

## CAPÍTULO 5

# Algoritmos: producción y distribución musical en la actualidad

*Gastón Chatelet*

Los algoritmos son conjuntos ordenados de operaciones o reglas, que permiten hallar la solución a un problema. Al presentarse un estado inicial o *entrada* de información, siguiendo pasos u operaciones se llega a un estado final o *salida*. Estos son más frecuentes de lo que creemos en nuestro día a día: las instrucciones que nos dan para realizar un trabajo, las decisiones que tomamos al manejar un vehículo, o el procedimiento de afinar una guitarra, son algunos ejemplos de algoritmos. Veamos a continuación [Imagen 1] un algoritmo cuyo objetivo es determinar cuál es la causa de que un teclado no suene e intentar solucionarlo.



**Imagen 1:** Algoritmo cuya entrada es el estado del teclado, cuyos procesos son las preguntas y cuyas salidas son las resoluciones de dichas preguntas.

Por simple que parezca, este razonamiento está planteado de forma algorítmica: hay un estado inicial (el teclado no suena), hay pasos a seguir (las preguntas que nos hacemos), y hay un estado final (al cual se arriba contestando a las preguntas). Casos tan simples como este, pueden complejizarse, por ejemplo, haciendo más preguntas o que estas sean más específicas. En nuestro ejemplo, podríamos preguntarnos si está bien enchufado, si los cables no están cortados, si genera voltaje al medirlo con un *tester*, si el dispositivo al que está conectado (parlantes, computadora, etc.) está encendido o bien configurado (seteado), etc.

Al igual que cuando pensamos a una música en términos de su *espectro de frecuencia*, lo que nos permite conocer cómo se distribuyen las magnitudes de grupos de frecuencias en el tiempo y que podríamos llamar una perspectiva *espectral* del sonido, para comprender como funcionan algunas de las funciones de los dispositivos de producción y distribución musical en la actualidad, nos será útil conocer cómo funciona su *algoritmia*, es decir, como son sus estados iniciales, sus procesos y sus salidas, lo que podríamos llamar una perspectiva *algorítmica* de dichos dispositivos.

## Dispositivos de producción y distribución musical

Cuando hablamos de dispositivos nos referimos a todo mecanismo capaz de producir una acción concreta y previsible. En el caso de los dispositivos de producción y de distribución musical, estos mecanismos son, tanto los procedimientos llevados a cabo al momento de producir y distribuir música, como los sistemas, soportes o plataformas que nos habilitan a realizar dichos procedimientos. Tomemos un ejemplo para comprender esta dualidad.

Al hablar de *la radio*, casi todos entendemos en gran medida de que estamos hablando ¿no? Vale la pena, a los fines de este escrito, aclarar y comprender las acepciones que se desprenden de esta palabra o los sentidos habituales que le solemos dar. Si entendemos a la “radio” como apócope de la radiodifusión, es decir al proceso de transmisión propiamente dicho, entonces *la radio* puede ser efectivamente un dispositivo, en este caso, de distribución musical, si se la utiliza con dicha finalidad. También podríamos pensar en *la radio* como el espacio físico desde donde se transmite, es decir la estación radial (transmisora), o bien a el conjunto de aparatos necesarios para la transmisión, los radiotransmisores.

Como artistas, podríamos presentar un nuevo material musical yendo a una estación radial, transmitiendo nuestra música mediante el proceso de radiodifusión y esperando que alguien nos escuche del otro lado.

Del mismo modo, si entendemos a *el radio* como apócope de radiorreceptor, es decir el aparato que recibe la señal, este también puede ser un dispositivo de distribución musical, siempre que se lo utilice con ese fin.

Por más que transmitamos nuestro material musical desde una estación radial, si del otro lado no hay ningún radio (receptor) encendido y sintonizando el dial de la estación, pues entonces no se concretaría el proceso y el dispositivo no estaría funcionando como es esperado. Ahora bien, si un colega nos comenta que va a presentar un material musical en un programa específico a una hora concreta y nosotros sintonizamos ese dial en un radio (receptor), entonces este estaría cumpliendo la función de un dispositivo de distribución musical. ¿Pero no sería más bien un dispositivo de consumo? Para nuestro colega es un dispositivo de distribución y para nosotros es un dispositivo de consumo musical, puesto que nuestro objetivo es el de consumir música y el de nuestro amigo distribuirla.

Como vemos, si bien todas estas acepciones del término son de diferente naturaleza (un proceso, un espacio físico o un elemento), estas operan de manera interdependientes. Es decir, el proceso de distribución musical no se finaliza (no cobra sentido, ni llega a su meta) sin la acción del objeto (sea la estación radial, el radiotransmisor o el radiorreceptor) y el objeto no cobra sentido sin el proceso (la radiotransmisión).

El ejemplo de el/la radio nos sirve como puntapié para comprender la naturaleza de los dispositivos de distribución musical, en tanto plantea una definición ambigua, cuyos límites son difusos. Esta indefinición nos habilita a reinterpretar los elementos que forman parte del campo y estudiarlos no de forma aislada, sino como parte de un todo.

Por lo tanto, cuando nos refiramos a dispositivos de producción musical, nos estaremos refiriendo a todos los elementos que forman parte del proceso de producción, y cuando nos refiramos a dispositivos de distribución musical, estaremos hablando de todos los elementos que forman parte del proceso de distribución.

Cuando adoptamos una perspectiva *algorítmica* de los dispositivos, vemos a los algoritmos como elementos de estos mecanismos. En esta oportunidad nos vamos a detener en tres de ellos que son de nuestra incumbencia como productores y distribuidores musicales independientes: i) la digitalización de señales acústicas, ii) la compresión con pérdida y iii) la normalización de señales de audio por parte de los servicios de distribución musical.

## Digitalización de señales acústicas

El “WAVE”, más conocido como WAV, es un formato de audio digital que surge a principios de la década del 90 y que al poco tiempo logra cierta estandarización, llegando hasta nuestros días como el formato por excelencia para trabajar *sin pérdida de calidad*<sup>23</sup>. Usualmente este formato es el que se utiliza al momento de grabar música digitalmente de forma profesional, en la mayoría de los programas conocidos como *Estaciones de Audio Digital* (del inglés *DAW* o *digital audio workstation*), como pueden ser Reaper, Nuendo, Cubase, Ableton Live, Logic Pro, etc., o bien en grabadoras portátiles o cualquier sistema que trabaje con audio digital. El método de grabación digital que utiliza este formato es el de *modulación por impulsos codificados* (del inglés *PCM* o *Pulse Code Modulation*). Este método de modulación transforma la señal analógica en una señal digital (secuencia de bits) mediante dos procedimientos: el *muestreo* y la *cuantificación*. Brevemente estudiaremos ambos.

