitch* detectados).

Este método es el que el programa dispone para «editar» el *pitch*, o sea, para corregir (dentro de ciertos márgenes) los valores encontrados automáticamente. También hay opciones en el menú superior de esta ventana para cambiar los valores por intervalos de octava o de quinta, siempre y cuando haya «valores candidatos» posibles.

Se puede abrir la ventana de edición de un objeto de tipo *Sound* y, simultáneamente, la ventana de edición del objeto *Pitch* correspondiente y coordinarlas mediante la activación del cuadro *Group*, localizado en la esquina inferior derecha de cada ventana. Con esta opción, la posición del cursor en una ventana se replica sincrónicamente en la otra. Esta posibilidad es útil cuando el investigador necesita saber un valor de F0 correspondiente a una señal cuyo *pitch* ha sido corregido.

3. OTRAS OPERACIONES

Hay un número de opciones disponibles para un objeto de tipo *Pitch* que conviene manejar (para analizar o para preparar presentaciones). Una de ellas es la opción *Synthesize*, que permite crear objetos de tipo *Sound* a partir de un objeto *Pitch*. Estos objetos son síntesis ya sea en forma de senoide, o bien como pulsos, o bien como un tono parecido al glotal. En todos los casos, el resultado conserva la información frecuencial de la curva de *pitch* de la que se parte.

La opción *Convert > Smooth* permite modificar levemente la curva de F0 recalculando cada uno de los valores de tal manera que se pierden las diferencias mínimas y se aproximan los valores de los puntos contiguos. El resultado es una curva más pareja que la original, con valores diferentes, pero idéntica en términos perceptivos. Es una curva más limpia que puede ser útil en presentaciones en que no son relevantes ciertos detalles.

Obsérvese la figura 4 que muestra dos curvas de la misma señal; la segunda fue sometida al proceso *Smooth*.

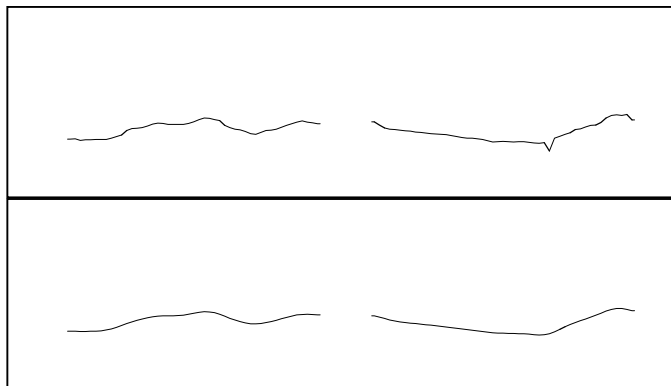


Figura 4. Dos gráficos de la misma señal. La del cuadro inferior ha sido sometida al proceso *smooth* con las opciones *por defecto*.

4. SÍNTESIS

El programa permite manipular los archivos de sonido de tal manera que es posible cambiar la frecuencia fundamental sin alterar otros valores frecuenciales. Aquí se mostrarán dos maneras que pueden tener aplicación en estudios perceptivos. Primero se muestra una forma de convertir la melodía de una emisión en una frecuencia plana y, luego, un procedimiento de manipulación en el que se pueden cambiar, prácticamente a voluntad, todos los valores del tono.

