
Diagnóstico de enfermedades del corazón a través de un ECG

Universidad de Murcia
4º Grado en Ingeniería Informática
Asignatura: Desarrollo de Sistemas Inteligentes
Curso 2019/2020

Proyecto realizado por:

Nicolás Enrique Linares La Barba

Rubén Marín García

Índice

1	Contexto	3
2	Objetivo	3
3	Datos de entrada: log ECG	3
4	Fuente de conocimiento experto	4
5	Decisiones de diseño	4
5.1	Diseño estructural del Ciclo Cardíaco	4
5.2	Lectura del fichero de entrada	5
5.3	Diseño de las reglas	5
5.3.1	Cálculo de componentes	6
5.3.2	Diagnóstico de enfermedades	6
5.3.3	Mostrar enfermedades detectadas	11
6	Manual de uso	11
7	Referencias	12

1 Contexto

El corazón es un órgano vital y su estudio de su morfología y comportamiento permite al médico poder identificar posibles enfermedades. El electrocardiógrafo hoy en día es una herramienta fundamental para estudiar el funcionamiento del corazón, de una forma eficaz, rápida y barata. Esencialmente el electrocardiógrafo registra la actividad eléctrica de los músculos del corazón, que son representados como una serie temporal e interpretados por un médico.

El procesamiento automático del ECG comienza con el procesamiento de la señal electrocardiográfica, caracterizando (identificando) las diferentes componentes morfológicas de la señal (ondas, segmentos e intervalos).

Una vez que realizado este procesamiento, es el turno de desarrollar un sistema inteligente capaz de obtener una conclusión diagnóstica mediante un razonamiento automático a partir de la fuente de datos de entrada y una fuente de conocimiento experto.

2 Objetivo

Implementación del prototipo de un sistema de alarmas que identifique, a partir de la interpretación de un ECG, posibles patrones de riesgo para la salud.

3 Datos de entrada: log ECG

Asumiremos que existe un pre-procesamiento de la señal electrocardiográfica cuyo resultado es un fichero donde queda caracterizada de manera secuencial la señal de ECG de diferentes pacientes. A este fichero lo denominamos “log ECG”.

Además, simplificaremos el número de derivaciones del ECG, centrándonos exclusivamente en una derivación (derivación II). **La información que utilizaremos es sesgada y no tendrá una validez clínica real**, sin embargo, desde el punto de vista del tratamiento de la información resulta bastante aproximada.

Formato de log ECG: El log es un fichero ASCII que contiene en cada línea la secuencia caracterizada de etiquetas del ECG de un paciente. Es decir, un paciente por línea. Las etiquetas que pueden estar presente en cada secuencia caracterizada:

P (<start>,<end>,<peak>)

Q (<start>,<end>,<peak>)

R (<start>,<end>,<peak>)

S (<start>,<end>,<peak>)

T (<start>,<end>,<peak>)

U (<start>,<end>,<peak>)

<start> = [0-9]+ comienzo de la onda en milisegundos.

<end> = [0-9]+ fin de la onda en milisegundos.

<peak> = [-][0-9]+.[0-9]+ pico máximo de la onda en miliVoltios.