El muestreo es el proceso de tomar “muestras” de sonido en intervalos definidos de tiempo (Miyara, 2006, p. 164). Estas muestras equivalen a valores de tensión eléctrica (voltaje) que llegan desde el micrófono y que varían en el trayecto de la onda acústica, en sus diferentes picos y valles. El intervalo de tiempo con el que se suele tomar estas muestras es de 0.0000226 segundos, o ~23  $\mu$ s. Este intervalo tan pequeño de tiempo nos permite tomar exactamente 44100 muestras en un segundo, por lo que decimos que la frecuencia de muestreo es de 44100 Hz (“hz” es la unidad de frecuencia que equivale a ciclos por segundo), o bien 44.1 kHz. Este número<sup>24</sup> tan particular se debe a que, para poder digitalizar una señal analógica con la mayor fidelidad posible, la frecuencia de muestreo (es decir la cantidad de muestras que tomemos en un segundo) debe ser igual o mayoral doble de la frecuencia máxima de interés<sup>25</sup>, que en nuestro caso sería el límite superior audible, aproximadamente 20 kHz.

---

23 La pérdida de calidad es inherente a cualquier sistema de codificación de audio, pero en los formatos “sin pérdida” como WAV, la pérdida es mínima o bien se encuentra por fuera de los rangos audibles por el humano (ya sea en términos de frecuencia o de amplitud)

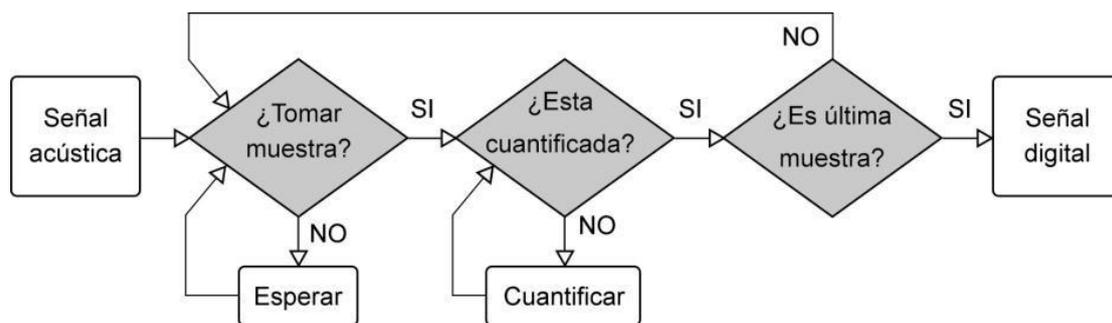
24 Esta frecuencia o tasa de muestreo es la estandarizada para la calidad CD de Audio, pero también suelen utilizarse otras frecuencias, como 48Khz o 96kHz. Cabe aclarar que cuanto mayor sea la frecuencia de muestreo, mayor será el espacio necesario para almacenar un mismo fragmento sonoro-musical.

25 Teorema de muestreo de Nyquist-Shannon (Miyara, 2006, p.166).

La cuantificación es el proceso que sigue al muestreo durante la grabación. Este proceso se encarga de otorgar valores digitales (bits) a los valores de voltaje tomados durante el muestreo (Rumsey-McCormick, 1994, p. 205). Como los valores de voltajes que pueden tomarse en las muestras son infinitos, es necesario que estos se acoten a valores razonables, es decir, valores con los cuales la fidelidad sea máxima, pero sean manejables por un ordenador. Dicho de otro modo, la cuantificación es el proceso de hacer *discretos* valores que son *continuos*. El número de valores discretos que puede adoptar una muestra está definido como la *resolución digital*, y en el caso del formato CDDA (*Compact Disc Digital Audio*, en adelante CD-Audio) es de 16 bits, lo que equivale a  $2^{16} = 65.536$  valores (por lo general 32.768 para valores positivos y 32.768 para valores negativos).

Esta breve explicación del proceso de digitalización de señales acústicas nos permite entender los requerimientos necesarios para que una grabación con calidad profesional, la del formato CD-Audio (44.1 kHz y 16 bits).

Recapitulando, vemos como el proceso de digitalización que acabamos de estudiar, responde a un procedimiento algorítmico [Imagen 2], siendo la *señal acústica* la entrada, el *muestreo* y la *cuantificación* los pasos a seguir, y la *señal digital* la salida.<sup>26</sup>



**Imagen 2:** Algoritmo esquemático del proceso de digitalización de una señal acústica que comprende dos instancias: la de muestreo y la de cuantificación.

Una vez que tenemos digitalizada nuestra señal, es momento de decidir qué hacer con ella. Supongamos que decidimos grabar de manera medianamente profesional, conseguimos el equipamiento adecuado, retocamos los ajustes en nuestro programa de grabación y conseguimos realizar una buena toma. Luego de trabajar en la edición y la mezcla de nuestra música, finalmente decidimos que está terminada y la exportamos en el formato estudiado anteriormente: WAV. Perfecto, ahora, ¿qué hacemos con ella? Podríamos grabarla en un CD-Audio, ya que sabemos que nuestro archivo se adapta a las especificaciones de este soporte, aunque si quisiéramos compartirla con muchas personas, grabar numerosos CD-Audio sería en principio costoso y demandaría mucho tiempo<sup>27</sup>. Podríamos entonces optar por compartir nuestra obra por otro medio, subiéndola a internet, por ejemplo.

<sup>26</sup> Este caso es esquemático, dentro del proceso de digitalización tienen lugar otros subprocesos como el de filtrado de frecuencias superiores al límite superior, o la corrección del error (ruido) de cuantificación.

<sup>27</sup> Previamente a la llegada de internet, este era el procedimiento a seguir para difundir nuestra música. De igual manera, el CD-Audio se sigue manteniendo hasta la actualidad como el standard de soporte físico para la difusión musical.

## Menor tamaño, menor calidad, mayor velocidad

Ya sea enviando el archivo de audio por WhatsApp, Instagram, Facebook, o subiéndolo a Spotify o YouTube, en algún momento, la información (por ejemplo, nuestra música) que queremos compartir se va a subir o cargar (por nosotros) y se va a bajar o descargar (por quien vaya a escuchar) de internet. Incluso en una transmisión en vivo (*streaming*) de cualquier red social, el proceso de carga y descarga está actuando. Ahora bien, lo que subimos ¿es exactamente lo que descarga quien está del otro lado? La respuesta, por lo general, es negativa. Por más que subamos un archivo de audio con la mejor calidad que tengamos, es muy probable que la plataforma en la que lo estemos subiendo realice una compresión de dicho archivo, por una sencilla razón: estos sitios necesitan que los archivos sean livianos. Pero, ¿por qué? Cuanto más espacio ocupen los archivos (archivos pesados), mayor será el espacio que necesita el sitio para su almacenamiento y más son los datos a transferir al momento de descargarlos, por lo que demandará una mayor *velocidad de transferencia de datos* para su correcta reproducción.