4.1. Cambio de características

Para un objeto de sonido, la opción *Convert-* permite la posibilidad *Change gender...* Con esta última, aparece un formulario en el que hay que completar los valores mínimos y máximos de *pitch* de la señal y una serie de valores que se podrían cambiar para la señal resultante. Dado el tema de esta presentación, solo se expondrá cómo alterar el *pitch*, pero, también se puede cambiar la duración y los valores de los formantes. Para que el resultado de la manipulación sea la señal con un tono plano, en los dos primeros campos se deben dar la gama en la que el programa debe buscar los valores de la frecuencia fundamental. Los otros campos se deben llenar de la siguiente manera: en *Formant shift ratio*, se debe indicar «1.0» para que no cambien los valores de los formantes. En *New pitch median (Hz)*

se debe llenar con «0.0» para que conserve el promedio del *pitch* de la señal original; en *Pitch range factor* se debe escribir «0.0» para que «monotonice» la señal. Si en *Duration* se conserva el valor «1.0», la señal no cambia en el aspecto temporal. El resultado es un interesante audio con todas las características del original, salvo el tono, que ha sido monotonizado en el valor promedio de la señal original.

4.2. Manipulación

Si se selecciona un objeto de sonido y se elige la opción *To Manipulation...* se despliega el mismo formulario que aparece cuando se crea el objeto *Pitch*. Se llenan los campos con los valores correspondientes y se crea el objeto *Manipulation*. Al editarlo, aparece una ventana que incluye una zona de *pitch* y una de duración, como en la figura 5: los dos aspectos se pueden modificar. En esta ocasión, los puntos con valores de *pitch* aparecen en color verde y se pueden cambiar de posición (vertical y horizontal) seleccionándolos y arrastrándolos manualmente, o bien con las posibilidades *Shift pitch frequency* y *Multiply shift frequency*, del menú superior *Pitch*.

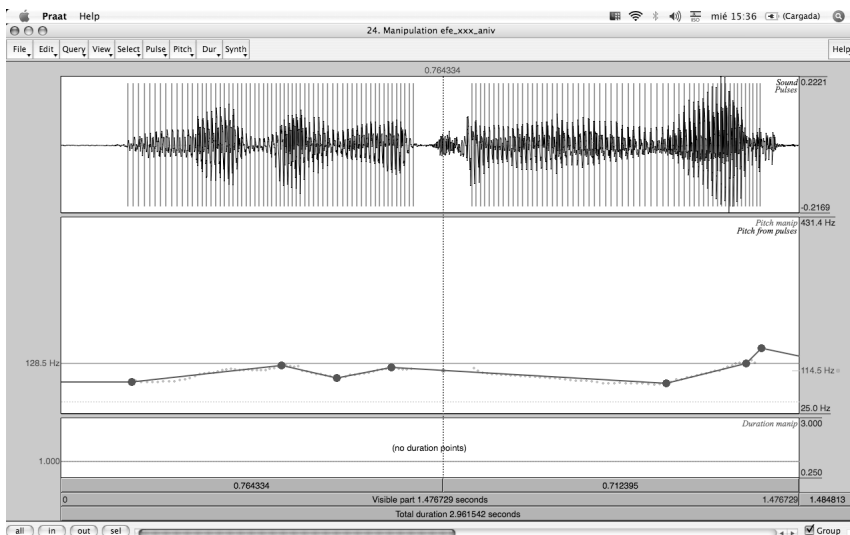


Figura 5. Ventana de manipulación con una curva de *pitch* estilizada automáticamente. Las líneas verticales en el oscilograma señalan los pulsos detectados.

En esta misma ventana se puede estilizar la curva de *pitch*. Para hacerlo automáticamente, los comandos correspondientes son *Stylice pitch (2 st)* y *Stylice pitch...*. El programa une los valores de F0 mediante líneas rectas sin considerar diferencias menores al umbral de 2 st, o lo que el usuario determine.

Esta estilización también se puede hacer manualmente ya que cada punto se puede mover a voluntad con el cursor, o con instrucciones dadas a partir del menú superior. Con el mismo menú, se pueden agregar puntos y eliminar los que estén seleccionados. Una forma de hacerlo manualmente es seleccionar la señal completa y eliminar todos los puntos de *pitch*; luego, insertar algunos en los lugares y a la altura frecuencial que el investigador considere pertinente.