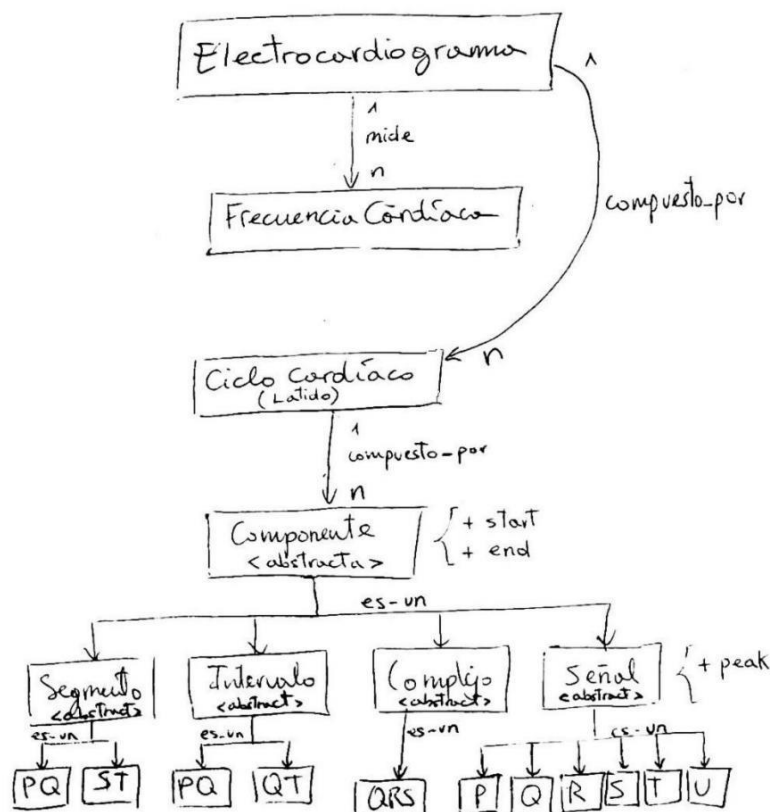
4 Fuente de conocimiento experto

La correcta interpretación de la señal del ECG requiere de una extensa formación y experiencia clínica. Sin embargo, parte del conocimiento esencial puede encontrarse en bibliografía y manuales de cardiología básicos. **En este proyecto, la información clínica utilizada será limitada y/o simplificada para propósitos académicos y, por tanto, sin validez clínica completa.**

5 Decisiones de diseño

5.1 Diseño estructural del Ciclo Cardíaco

La decisión de diseño más importante se encuentra en la forma de estructurar las señales, segmentos, Intervalos y complejos. Se ha optado por crear la siguiente jerarquía de clases:



Se puede observar que cada ciclo (cada latido del corazón), está compuesto principalmente por las señales P, Q, R, S, T, U que representan un rango de tiempo, inicio y final, además de un pico. También tenemos los segmentos, intervalos y complejos, que al igual que las señales tienen un inicio y un final. A su vez se van a utilizar subclases de estas componentes (segmentos, intervalos y complejos), que serán calculadas a partir de las reglas.

Con esta idea se ha abstraído a una clase padre *Componente*. Un componente no es más que un elemento de un *Ciclo*, compuesto por un tiempo inicial y un tiempo final. Solo en la clase *Señal* se ha añadido un atributo más que es el pico de la onda.

5.2 Lectura del fichero de entrada

Se ha creado una clase *Electrocardiograma* (se puede ver en el esquema anterior) que se va a encargar de almacenar los datos leídos del fichero de entrada. La clase *LectorFichero* será llamada también del *main* para realizar la lectura y el *parseo* de cada línea del fichero y guardar los datos.

Se obtienen los datos relevantes mediante expresiones regulares, como son los bpm (heart rate) y el número de ciclos especificados en las primeras líneas del fichero, así como el nombre de la señal leída y el resto de sus datos.

A continuación, se van construyendo las señales leídas (P, Q, R, S, T, U) y se añaden al conjunto que forma los hechos iniciales.

5.3 Diseño de las reglas

En primer lugar, se insertan los datos leídos (hechos iniciales) en la *kieSession*: la lista de *Señales* de cada tipo y la *FrecuenciaCardiaca* que contiene el número de ciclos y el bpm. A continuación, se aplican las reglas en el orden que indica los grupos de agendas, en los cuales, hemos insertado tres grupos de reglas diferentes:

```
Agenda agenda = kSession1.getAgenda();
agenda.getAgendaGroup( "mostrar" ).setFocus();
agenda.getAgendaGroup( "deteccion" ).setFocus();
agenda.getAgendaGroup( "insertar" ).setFocus();
```

Los grupos de reglas se disparan en este orden:

1. *insertar*. En las reglas de este grupo se calcula los complejos, intervalos y segmentos a partir de la lista de *Señales* y se insertan en la base de hechos.
2. *detección*. Estas reglas analizan la base de hechos y, según los componentes introducidos anteriormente, se busca las posibles enfermedades y, en caso de encontrarlas, también se insertan en la base de hechos.
3. *mostrar*. Finalmente, se disparan las reglas para mostrar las enfermedades que se encuentren en la base de hechos.

5.3.1 Cálculo de componentes

Se han diseñado 5 reglas para el cálculo de los componentes de cada ciclo:

- `rule "Calcular Segmento PQ"`
- `rule "Calcular Segmento ST"`
- `rule "Calcular Intervalo PQ"`
- `rule "Calcular Intervalo QT"`
- `rule "Calcular Complejo QRS"`

Estas reglas siguen la siguiente estructura:

```
rule "Calcular Segmento PQ"
  agenda-group "insertar"
  when
    $sP : P()
    $sQ : Q(getIdCiclo() == $sP.getIdCiclo())
  then
    insert(new SegmentoPQ($sP.getEnd(), $sQ.getStart(), $sP.getIdCiclo()));
  end
```

En el **when** se comprueba que las señales se encuentran en el mismo ciclo (ya que se ha ido guardando en cada señal el ciclo al que pertenece) y si la condición es válida se inserta el componente en ese mismo ciclo.