La *velocidad de transferencia de datos* es el factor que va a determinar cuán “rápido” se van a descargar y a subir los datos o la información que estemos transfiriendo. En la telefonía móvil, los cambios de tecnología de 2g a 3g y luego a 4g, improvisaron la velocidad de transferencia de datos [Tabla 1], llevándola a valores cada vez mayores (Segura i Navarro, 2003, p. 151). Es por ello que, si configuramos nuestro celular para que trabaje con 2g, probablemente tarde mucho tiempo en descargar los archivos, pero si lo configuramos en 4g lo descargue casi de forma instantánea. Para simplificar veamos la siguiente tabla:

Generación	Conexiones	Velocidad máximas	
		Bajada	Subida
2G	2G o GSM	9.6 kbps	9.6 kbps
	G	80 kbps	20 kbps
	E	236 kbps	59 kbps
3G	3G	384 kbps	384 kbps
	H	7,2 mbps	7,2 mbps
	H+	22 mbps	22 mbps
4G	LTE	75 mbps	25 mbps

**Tabla 1:** Velocidades máximas de bajada y de subida (expresadas en kilobits por segundo) para cada generación de tecnología móvil y cada tipo de conexión

Si quisiéramos descargar un audio de WhatsApp que pesa 5MB (5 megabytes = 40 megabits = 40000 kilobits)<sup>28</sup> tardaríamos 5 minutos en modo G3-H y 0.5 segundos en modo 4G. Lo mismo se aplica para la reproducción<sup>29</sup> de archivos alojados en Spotify o YouTube (aunque este último contiene videos por lo que el peso de los archivos podría ser mayor e incluso reducir el tamaño del audio en

28 1 MB (megabyte) = 8 Mb (megabit) = 0.953674 MiB (mebibyte)

29 Cuando reproducimos archivos de plataformas como YouTube o Spotify, estamos descargando datos desde el proveedor a la vez que los vamos usando, por lo que no necesitamos esperar a que se termine de descargar el archivo por completo. Este tipo de descarga utiliza un buffer de datos creado en nuestra computadora, donde se almacena la información de manera temporal y que, terminada la sesión se elimina automáticamente.

favor de incrementar el de la imagen). Al igual que en la telefonía móvil, la velocidad de transferencia de datos también se aplica en redes Wifi, tomando los valores específicos de esta. Por ejemplo, si tuviésemos contratado un servicio de internet de 50 Megas con Wifi, la velocidad de descarga teóricamente sería de 50 megabytes por segundo (aunque en la realidad esto se limita a un 70% de la capacidad: ~36Mbps).

Además, cuando estamos subiendo o bajando contenido con nuestros celulares, no solo nos interesa que la *velocidad de transferencia* sea la mayor posible, sino también (en caso de no estar conectados a una red Wifi) el peso de los archivos, puesto que la cantidad de datos para descargar y cargar por mes es limitada (a diferencia de las redes Wifi actuales, donde aparentemente no rige un límite real, sino que el límite se da por la velocidad misma de velocidad de transferencia). Con un plan de 1GB de datos mensuales, tenemos solamente 1000 MB para descargar/cargar. Si viésemos 4 capítulos de 40 minutos de alguna serie en Netflix, es probable que al querer mirar el 5 capítulo nos encontremos con que hemos agotado nuestro pack de datos móviles. Este es un punto que retomaremos al estudiar las configuraciones de Spotify.

Por lo general, si estamos utilizando un ordenador que está conectado a Wifi, no deberíamos preocuparnos por las velocidades y el peso de los archivos, ya que los ordenadores tienen procesadores más eficientes que los móviles, así como mayor capacidad de almacenamiento y mejor transferencia de datos. Si vemos un video en YouTube, por ejemplo, podemos notar que, si no hemos modificado nada previamente, la calidad de reproducción por defecto va a estar configurada en “automático”. Esto significa que la plataforma (algoritmos de por medio) va a decidir cuál es la *mejor calidad* para reproducir el video sin que este “corte”, es decir sin que se interrumpa la reproducción en función de nuestra conexión. De este modo, si nuestra conexión es defectuosa, tenemos problemas con internet, o estamos descargando otras cosas, el ancho de banda de descarga va a saturarse y YouTube puede decidir que lo conveniente sea reproducir un archivo en baja calidad (o en una calidad inferior a la que deseáramos). Esto podríamos modificarlo cambiando el tipo de calidad a la que quisiéramos, pero probablemente la reproducción se vea interrumpida por las razones mencionadas.

Pero, ¿qué significa que YouTube reproduzca en diferentes calidades? Lo que significa, es que la plataforma tiene alojados múltiples formatos de video y de audio con diferentes calidades cada una y que precisan, por lo tanto, una mínima tasa de bits<sup>30</sup>. A continuación [Tabla 2], veamos algunos ejemplos de tasa de bits que YouTube nos recomienda al momento de configurar nuestro video, previo a la carga del mismo en la plataforma:

---

<sup>30</sup> La tasa de bits o Bitrate se define como el número de bits que se transmiten (consumen) en un segundo en los sistemas de transmisión digital, por lo que es, en definitiva, la velocidad de transferencia de datos (Gupta, 2006, p. 21).

Calidad	Tasa de Bits	
	24, 25, 30 fps	48, 50, 60 fps
2160p (4K)	35 a 45 mbps	53 a 68 mbps
1440p (2K)	16 mbps	24 mbps
1080p (HD)	8 mbps	12 mbps
720p (HD)	5 mbps	7,5 mbps
480p	2.5 mbps	4 mbps
360p	1 mbps	1,5 mbps

**Tabla 2:** Valores recomendados por YouTube para la codificación de la carga de videos, en función de la velocidad (standard y alta) de fotogramas (fps).

Pero estos valores<sup>31</sup> son de video, consideremos que cuando subimos a YouTube un archivo de video, en el mismo ya se encuentra codificado canal de audio. Esto abre un nuevo abanico de posibilidades, ya que cada formato de video permite un formato de audio asociado. Para comprenderlo mejor imaginaremos una situación.

Consideremos que una vez que hemos concluido con nuestra producción musical, habiéndola grabado y exportado en una buena calidad, digamos en calidad CD-Audio (44.1 kHz y 16 bits), queremos subirla a YouTube para que el mundo la conozca. Hemos conseguido o realizado el clip de video que va a acompañar nuestra pieza musical y hemos estado trabajando en un software de edición de video donde fuimos mezclando ambas dimensiones, la visual y la sonora. Hemos llegado al punto en el que tenemos que renderizar (generar un archivo para exportar) nuestro video. ¿Qué decisión tomamos? Como buena medida, deberíamos tener en cuenta las especificaciones de las plataformas en las que vamos a cargar el material.

Si fuese YouTube, el servicio nos recomienda algunos parámetros para codificar el video antes de subirlo:

- que el contenedor sea MP4;
- que el códec de audio sea AAC-LC con tasa de muestreo de 48 kHz o 96 kHz;
- que el códec de video sea H.264; iv) que la tasa de bits de video sea la especificada en la tabla anterior; v) que la tasa de bit de audio sea 128kbps (mono), 384 kbps (stereo) o 512 kbps (5.1).

