

Tema 7

Compresión de bioseñales

7.1. Introducción

Un sistema típico de procesado de señales biomédicas adquiere una gran cantidad de datos que deben almacenarse o transmitirse. Es necesario aplicar algún método para reducir el espacio de almacenamiento preservando el contenido significativo de la información para poder reconstruir posteriormente la señal. En algunas aplicaciones, este proceso de compresión/descompresión tendrá que realizarse en tiempo real. Existen por tanto tres factores que caracterizan a los algoritmos de compresión de datos:

- Eficiencia de compresión: el objetivo es minimizar el número de bits de código almacenados eliminando redundancias presentes en la señal original. Se define la razón de compresión (RC) como el cociente entre el número de bits de la señal original por el número de bits almacenado en la señal comprimida. Factores tales como la anchura de banda, frecuencia de muestreo y fidelidad tienen un efecto importante sobre la razón de compresión.
- Fidelidad de reconstrucción: Un algoritmo de compresión de datos debe representar los datos con una fidelidad aceptable. En el campo de la Ingeniería Biomédica, es frecuente validar la aceptabilidad clínica de la señal reconstruida mediante inspección visual. Se pueden medir también los residuos, es decir, la diferencia entre la señal reconstruida y la original mediante parámetros como el PRD (el tanto por ciento de la raíz cuadrada de las diferencias medias al cuadrado), definido como:

$$\text{PRD} = \left\{ \frac{\sum_{i=1}^N [x_{\text{org}}(i) - x_{\text{rec}}(i)]^2}{\sum_{i=1}^N [x_{\text{org}}(i)]^2} \right\}^{1/2} \times 100\%$$

donde N es el número de muestras y las x_{org} y x_{rec} representan las muestras originales y reconstruidas, respectivamente.

- **Complejidad:** esta característica incide directamente en el tiempo de cálculo del algoritmo, y por tanto influye fundamentalmente en el caso de implementaciones en tiempo real.

La elección de un algoritmo de compresión se basa en un balance de las características comentadas, en función de la aplicación concreta.

Existen dos grupos de algoritmos de compresión:

1. **Algoritmos sin pérdidas:** producen un residuo cero, y la señal reconstruida es idéntica a la original. Se suelen basar en algún tipo de codificación.
2. **Algoritmos con pérdidas:** en este caso, la reducción de datos implica que la señal reconstruida no es exactamente igual a la original. Generalmente las pérdidas (los residuos) corresponden a características no fundamentales de la señal, por lo que se puede obtener una calidad aceptable clínicamente. Por ejemplo, un algoritmo de compresión de ECG puede eliminar oscilaciones basales de pequeña amplitud. En este caso, el residuo contiene información no interesante desde el punto de vista clínico. La señal reconstruida puede ser bastante aceptable clínicamente a pesar de tener un residuo alto. Hay tres clases de algoritmos con pérdidas:
 - **Algoritmos de compresión en el dominio del tiempo:** se basan en la extracción de puntos significativos, guardando muestras que contienen información importante sobre la señal y descartando el resto.
 - **Algoritmos de compresión por transformación:** se basan en la estimación de la información relevante de la señal en algún dominio transformado.
 - **Algoritmos de compresión por extracción de características:** se basan en obtener una representación de la señal según un conjunto de parámetros (características) que definen la información relevante. Suelen utilizarse en sistemas de clasificación automática, ya que esta representación más compacta simplifica la tarea del clasificador al comparar con los patrones de cada clase. La reconstrucción de la señal debe hacerse mediante técnicas de interpolación.

La comparación de prestaciones entre distintos algoritmos no es fácil, puesto que suelen aplicarse a registros con características diferentes (frecuencia de muestreo, ancho de banda, resolución de las muestras y nivel de ruido), que afectan directamente a las medidas

de rendimiento. En las siguientes secciones se revisan brevemente algunas de las técnicas más usuales.