Un audio modificado se puede convertir en un objeto de tipo *Sound*, con el comando *Publish resynthesis* (en el menú *File*), que se puede guardar como archivo de sonido en el ordenador.

5. SCRIPTS

Si el usuario conoce la técnica de etiquetar señales mediante los *TextGrids*, la obtención de datos se puede automatizar mediante el lenguaje de programación incorporado. Esta es una de las características de este programa que lo distingue de otros. Las rutinas creadas por el investigador se pueden guardar e incluso se pueden agregar a la interfaz del programa. Estas subrutinas se abren y se ejecutan desde el menú específico *Open Praat script...*

Si se tiene un archivo de sonido, con un *TextGrid* asociado con un estrato de tipo puntual, en el que hay marcas donde el investigador desea obtener los valores de la frecuencia fundamental, es posible escribir instrucciones para que el programa encuentre esos valores y los vuelque a una pantalla desde donde los podamos copiar a una planilla electrónica.

En la figura 6 se muestra un *TextGrid* con un estrato de intervalo en el que el investigador ha puesto número a las sílabas encontradas y en el estrato 2 ha puesto marcas en lo que considera que son los centros de cada vocal. Ha etiquetado también cada punto con la vocal correspondiente.

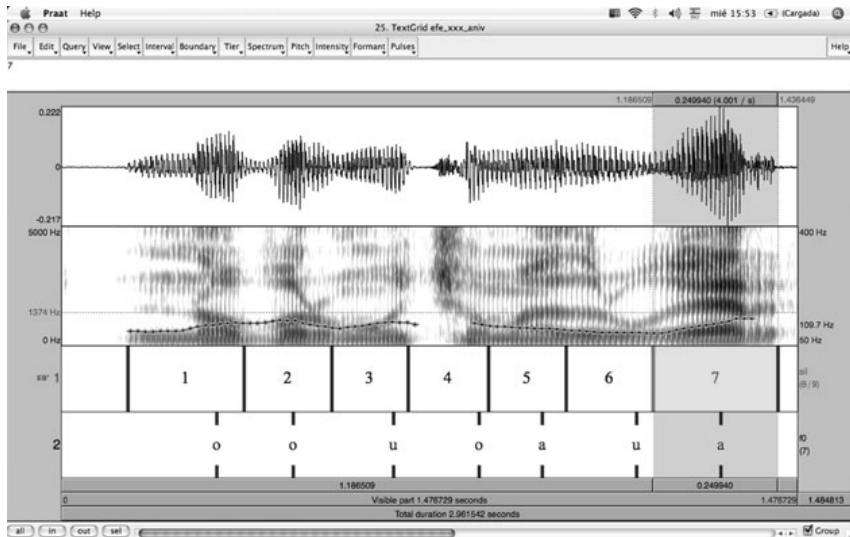


Figura 6. Señal y TextGrid correspondiente con dos estratos.

Lo que el investigador necesita es obtener las etiquetas vocálicas y el valor de F0 correspondiente. Se mostrará a continuación una manera automatizada para tener una lista de dos columnas en que en la primera aparezca la etiqueta y en la segunda el valor de F0 correspondiente. Por cierto, para beneficio de la exposición, se presenta este requerimiento relativamente sencillo, pero las rutinas pueden ejecutar tareas bastante más complejas. Veremos los pasos que tiene que hacer una subrutina para este propósito y luego mostraremos el *script* correspondiente. Para esta demostración, se asume que el investigador ha creado el *TextGrid* y el objeto *Pitch* previamente.