Estos parámetros, recordemos, son los proporcionados por YouTube para aprovechar al máximo la plataforma. Podríamos perfectamente subir nuestro archivo en otras calidades, pero, al parecer, YouTube solo nos aseguraría las calidades de salida(descarga) antes mencionadas.

Con esta información, más allá de que sepamos o no configurarla antes de codificar nuestro video para subirlo a la plataforma, podemos darnos cuenta de que nuestro muy preciado archivo WAV de 44.1 kHz y 16 bits, no será el que se cargará a YouTube, y por lo tanto no va a ser que se descargue al momento de la reproducción. Entonces, ¿Qué oiremos?, o quizás más importante aún ¿Qué oirá nuestro público? Lo que se oirá será una versión *comprimida* y con pérdidas de nuestro archivo inicial. Dicha pérdida, como veremos más adelante, se traduce

---

31 Estos valores son para cargas de rango dinámico estándar o SDR (standard-dynamic-range video), para cargas con rango dinámico alto o HDR (high-dynamic-range video) los valores son mayores.

en una merma en la calidad acústica del material, producto de una búsqueda por disminuir el tamaño de los archivos.

## Cuentas claras - algo ritmos (K.J.)

Tomemos un ejemplo real, analicemos el caso de la canción “Cuenta Claras” de Kevin Johansen, perteneciente al material discográfico publicado, en 2019, el álbum “Algo Ritmos”. La versión de audio digital que tomaremos para su estudio está alojada en el canal oficial del artista en YouTube (Johansen, 2019) y es patrocinada por VEVO<sup>32</sup>. Cuando vamos a la configuración de calidades nos aparecen las siguientes: 1080p<sup>HD</sup>, 720p<sup>HD</sup>, 480p, 360p, 240p, 144p. Para saber exactamente qué tipo de video, y por consiguiente que tipo de archivo o codificación de audio se está utilizando y así saber su calidad, debemos encontrar los archivos alojados en el link. Utilizando una herramienta que permite analizar repositorios web, se pudieron rastrear todos los archivos alojados en el repositorio. Esta herramienta nos permitió encontrar en cuestión [Tabla 3]:

- 35 archivos de video, de los cuales 11 son en formato *mp4*, 18 en formato *webm* y 6 en formato *mkv*. La *tasa de bit (Bitrate) de audio* varía entre 48, 64 y 128 kbps y los códec de audio utilizados son Opus y AAC. El tamaño de los archivos varía entre 65.116 KB y 3.584 KB.
- 7 archivos de audio: de los cuales 4 son en formato *m4a*, 2 en formato *aac* y 3 en formato *ogg*. La *tasa de bit (Bitrate) de audio* varía entre 64, 96 y 152 kbps y los códec de audio utilizados son Opus y AAC. El tamaño de los archivos varía entre 3.443 KB y 1.308 KB

A partir de la información recolectada y cotejando con otros ejemplos de artistas y conjuntos reconocidos y patrocinados por VEVO, podemos inferir que:

- por más que subamos un video a YouTube con las especificaciones técnicas que nos brinda la plataforma, esta va a comprimir tanto el video como el audio de nuestros archivos;
- si subimos videos con mayor calidad que la que nos especifica la plataforma, también va a comprimirse el contenido<sup>33</sup>.

---

32 VEVO es un sitio web y una plataforma de videos musicales de gran envergadura que en la actualidad tiene una presencia importante en el mercado. La empresa administra la marca de artistas de renombre, en conjunto con Google, Sony Music Entertainment, Universal Music Group, Abu Dhabi Media Company y Arthur Music Company.

33 Esto se pudo cotejar tras subir un contenido audiovisual de mayor calidad a la descrita como óptima para la plataforma (con mayor Bitrate tanto en video como en audio).

Tipo	Formato	Anchura (px)	Altura (px)	Codec de Video	Codec de Audio	Bitrate de Audio	Tamaño	
Video	mp4	1920	1012	H264	AAC	128 kbit/s	65.116 KB	
		1280	674	AV1			31.554 KB	
				H264			21.572 KB	
		854	450	AV1			17.440 KB	
				H264			16.116 KB	
		640	338	AV1			11.356 KB	
				H264			11.735 KB	
		426	224	AV1			7.874 KB	
	H264			7.353 KB				
	256	134	AV1	5.513 KB				
			H264	5.813 KB				
	webm	1920	1012	VP9	Opus	128 kbit/s	62.286 KB	
						64 kbit/s	60.515 KB	
		48 kbit/s	60.075 KB					
		1280	674			128 kbit/s	36.729 KB	
						64 kbit/s	34.958 KB	
		48 kbit/s	34.518 KB					
		854	450			128 kbit/s	20.388 KB	
						64 kbit/s	18.617 KB	
		48 kbit/s	18.177 KB					
		640	338			128 kbit/s	12.885 KB	
						64 kbit/s	11.115 KB	
		48 kbit/s	10.675 KB					
		426	224			128 kbit/s	8.661 KB	
						64 kbit/s	6.890 KB	
		48 kbit/s	6.450 KB					
		256	134			128 kbit/s	5.795 KB	
	64 kbit/s			4.024 KB				
48 kbit/s	3.584 KB							
mkv	1920	1012	VP9	AAC	128 kbit/s	62.160 KB		
						1280	674	36.603 KB
						854	450	20.261 KB
						640	338	12.759 KB
						426	224	8.535 KB
						256	134	5.669 KB
Audio	m4a			AAC		128 kbit/s	3.416 KB	
						96 kbit/s	2.572 KB	
	aac				AAC		128 kbit/s	3.443 KB
							96 kbit/s	2.598 KB
	ogg				Opus		152 kbit/s	3.523 KB
							96 kbit/s	1.742 KB
						64 kbit/s	1.308 KB	

**Tabla 3:** Archivos individuales disponibles en el repositorio de YouTube para el video oficial de *lacanción Cuentas Claras* de Kevin Johansen, alojada en el canal oficial del artista.

Pero, en definitiva, ¿cómo se traducen estos números (*kpbs*) y estas siglas (ogg, aac, m4a, etc.) a lo que oímos?, ¿en qué medida varía psicoacústicamente una compresión de 128 kbps y una de 48 kbps? Para comprender los efectos auditivos que involucra la compresión de audio es necesario entender cómo funciona el proceso de compresión<sup>34</sup> en sí mismo.