7.2. Algoritmos sin pérdidas

La principal ventaja de los algoritmos sin pérdidas es que la señal puede reconstruirse sin error. No obstante, el bajo valor de RC que suele obtenerse (del orden de 2:1), hace que no se utilicen generalmente en el campo de bioseñales, o al menos como único método de compresión. En ocasiones, se aplican a la señal ya comprimida por un algoritmo con pérdidas (que permite mayor RC), para obtener un incremento de compresión adicional.

Codificación Huffman

La codificación Huffman es una de las más usuales, y se basa en el hecho de que las distintas amplitudes discretas de la señal digitalizada no ocurren con igual probabilidad. Asigna una palabra codificada de longitud variable a una muestra de acuerdo con su probabilidad de ocurrencia. Los valores más frecuentes tienen asignadas palabras de código más cortas.

Su implementación requiere una tabla de conversión, donde a cada posible valor de la muestra se le hace corresponder una única palabra código. Si los datos originales son de 16 bits, la tabla necesitaría 65536 posiciones. Una tabla de este tamaño crea problemas de memoria e ineficacia de proceso.

Para reducir el tamaño de la tabla, el esquema de la codificación Huffman modificada realiza particiones de los datos en un conjunto frecuente y uno infrecuente. Para los símbolos del frecuente, se genera un código Huffman como en el caso anterior. Utilizamos una palabra código especial como prefijo para indicar símbolo infrecuente y añadimos un sufijo correspondiente a la codificación binaria ordinaria del símbolo.

Codificación adaptativa

La codificación Huffman requiere generar una tabla para lo cual es necesario examinar el conjunto completo de datos para determinar las estadísticas de los mismos. La tabla de conversión debe ser transmitida o almacenada para permitir la correcta descompresión.

Un esquema de codificación adaptativa intenta construir la tabla según llegan los datos. Una tabla de conversión obtenida dinámicamente es sensible a la variación de la información de la estadística local. Puede por tanto alterar sus palabras código de acuerdo con la estadística local para maximizar la compresión. Además proporciona espacio extra puesto que no se necesita una tabla estática.

Un ejemplo de codificación adaptativa es el algoritmo de Lempel-Ziv-Welch (LZW), que utiliza una tabla de tamaño fijo. Inicializa algunas posiciones de la tabla para algunos conjuntos de datos. Cuando encuentra nuevos datos, utiliza las posiciones no utilizadas, por lo que cada palabra de datos tiene asignada su propia posición. Cuando la tabla está llena, el LZW reinicializa la posición más antigua o menos utilizada de acuerdo con los nuevos valores. Durante la reconstrucción, rehace incrementalmente la tabla de conversión a partir de los datos codificados.

Diferencia residual

Generalmente, las muestras contiguas de la señal no son estadísticamente independientes, sino que están relacionadas entre sí. Por tanto, podemos descomponer una muestra en una parte que está correlacionada con las muestras anteriores y otra parte que no lo está. La componente correlacionada puede predecirse mediante algún tipo de predictor, por lo que la parte no correlacionada representará el error de predicción o señal residual. Puesto que el rango de amplitudes de la señal residual es generalmente mucho menor que el de la señal original, requiere menos bits para su representación. Podemos reducir todavía más aplicando codificación Huffman a la señal residual.

Hamilton y Tompkins (1991) usan el hecho de que un ECG típico está compuesto de patrones repetitivos con pequeños cambios pulso a pulso. El algoritmo calcula y actualiza un pulso promedio. Cuando se detecta un pulso, se alinea y sustrae el detectado del patrón. La señal residual es codificada Huffman y almacenada con información sobre la localización del pulso. Finalmente, el algoritmo usa el pulso detectado para actualizar el patrón.

Codificación '*run-length*'

Utilizada en tecnología de facsímil, esta codificación se basa en la alta correlación entre bits sucesivos en una línea de bits de un facsímil. En este caso, los valores serán 1 ó 0, dependiendo de si el pixel es blanco o negro. En un documento típico, existen zonas de pixeles blancos y negros que proporcionan esta alta correlación. Esta codificación simplemente transforma las líneas de bits en cadenas del tipo $\{v_i, l_i\}$, donde v_i son los

valores y l_i las longitudes. Se puede obtener una mayor compresión aplicando codificación Huffman a la salida del codificador 'run-length'.