La subrutina deberá seleccionar el *TextGrid* y en este objeto deberá seleccionar, a su vez, el estrato 2; en este estrato, obtendrá los valores temporales en que está cada marca para conseguir después, en esos mismos momentos, los valores frecuenciales. Después leerá y almacenará la etiqueta de cada marca. Finalmente vaciará a una pantalla de texto simple la etiqueta y el valor frecuencial correspondiente. El estrato 1, en este ejemplo, se usa para etiquetar sílabas pero no para obtener datos. En términos de programación, las tareas presentadas se hacen con el siguiente *script*:

```

clearinfo
este$ = selected$ ("Sound")
select TextGrid 'este$'
n_de_punto = Get number of points... 2
for i to 'n_de_punto'
  select TextGrid 'este$'
  tiempo_de_punto = Get time of point... 2 'i'
  etiqueta_de_punto$ = Get label of point... 2 'i'
  select Pitch 'este$'
  F0 = Get value at time... 'tiempo_de_punto' Hertz Linear
  printline 'etiqueta_de_punto$"tab$"F0:2'
endfor

```

La primera línea despeja la pantalla de texto; la segunda, identifica el objeto de tipo *Sound* (el símbolo «\$» sirve para identificar un nombre de variable literal). De aquí en adelante, «este\$» sirve para identificar el nombre del objeto ya sea el de un *TextGrid* o el de uno de tipo *Sound*. La línea 3 selecciona el objeto *TextGrid*. La cuarta línea asigna el nombre «n_de_punto» a la variable numérica que consiste en la cantidad de puntos marcados en el estrato 2. En seguida se desarrolla un comando de tipo *for* que considera el número de puntos marcados y, para cada uno de esos puntos, seleccionará en el *TextGrid* la información temporal y la etiqueta correspondiente. Luego seleccionará, en el objeto *Pitch*, el valor de la frecuencia en ese momento. Repetirá la operación para cada punto marcado. La última línea antes de cerrar el comando *for*, envía a la pantalla de texto la etiqueta y el valor de F0 asociado separados por una tabulación («:2» significa ‘con dos decimales’).

Con ese *script* se pueden obtener datos de manera bastante rápida; obviamente funciona con cualquier archivo que tenga un *TextGrid* asociado de similares características. Normalmente lo único que resta por hacer es copiar el texto en una hoja de cálculo y normalizar el símbolo de separador decimal (coma o punto).

Si el investigador necesita obtener la diferencia en semitonos entre dos frecuencias, puede tener a la mano el siguiente *script*:

```

clearinfo
form Convertir diferencia de Hz a St
natural frecuencia_1 100
natural frecuencia_2 200
endform
resultado = (log2('frecuencia_2'/frecuencia_1))*12)
print 'resultado:2'

```

Con la segunda línea, se despliega un formulario en el que se piden las frecuencias (100 Hz y 200 Hz, son los valores por defecto) y en la última línea se envía a la pantalla de texto el resultado de la conversión de la diferencia en Hz a semitonos.

6. PALABRAS FINALES

Con lo presentado en este trabajo no se agotan las posibilidades de análisis en el programa Praat; muchos elementos han quedado sin mencionar. Al investigador interesado, se le recomienda en primer lugar la lectura detenida del menú de ayuda del programa y también la visita a la lista de discusión de Praat. Muchas inquietudes se comparten por esa vía y, en numerosos casos, el mismo creador del software da respuesta a las preguntas de los usuarios. También ahí se pueden encontrar numerosos *scripts* de utilidad para los investigadores.

7. REFERENCIAS BIBLIOGRÁFICAS

- BOERSMA, P. y D. WEENINK (2008): *Praat: doing phonetics by computer* (Versión 5.0.20) [Programa computacional].
3-04-2008: <http://www.praat.org/>
- COLEMAN, J. (2005). *Introducing Speech and Language Processing*, Cambridge, Cambridge University Press.
- JOHNSON, K. (1997): *Acoustic & Auditory Phonetics*, Blackwell, 2005.
- LADEFOGED, P. (1962): *Elements of Acoustic Phonetics*, Chicago and London, The University of Chicago Press, 1996.
- LADEFOGED, P. (2003): *Phonetic Data Analysis. An Introduction to Fieldwork and Instrumental Techniques*, Oxford, Blackwell Publishing.