## Algoritmos de compresión con pérdida

El proceso de compresión de audio, al igual que el de la compresión de video, tiene por objeto reducir el tamaño de los archivos. En el caso de compresión con pérdida, el proceso se realiza

<sup>34</sup> En este capítulo, al hablar de compresión de audio, nos referimos estrictamente al proceso de reducción del tamaño de los archivos de audio y no al proceso de compresión de sonido en donde se reduce el rango dinámico de la señal.

mediante *algoritmos de compresión con pérdida*<sup>35</sup>, en adelante ACP, donde la reducción en el tamaño de los archivos se da a costa de una merma en la calidad del sonido. Los ACP involucran procedimientos que, mediante la codificación, buscan representar una cierta cantidad de información utilizando una porción de esta. Es decir, dada una *entrada* de información, realizan procedimientos que llegan a una salida en donde la información inicial esté representada, pero de forma parcial. En otras palabras, estos algoritmos buscan reducir en alguna medida, la cantidad de datos necesarios para representar una información dada. Los ACP son aquellos utilizados al momento de codificar o convertir ciertos formatos de archivos en otros de menor tamaño pero también con menos información.

Este procedimiento tiene sus ventajas y desventajas: i) Al comprimir se obtienen archivos de menor tamaño, lo cual, ii) nos permite el ahorro de espacio (en una misma porción de almacenamiento pueden entrar más archivos, puesto que su tamaño es menor) y de la velocidad de transferencia de datos (al contener los archivos, menos datos, la información que se transfiere necesita menor velocidad de transferencia: por ejemplo, al descargar o al subir archivos, al compartirlos, etc.); por otro lado, iii) los archivos sufren una pérdida de calidad, puesto que se eliminó información (datos, bits) que al momento de la reproducción (o el análisis) no será posible acceder; además, iv) si bien es posible volver a convertir los archivos a formatos sin pérdida (por ejemplo de MP3 a WAV), la información eliminada durante el proceso de compresión no será reconstruida, de este modo obtendremos un formato (contenedor) de alta calidad con información (contenido) de baja calidad.

El “MPEG-1 Audio Layer III”, más conocido como MP3, es un formato de audio que nace en la década de 1980 y que se populariza a mediados de los 90 (todas las patentes relacionadas al formato MP3 expiraron en el 2017). Este tipo de archivo de audio se caracteriza por ser un formato comprimido con pérdida. Esto significa que en algún momento en su creación fue procesado por un ACP.

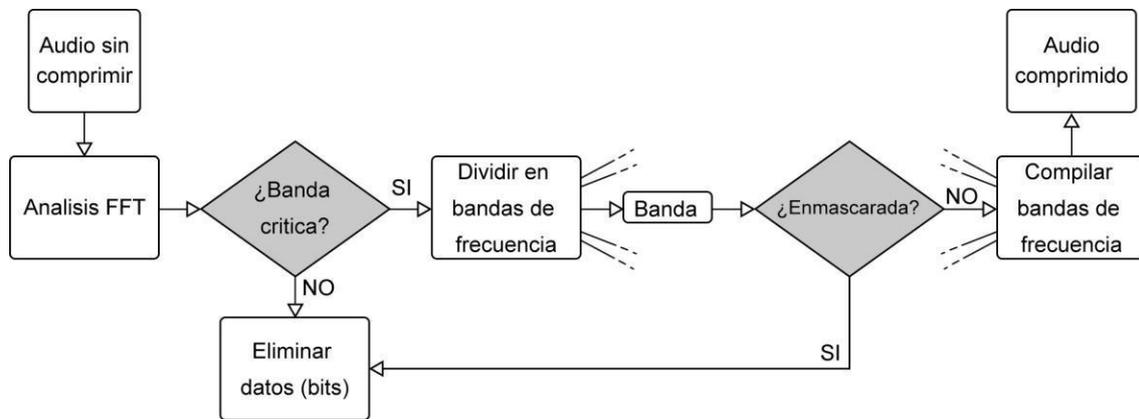
A diferencia de los algoritmos estudiados anteriormente, los ACP son más complicados de esquematizar [Imagen 3]. Por ejemplo, en el caso del algoritmo de compresión MP3, lo que realiza el Códec<sup>36</sup> es basarse en el fenómeno de enmascaramiento sonoro de nuestro propio sistema auditivo. Este fenómeno psicoacústico se da cuando algunos sonidos (enmascarantes) sobresalen de otros (enmascarados) por su intensidad. Cuando grabamos e incluso exportamos un archivo sin compresión, todos esos sonidos están allí, más allá de cómo se comporte nuestro oído oyendo o no oyendo algunos sonidos. La compresión MP3 toma esta cualidad psicoacústica a su favor y elimina los sonidos enmascarados para reducir el tamaño de los archivos. Este proceso considera dos tipos de enmascaramientos: frecuencial y temporal. El *enmascaramiento frecuencial* es la disminución de la sonoridad (parámetro psicoacústico) de un sonido de cierta frecuencia

---

35 Cabe aclarar que existen algoritmos de compresión sin pérdida. Estos, a diferencia de los ACP, permiten una reconstrucción exacta del material luego de su descompresión. Por lo general son utilizados en compresores de archivo como RAR o zip, y en formatos de audio como FLAC, ALAC y Monkey's Audio, siendo estos poco usuales en el mercado.

36 Los códec son programas capaces de codificar y decodificar un flujo de datos. En el caso de MP3, el códec más tradicional es el LAME.

ante la presencia de un sonido de mayor intensidad y en una frecuencia aledaña, cuando ambos sonidos se producen simultáneamente. El *enmascaramiento temporal* es la disminución de la sonoridad de un estímulo sonoro de cierta intensidad respecto de otro de mayor intensidad, cuando estos se producen de forma no simultánea (Miyara, 2006, p. 29).



**Imagen 3:** Algoritmo esquemático del formato de compresión MP3. Se eliminan los datos de las bandas que exceden a la banda crítica (audible) y se divide la señal en sub-bandas. Estas luego se cuantifican en función de si son o no enmascaradas, reduciendo así los datos (bits) necesarios para la representación de la señal entrante.

Así como el formato MP3, para cada formato que haya sido comprimido con pérdida, un ACP tuvo lugar. En el ejemplo de Kevin Johansen estudiado anteriormente, no vimos archivos mp3, sino: mp4, webm, mkv, aac, m4a y ogg. De todos ellos, lo que nos interesa es saber cómo fueron comprimidos, y en particular como fue comprimido el audio. Como se puede apreciar en la *Tabla 3*, tanto para los archivos de audio como los de video se utilizaron dos códec: AAC y Opus. Estos códec pertenecen a diferentes empresas y tiene diferentes algoritmos de compresión, por lo que dan diferentes resultados psicoacústicos.

AAC pertenece al grupo MPEG (Moving Picture Experts Group), creador de MP3, MP4, y todas sus variantes, incluido este códec. Opus es un códec abierto desarrollado por la fundación Xiph.Org (en colaboración con Mozilla y Skype). Esta organización, sin fines de lucro, también desarrolló otros códec de compresión como Vorbis<sup>37</sup> (anterior a Opus) y FLAC. Este último es un códec abierto de compresión *sin* pérdidas, que permite la reconstrucción exacta (sin pérdida de información) del material comprimido una vez que este se descomprime.