7.3. Compresión en el dominio del tiempo

Los algoritmos de compresión en el dominio del tiempo tienen como objetivo seleccionar aquellos puntos característicos de la señal, eliminando la información redundante. Utilizan conocimiento previo sobre la señal a comprimir. A continuación se comentan algunos de ellos.

Algoritmo AZTEC

El algoritmo AZTEC (*Amplitude Zone Time Epoch Coding*) fue desarrollado originalmente para preprocesar ECG en análisis de ritmo. Descompone las muestras en pendientes y tramos rectos, proporcionando una secuencia de segmentos que producen una aproximación lineal a tramos del ECG.

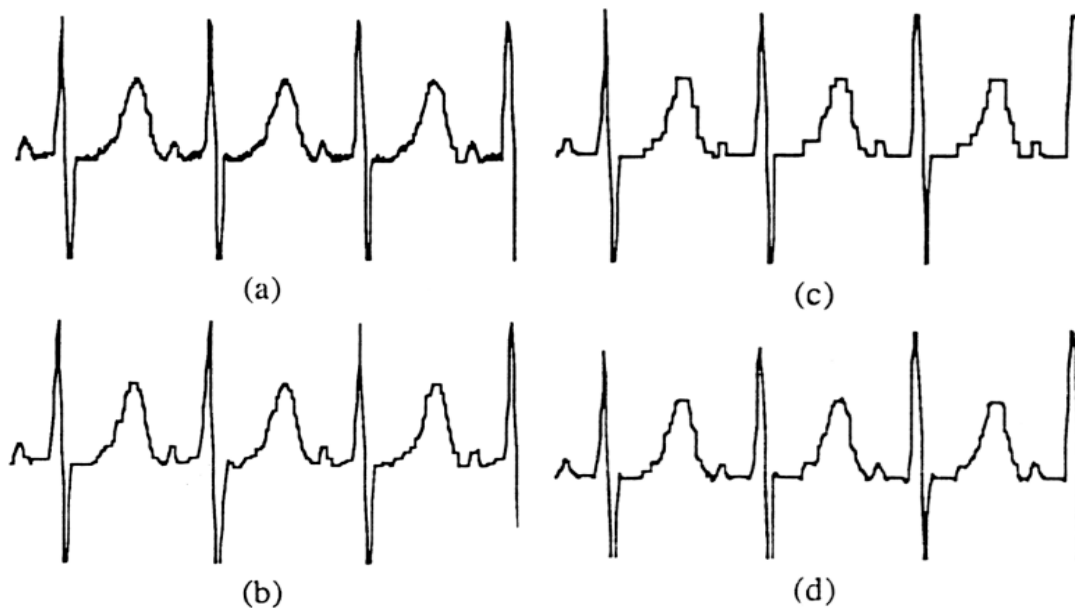


Figura 7.1. Algoritmo AZTEC. a) ECG original. b) Umbral pequeño, razón de compresión 512:233, PRD=24.4%. c) Umbral grande, razón de compresión 512:153, PRD=28.1%. d) Señal filtrada, promediador de 7 puntos, PRD=26.3%. (W. J. Tompkins Ed. "Biomedical Digital Signal Processing". Prentice Hall)

Para cada tramo recto se guarda el número de muestras y la amplitud media. Para cada pendiente se guarda el número de muestras y la amplitud final. Se distingue entre recta y pendiente porque el número de muestras del primero se guarda con signo + y el de la segunda con signo -. El umbral o máximo error permitido en un tramo determina el intervalo de amplitudes de las muestras que se consideran pertenecientes al mismo segmento, y está directamente relacionado con la RC e inversamente con la fidelidad de reconstrucción que proporciona el algoritmo. AZTEC no produce una reducción fija de datos. La razón suele ser mayor que 10, dependiendo de la señal y del valor del umbral seleccionado.

El resultado generado es una secuencia alternante de duraciones y amplitudes. Los datos se reconstruyen expandiendo los tramos rectos y las pendientes en muestras discretas. Este proceso de reconstrucción produce una señal con perfil de escalera, lo que no resulta clínicamente aceptable. Por lo tanto requiere postprocesado mediante un filtro de suavizado o pasa-bajo, que reduzca las discontinuidades. Aunque la salida mejora en apariencia, introduce distorsión en amplitud.

Algoritmo Fan

Usado originalmente para biotelemedicina, el FAN traza líneas entre pares de puntos inicial y final, de manera que las muestras intermedias están dentro de un error predeterminado, ϵ .

La figura 7.2 ilustra el método utilizado por el algoritmo. Se toma una muestra inicial X_0 . A continuación se determinan las rectas U_1 y L_1 , que pasan por los puntos $(X_0, X_1+\epsilon)$ y L_1 pasa por $(X_0, X_1-\epsilon)$, respectivamente. Si X_2 está localizada dentro del área que subtienden ambas rectas, se generan otras dos, U_2 y L_2 , que pasan por los puntos $(X_0, X_2+\epsilon)$ y $(X_0, X_2-\epsilon)$. De las cuatro rectas generadas, se seleccionan las que encierran menor área (U_1 y L_2 en el ejemplo de la figura). El proceso se repite hasta que se encuentra una muestra localizada fuera del área definida por las rectas. En ese caso, se guarda el número de muestras procesadas desde X_0 , y la amplitud de la última muestra, que se convierte en el nuevo X_0 para la siguiente iteración.

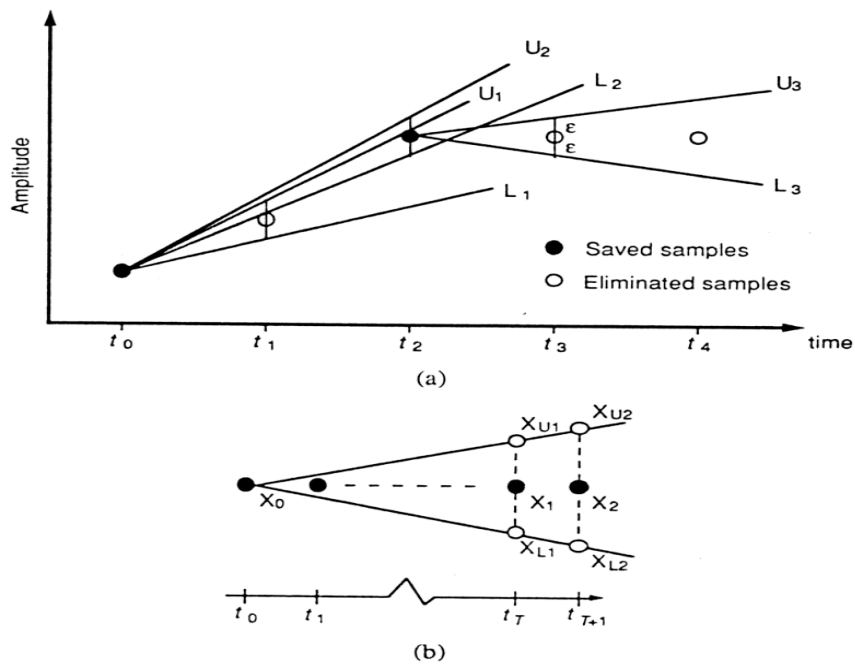


Figura 7.2. Algoritmo FAN. a) Pendientes superior e inferior (U, L) trazadas dentro de un umbral de error ϵ alrededor de las muestras. b) Extrapolación de X_{U2} y X_{L2} desde X_{U1} y X_{L1} y X_0 . (W. J. Tompkins Ed. "Biomedical Digital Signal Processing". Prentice Hall)

La señal se reconstruye expandiendo los valores obtenidos en muestras discretas. La razón de reducción depende de la tolerancia de error. La siguiente figura muestra el mismo ECG procesado con AZTEC en la figura 7.1, comprimido en este caso con Fan. Como puede observarse, Fan produce mejor fidelidad de señal que AZTEC para igual razón de reducción.