Para poder comprender mejor cómo y en qué medida varía la información contenida en los archivos luego de ser comprimidos, realizamos una pequeña prueba.

Grabamos un fragmento musical perteneciente al riff de guitarra con el que inicia la canción “Luna de Abril” de *Spinetta y los Socios del Desierto*. Se procuró una grabación en las mejores condiciones posibles: en un espacio acústicamente apto, con un micrófono capacitivo (también llamado *de condensador*) Samson C03, una placa de sonido Komplete Audio 6 (de Native Instruments), en una computadora Acer Aspire ES1 (con procesador i5 y 8GB de memoria RAM)

<sup>37</sup> Vorbis tiene un tiempo de retraso de 100ms, cuando en Opus es de 26,5 ms (es decir el tiempo necesario para ir codificando bloques de audio), lo que lo hace ideal para la transmisión en vivo (streaming) y la telecomunicación.

y utilizando el software de audio (DAW) Ableton Live (versión 10.1 Suite)<sup>38</sup>. Este *software* se configuró para que la grabación (es decir la digitalización de la señal acústica) se realice en calidad Cd-Audio, 44.1 kHz para el muestreo y 16 bits de resolución.

Los potenciómetros de ganancia en el canal de la consola se mantuvieron a valores razonables, y no se alteraron valores de la pista de audio en el software de grabación, ni se agregaron efectos. Luego de esta configuración se procedió a grabar el fragmento musical. Una vez realizada una toma satisfactoria, se procedió a exportar la sesión a un archivo WAV.

Perfecto, ya tenemos nuestro fragmento musical en buena calidad (WAV 44.1 kHz 16 bits). Ahora vamos a proceder a comprimirlo con diferentes códec de compresión para luego comparar los resultados. El conversor de audio utilizado es el *AIMP2 audio converter*, el cual contiene todos los códec que necesitamos. Se realizó una conversión del formato WAV a FLAC, MP3 y OGG, con valores de Bitrate variables [Tabla 4].

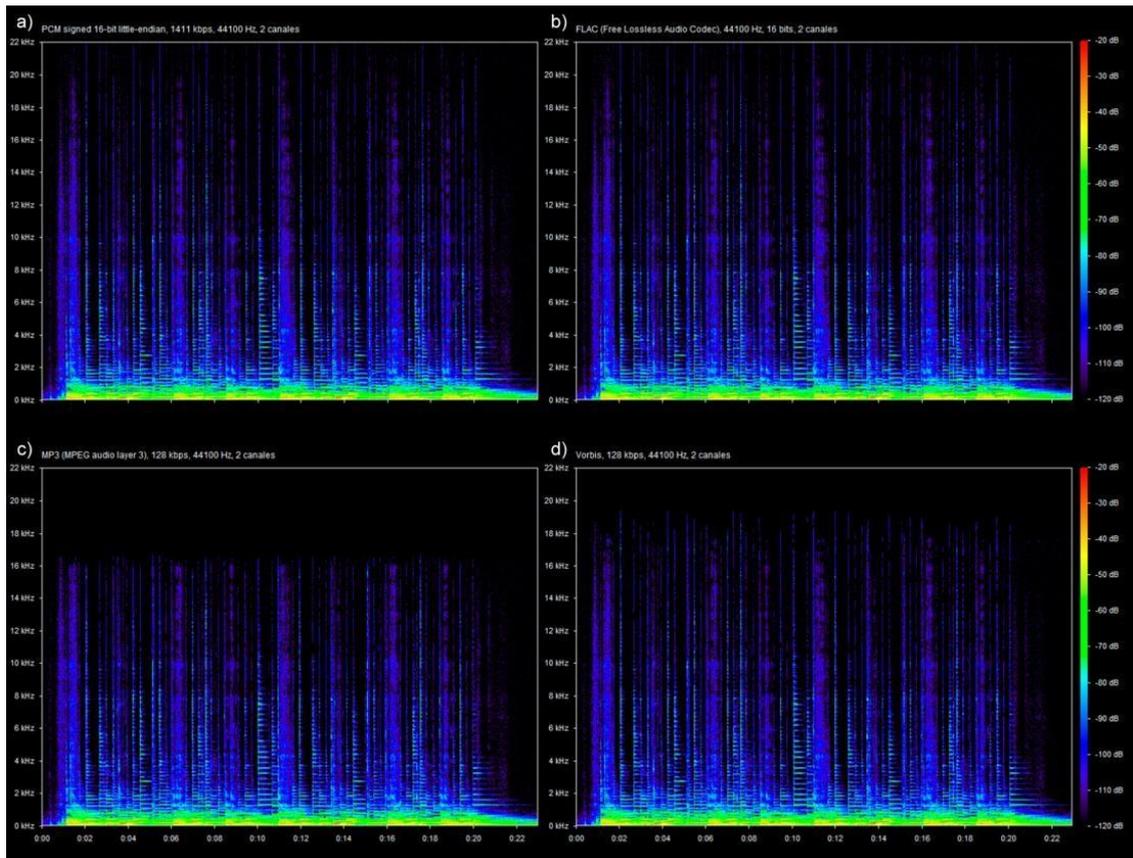
Formato	Bitrate	Tamaño	Frecuencia de corte
WAV (original)	1411 kbps	3.953 KB	22,05 kHz
FLAC	315 kbps	848 KB	22,05 kHz
OGG	192 kbps	548 KB	20,00 kHz
MP3	192 kbps	544 KB	18,50 kHz
OGG	128 kbps	368 KB	20,0 kHz
MP3	128 kbps	364 KB	16,5 kHz
OGG	96 kbps	278 KB	16,5 kHz
MP3	96 kbps	275 KB	15,5 kHz
OGG	48 kbps	141 KB	14,0 kHz
MP3	48 kbps	139 KB	7,5 kHz

**Tabla 4:** Detalle de la tasa de bits, el tamaño (KB) y la frecuencia de corte (aproximada) del archivo original (WAV) y archivos comprimidos.

El recorte de frecuencias se puede apreciar mejor en la [Imagen 4 (a, b, c y d)], donde se realizó un análisis espectral de 4 archivos (WAV, FLAC, MP3 y OGG) mediante el software *Spek 0.8*.

---

<sup>38</sup> Todos estos datos no son menores, puesto que, al momento de la digitalización de la señal acústica, cada componente de nuestro sistema de audio va a influir en la calidad del archivo grabado



**Imagen 4:** Análisis espectral del fragmento grabado (“Luna de abril” de Spinetta) en sus diferentes formatos: a) archivo WAV original, sin comprimir; b) archivo FLAC comprimido sin pérdidas; c) archivo MP3 de 128 kbps, recorte de frecuencias a los 16.5 kHz; d) archivo OGG de 128 kbps recorte de frecuencias a los 20 kHz

Como podemos observar, la compresión genera archivos con menor tamaño a costas de una pérdida de calidad, la cual se ve reflejada en la tasa de bits (bitrate) y la frecuencia de corte (por encima de esta frecuencia se eliminan todos o la mayoría de los datos). Esta información nos permite saber, realmente, qué es lo que oímos al reproducir, por ejemplo, un archivo alojado en YouTube que tiene 128 kbps. Si bien la plataforma nos recomienda subir archivos de 192 kbps, lo que oiremos será otra cosa.