Existen 3 algoritmos basados en la misma aproximación utilizada por Fan. El SAPA-2 (*Scan-Along Approximation*) produce los mejores resultados de los tres. Como en el Fan, SAPA-2 garantiza que la desviación entre la señal reconstruida y la original nunca excede el error preseleccionado.

Además de las dos pendientes calculadas por el Fan, SAPA-2 calcula una tercera pendiente denominada central entre la muestra origen y la actual. Cada vez que la pendiente central no está comprendida entre las dos pendientes de convergencia, la muestra inmediatamente precedente se toma como punto origen. Por tanto, la diferencia principal entre SAPA-2 y Fan es que el primero utiliza un criterio de pendiente central en lugar del de valor de la muestra actual.

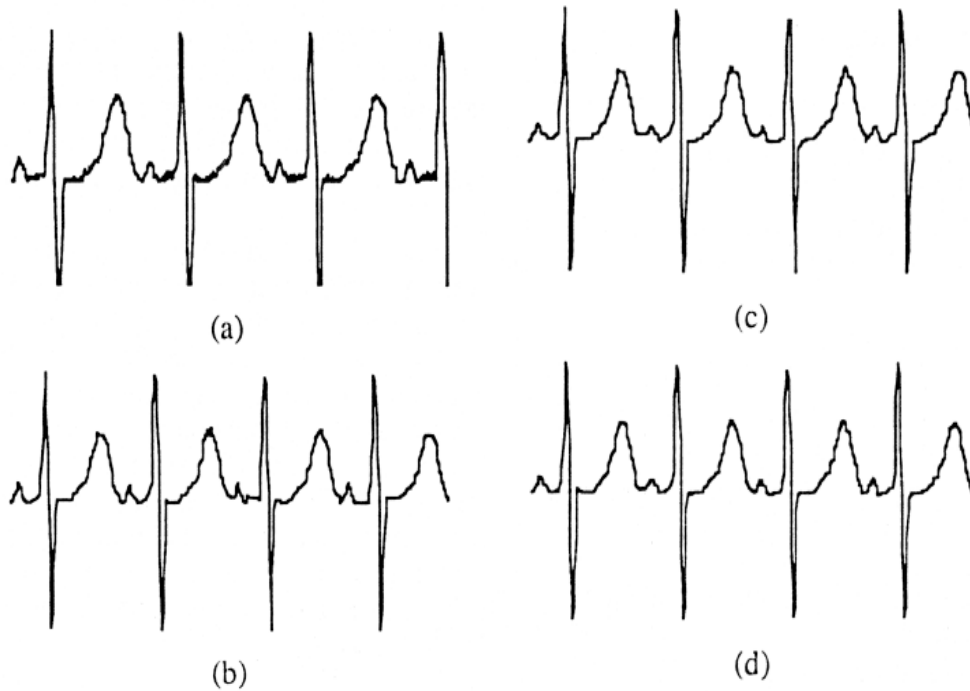


Figura 7.3. Aplicación del algoritmo FAN. a) ECG original. b) Tolerancia pequeña: razón de reducción 512:201 PRD=5.6%. c) Tolerancia grande: razón de reducción 512:155 PRD=7.2%. d) Señal filtrada de c): promediador de 3 puntos, PRD=8.5%. (*W. J. Tompkins Ed. "Biomedical Digital Signal Processing". Prentice Hall*)

7.4. Compresión por transformación

En el caso de compresión por transformación, el proceso consiste en aplicar la transformada a la señal, evaluar el resultado y seleccionar sólo aquellas componentes que representan el mayor porcentaje de la potencia de la señal. Para realizar la descompresión, se aplica una transformación inversa.

Compresión en el dominio de la frecuencia

Para transformaciones en el dominio de la frecuencia, la señal se divide en bloques de muestras antes de aplicar la transformada a cada uno de ellos. Dentro de este grupo se

encuentran la transformada discreta de Fourier (DFT), la de Karhunen-Loeve (KLT) y la transformada discreta del coseno (DCT).

La DCT es una de las más utilizadas cuando existe alta correlación entre las muestras de entrada, lo que suele ocurrir en el caso de bioseñales, imágenes, voz, etc. Se define como:

$$v_0 = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} x_n$$
$$v_k = \sqrt{\frac{2}{N}} \sum_{n=0}^{N-1} x_n \cos \frac{(2n+1)k\pi}{2N}; k = 1 \dots N-1$$

donde v_k es el k -ésimo coeficiente. La correspondiente transformada inversa (IDCT) viene dada por:

$$x_n = \frac{1}{\sqrt{N}} v_0 + \sqrt{\frac{2}{N}} \sum_{k=0}^{N-1} v_k \cos \frac{(2n+1)k\pi}{2N}; k = 0, 1 \dots N-1$$

Al igual que el en caso de la FFT, existe un algoritmo rápido para el cálculo de la DCT. Dos estándares de compresión de imagen y vídeo (JPEG y MPEG) se basan en la DCT. En compresión de ECG se han obtenido RC de hasta 26:9.

Compresión mediante transformada *wavelet*

La compresión mediante transformada *wavelet* se basa en la posibilidad de representar la señal original mediante algunos coeficientes de aproximación y detalle de la descomposición obtenida mediante la transformada. Los términos de aproximación son las componentes de baja frecuencia y alto valor de escala, mientras que los de detalle corresponden a las frecuencias altas (escalas bajas). El proceso de descomposición de la señal puede verse, por tanto, como un filtrado sucesivo pasa-bajo y pasa-alto, de múltiples niveles, que para el caso de la transformada discreta proporciona el árbol de descomposición (figura 7.4).

El proceso de compresión se realiza en tres etapas. En primer lugar, se calcula la descomposición de la señal mediante la aplicación de una determinada *wavelet* para un cierto nivel N . A continuación, se determinan los umbrales para las componentes de detalle

correspondientes a todos los niveles desde 1 a N. Por último, la reconstrucción de la señal comprimida se realiza utilizando los coeficientes de aproximación originales para nivel N y los de detalle modificados tras la aplicación del umbral desde 1 hasta N.

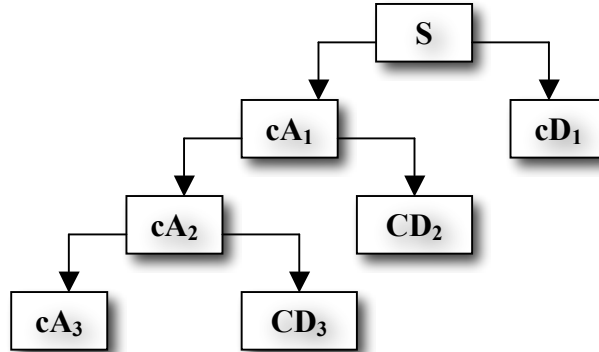


Figura 7.4. Árbol de descomposición de la transformada *wavelet* para una señal S. cA y cD representan los coeficientes de aproximación y detalle, respectivamente. Los subíndices indican en nivel de descomposición.

El método de elección de umbrales en el paso 2 da lugar a dos aproximaciones diferentes de compresión. En un caso, se aplica un umbral global para todas las componentes quedándose con los coeficientes de mayor valor absoluto. En otro, se aplican umbrales dependientes del nivel, que pueden ser ajustados por separado proporcionando un mayor control sobre el proceso.