## Normalización y LUFS

La normalización es un proceso que busca modificar la amplitud de una señal de audio a un valor asignado como objetivo. Este proceso puede ser a partir del valor máximo que se registre de la señal (llamado valor de cresta o pico) o a partir de la sonoridad, promediando el valor medio de potencia (por ejemplo, con el valor RMS) y modificando la amplitud para llevar este valor al objetivo.

En ambos casos, la búsqueda tiene como finalidad llevar la amplitud de una señal a un valor específico. En la etapa de masterización musical, este proceso es útil para equilibrar la sonoridad entre múltiples canciones, si es que esa es la búsqueda del artista. Ahora bien, ¿este proceso

finaliza en la etapa de masterización? Luego de esta etapa, al difundir nuestra música, ¿lograremos los niveles de sonoridad que buscamos para nuestro público?

En octubre del año 2015, la ITU (Unión Internacional de Telecomunicaciones) aprobó la Recomendación BS.1770-4 bajo el nombre: “Algoritmos para medir la sonoridad de la señal de audio de los programas y el nivel de cresta de audio real” perteneciente a la serie *Broadcasting Service (sound)*. Esta recomendación, como lo indica su nombre, especifica los algoritmos de medición de audio con el propósito de detectar valores subjetivos de sonoridad de programas de audio y valores de cresta (pico) real de señales de audio.

En resumidas cuentas, el objetivo de esta recomendación es la de brindar a servicios de difusión de audio (como YouTube, Spotify, SoundCloud, etc.), una herramienta (algoritmo) al momento de realizar las mediciones de sonoridad de los archivos a difundir (por ejemplo, nuestras cargas). Con esta herramienta, los programas radiales, televisivos, de difusión online, etc., buscan estandarizar un nivel de sonoridad subjetivo. ¿Qué significa esto? Que al escuchar múltiples tracks de audio (ya sean de un álbum alojado en Spotify, de videos de YouTube o bien capítulos de una serie en Netflix), tengamos la sensación de que todos están nivelados en intensidad. Esta búsqueda de las plataformas radica en quitarle al usuario la “molestia” de estar constantemente modificando el volumen de su reproductor (por ejemplo, en la televisión, usando constantemente el control remoto para subir o bajar el volumen al ir cambiando los canales).

En principio, la medida parecería tener un buen horizonte, ya que muchas veces puede resultar molesto encontramos con tracks consecutivos con mucha diferencia en su sonoridad. Esto nos lleva a preguntarnos: en el caso de trabajar con música ¿No será acaso que la diferencia en el tratamiento de la sonoridad y el rango dinámico de múltiples canciones es el resultado de una búsqueda estética por parte del artista? ¿Acaso no habremos querido que una canción suene de tal manera y la siguiente de otra? Si nuestra búsqueda como artistas es la de presentar estas diferencias en sonoridad, lo que muy probablemente suceda (si no nos anticipamos a ello) es que las plataformas de difusión donde carguemos nuestros archivos de audio, realicen una normalización de los mismos.

La normalización llevada a cabo por la mayoría de los servicios de difusión musical (así como televisivo, radial, etc.) se realizan en función de la recomendación antes mencionada (BS.1770- 4). Esto significa, que cada plataforma puede utilizar un algoritmo para medir los niveles de pico y de sonoridad relativos de un archivo que contenga audio, y de allí tomar las decisiones que considere óptimas para equilibrar la sonoridad en función del resto de archivos de su repositorio.

En resumidas cuentas, lo que sucede cuando subimos un archivo a Spotify o YouTube, es que, si no fuimos cuidadosos, nuestra música se normalizará, ya sea subiendo o bajando su sonoridad.

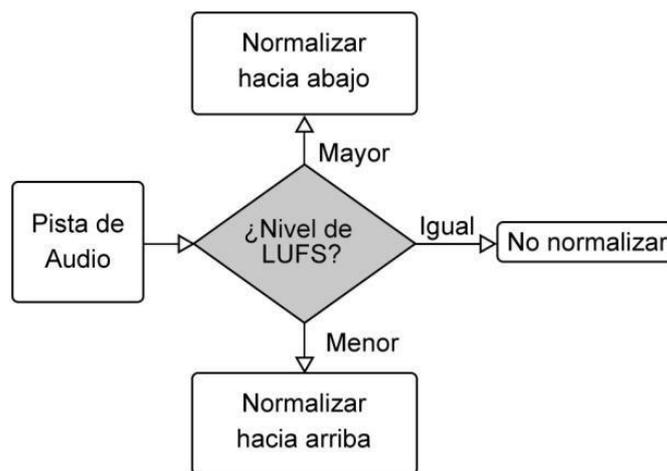
Como la sonoridad es una cualidad psicoacústica del sonido (Miyara, 2006, p. 20), es decir que es un parámetro subjetivo del mismo y que depende de nuestro sistema auditivo, los algoritmos que miden la sonoridad o el pico-real toman en cuenta: estándares en respuesta de sonoridad para distintas frecuencias, el intervalo temporal necesario para realizar medición de valores a corto tiempo, el modo de percibir la sonoridad en función de los picos y los promedios de sonoridad, etc. En general, estos valores surgen de estudios realizados en muchas personas, pero no dejan de ser valores subjetivos.

Es a partir de estos algoritmos como se llega al concepto de *Unidades de Sonoridad relativas a Fondo de Escala*, más conocido como LUFS (Loudness Units relative to Full Scale). Los LUFS son actualmente la unidad de medida estandarizada por la mayoría de los servicios de distribución musical para medir los niveles de sonoridad de pistas de audio. La escala de las LUFS, surge a partir de los complejos algoritmos de medición de sonoridad y busca crear un modelo que estandarice los valores (Taylor, 2012, párr. 7). Para no enredarnos más en nuevos conceptos, vamos a decir que cada plataforma [Tabla 5] tiene valores de LUFS a los que va a intentar acomodar los tracks de audio de nuestros archivos. Veamos la siguiente tabla (Masteringthemix, 2016) para dejar esto más claro.

Sevicio	Niveles aproximados
Apple Music	~ -16 LUFS
Amazon Music	~ -14 LUFS
YouTube	~ -13 LUFS
Spotify	~ -14 LUFS
Tidal	~ -14 LUFS

**Tabla 5:** Valores aproximados de LUFS para diferentes servicios.

Como se puede observar, cada servicio de difusión musical tiene un valor diferente de LUFS. ¿Pero esto que significa en definitiva? Lo que significa es que, si nuestro track se encuentra por encima de ese valor, la plataforma lo va a normalizar hacia abajo (con valores de ganancia inferiores a 1), y si se encuentra por debajo de dicho valor lo va a normalizar hacia arriba (con valores de ganancia mayores a 1)<sup>39</sup>. En definitiva, lo que sucederá es que, si nuestra canción *suena suave*, la plataforma subirá su *volumen*, pero si nuestra canción se pasa en decibelios, reducirá su “volumen”, como se observa a continuación [Imagen 5].



**Imagen 5:** Algoritmo que decide si normalizar o no una pista de audio, y cómo hacerlo en función del nivel de LUFS

39 A diferencia de la normalización hacia abajo, no todos los servicios normalizan hacia arriba, por lo que pueden haber tracks que presenten menos sonoridad respecto de otros en algunas plataformas como Amazon music.

Habiendo entendido como funciona la detección de este nivel en las plataformas, una medida recomendada para tomar al momento de la masterización de nuestra música es considerar tener un plug-in que mida el nivel de LUFS de nuestro *master*. Esto nos va a permitir lograr un sonido controlado, con el rango dinámico y los matices que buscamos, teniendo la certeza de que la plataforma en la que subamos nuestra música no la normalizará<sup>40</sup>.

Por último, cabe destacar que existen servicios, como Spotify, que nos dejan controlar algunos de los parámetros que venimos estudiando, como la compresión o la normalización. Este servicio nos permite elegir la calidad de la música para la transmisión y para la descarga de archivos, nos permite utilizar un ecualizador y decidir si queremos o no que nuestra música se normalice. Si bien, a diferencia de YouTube, Spotify tiene mayores complicaciones al momento de analizar los repositorios y los archivos que efectivamente estamos descargando<sup>41</sup> u oyendo, la [Tabla 6] (proporcionada por Spotify) nos brinda información sobre las calidades proporcionadas por el servicio.

		Servicio Gratuito	Servicio Premium
<b>Reproductor Web</b>		AAC 128 kbit/s	AAC 256 kbit/s
<b>PC, móvil y tablet</b>	<b>Auto.</b>	Se ajusta a la conexión de red	Se ajusta a la conexión de red
	<b>Baja</b>	~24 kbit/s	~24 kbit/s
	<b>Normal</b>	~96 kbit/s	~96 kbit/s
	<b>Alta</b>	~160 kbit/s	~160 kbit/s
	<b>Muy Alta</b>	<i>no disponible</i>	~320 kbit/s

**Tabla 6:** Diferentes calidades en función del tipo de reproductor y el servicio contratado de Spotify (la notación kbit/s en lugar de kbps es propia de la plataforma). La calidad "baja" no se encuentra disponible en la app de ordenadores para Windows

## Algunas conclusiones

Llegados a este punto consideramos que, luego de haber estudiado los distintos procedimientos, entendidos como algoritmos, que forman parte de las etapas de producción y distribución musical, hemos sumado algunas herramientas para comprender como se transforma nuestra música al pasar por las distintas instancias o plataformas musicales. Estas herramientas nos podrán ser de gran utilidad como músicos y músicas, si es que las usamos de forma crítica y en función de nuestra búsqueda estética.

<sup>40</sup> Aquí cabe aclarar que solo hablamos del proceso de normalización. Como vimos anteriormente, las plataformas comprimen los archivos por lo que hay un detrimento en la calidad sonora de los mismos. Controlando los LUFS, nos aseguramos que no se normalice únicamente.

<sup>41</sup> Debido principalmente a que los archivos que se descargan están codificados en un formato desarrollado por Spotify cuyo códec solo se encuentra en la aplicación.

Conociendo los protocolos llevados a cabo por las plataformas de distribución musical para detectar los LUFS, podemos incluir, en nuestra etapa productiva, plug-ins que nos permitan contabilizar esta unidad de sonoridad para que luego, al momento de subirla a alguna plataforma, no se apliquen algoritmos de normalización sobre nuestro material.

Entendiendo el modo en que se descarta información sonora al momento de una compresión con pérdida, podemos optar por el compresor (códec / formato de audio) que nos sea más adecuado, en tanto responda a nuestra demanda en términos de calidad acústica. Así, podemos optar por Opus o AAC y decidir si nuestro archivo será mp3 u ogg, por ejemplo, también decidiendo la calidad de salida del compresor en función de la plataforma a la cual vamos a subir el archivo.

Comprendiendo los subprocesos que conlleva la digitalización de una señal acústica, podemos comprender lo que representan los valores al momento de decidir la calidad de la digitalización. Cuando configuremos nuestros programas o bien veamos las propiedades de un archivo de audio, sabremos que  $44.1\text{ kHz}$  representa una frecuencia en la que se tomaron (o se grabaron) muestras, y que  $16\text{ bits}$  es la resolución de bits, es decir el número discreto de valores que puede adoptar una muestra, que es igual a  $2^{16}$ .

Cada uno de estos procesos tiene un correlato auditivo, o bien psicoacústico. La normalización de una pista de audio hará que esta se escuche con mayor o menor intensidad que la pista original. La compresión con pérdida hará que se pierdan bandas, por lo general superiores, lo que dará como resultado una pista con ruido en altas frecuencias producto del recorte de las mismas en la pista original. La digitalización a valores bajos de resolución de bits hará que sea audible el error (ruido) de cuantificación, y a valores inferiores a  $44.1\text{ kHz}$  de muestreo, hará que las frecuencias cercanas a la banda crítica superior se distorsionen.

## Referencias

- Gupta, P. C. (2006). *Data Communications and Computer Networks*. New Delhi: PHI Learning.
- Johansen, K. (26 de Abril 2019) - *Cuentas Claras* - [Archivo de Video]  
<https://www.youtube.com/watch?v=3kaZLV1DlcY>
- Masteringthemix. (30 de Agosto de 2016). Song mastering for CD, Club Play and Streaming.  
*masteringthemix.com*. Recuperado de <https://www.masteringthemix.com/blogs/learn/song-mastering-for-cd-club-play-and-streaming>.
- Miyara, F. (2006). *Acústica y sistemas de sonido*. Rosario: UNR Editor.
- Rumsey F. y McCormick T. (1994). *Introducción al sonido y la grabación*. Madrid: Instituto Oficial de Radio Televisión Española
- Segura i Navarro, A. y Tébar-Martínez, E. M. (2003). *Tecnología de las comunicaciones móviles*. Alicante: Universidad Miguel Hernández.
- Taylor, G. (20 de Febrero de 2012). What's the difference between LKFS and LUFS?.  
*gameaudionoise.blogspot.com*. Recuperado de  
<http://gameaudionoise.blogspot.com/2012/02/whats-difference-between-lkfs-and-lufs.html>