La siguiente figura muestra un ejemplo de compresión de ECG mediante TW. Se ha utilizado una *wavelet* madre de Daubechies tipo 5 y se ha seleccionado un nivel de descomposición $N=6$ (figura 7.5.a). A continuación se aplican umbrales dependientes del nivel a las componentes de detalle (figura 7.5.b), y se anulan aquellos coeficientes de la transformada que no superen los correspondientes umbrales. En la figura 7.5.c pueden verse los coeficientes originales (central), los coeficientes resultantes, muchos de los cuales son cero (inferior), y la superposición de la señal original y reconstruida (superior). El porcentaje de energía de la señal original retenida por los coeficientes resultantes es del 99.02%, y el de coeficientes nulos respecto de los originales es del 94.69%.

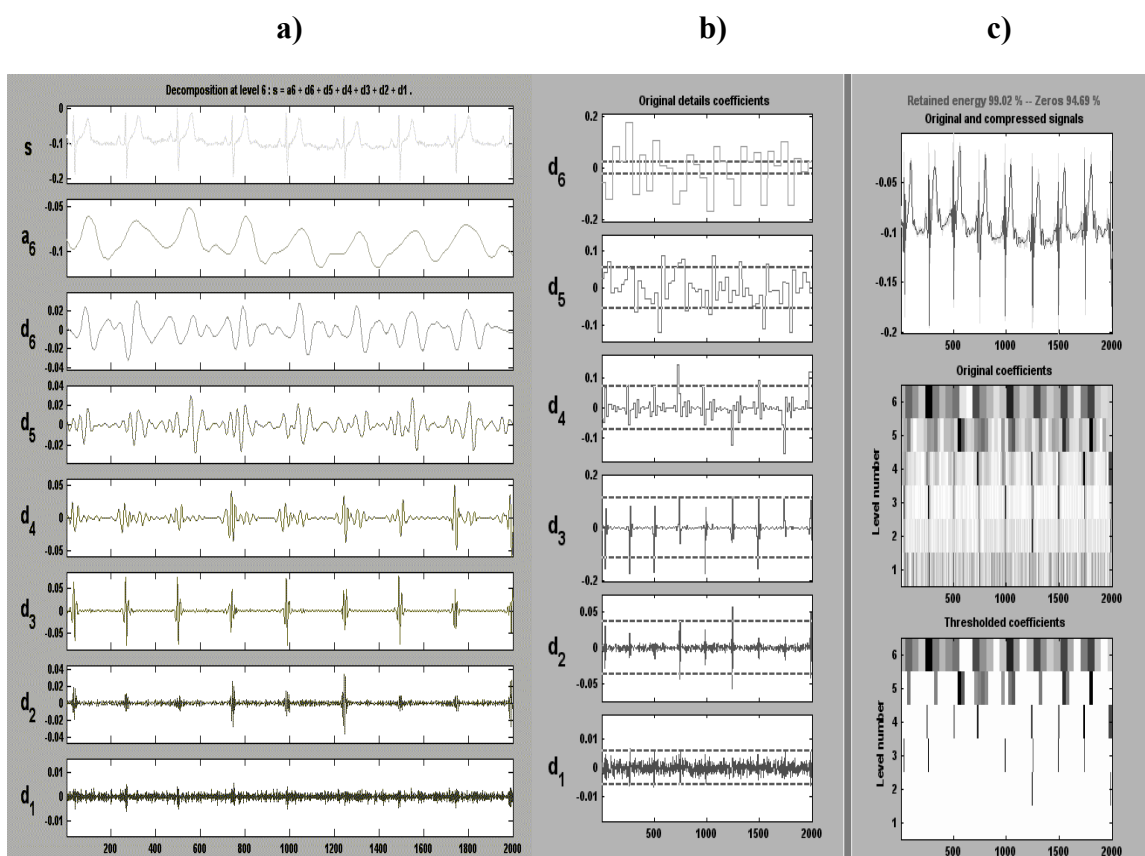


Figura 7.5. Compresió de ECG mediante transformada *wavelet*. a) Señal original y descomposició mediante *wavelet* Daubechies tipo 5. b) Umbrals seleccionados para los coeficientes originales. c) Señal original y reconstruida, y representación de los coeficientes originales y finales.