5.3.2 Diagnóstico de enfermedades

Para la detección de enfermedades se busca anomalías en la duración de los componentes, así como la frecuencia cardíaca. Para ello, se compara estos valores con los valores normales que debería tener cada componente (es decir, los valores en cada uno de estos componentes que debería tener un corazón sano).

Estos valores normales se pueden obtener, por ejemplo, en la Wikipedia, donde podemos ver entre qué rangos de valores para los diferentes componentes oscila un ECG normal [7]. No obstante, ya que los ficheros que se nos proporciona en la práctica pueden variar un poco sobre estos valores, lo que hacemos es comprobar que, en los ficheros de ejemplo de corazón sano, efectivamente, esos rangos de valores se verifiquen.

La duración de los componentes (correspondientes a un diagnóstico sano) en los que nos basamos para detectar las enfermedades son:

- **Complejo QRS.** La duración normal del complejo QRS es de entre 80-120 milisegundos. [1] [2] La duración corta de este complejo puede provocar

enfermedades como taquicardia ventricular y contracción prematura ventricular. Por otro lado, una duración larga puede provocar hipocalcemia, hipopotasemia o, también taquicardia ventricular.

- **Intervalo PQ.** La duración normal, es de entre 120 y 200 milisegundos. En nuestros ficheros de ejemplo es de 190 milisegundos. [3] El acortamiento del intervalo puede ser causa de enfermedades como el síndrome de Wolff-Parkinson-White, en cambio, su alargamiento puede causar hipopotasemia o hipotermia.
- **Segmento ST.** Su duración normal suele ser en un rango de entre 80-140 milisegundos. [4] Un valor inferior a este podría ser causa de hipocalcemia y un valor superior podría significar hipocalcemia.
- **Señal T.** La onda T puede ser positiva o negativa. [5] Las ondas T con un pico positivo son la causa de hipopotasemia, en cambio, las de pico negativo pueden causar isquemia coronaria o hipopotasemia.

También se estudia la frecuencia cardíaca del ECG, que en valores normales se encuentra sobre los 60 y 100 latidos por minuto. Por debajo de 60 es posible causa de bradicardia [6] y si supera los 100 latidos por minuto, puede ser causa de taquicardia.

En la lectura de un electrocardiograma también influye el valor de las duraciones del resto de ondas, intervalos y segmentos. Tras varias pruebas realizadas con los ficheros de las distintas enfermedades facilitados en el aula virtual, hemos concluido que los componentes, explicados antes, son los adecuados para el correcto funcionamiento de la práctica. Con estos componentes el programa es capaz de detectar todas las enfermedades de ejemplo.

En base a que las enfermedades comparten causas comunes, se ha optado por no establecer reglas con algunos intervalos y segmentos, por ejemplo, el intervalo QT. La duración prolongada de este componente puede causar hipopotasemia, hipocalcemia o bradicardia. Con el objetivo de obtener una salida más limpia se prescinde de este componente.

Las reglas para la detección de enfermedades son:

- **Bradicardia**

```
rule "Detectar Bradicardia"
    agenda-group "deteccion"
    when
        $F : FrecuenciaCardiaca(getBPM() < 60)
    then
        insert(new Bradicardia("*** Low Heart Rate (Bradicardia) : " +
            $F.getBPM() + " pul/min"));
    end
```

La condición de esta regla es tener una frecuencia cardíaca con un ritmo bajo, menor de 60 latidos por minuto. En la acción se inserta un objeto Bradicardia en la base de hechos.

- **Taquicardia**

```
rule "Detectar Taquicardia"
    agenda-group "deteccion"
        when
            $F : FrecuenciaCardiaca(getBPM() > 100)
        then
            insert(new Taquicardia("*** Unusual Heart Rate (Taquicardia): "
                + $F.getBPM() + " pul/min\n"));
        end
    end
```

La cor
latido e 100

- **Contracción prematura ventricular**

```
rule "Detectar ContraccionPrematuraVentricular"
    agenda-group "deteccion"
        when
            $C : ComplejoQRS( (getEnd()-getStart()) < 80)
        then
            insert(new ContraccionPrematuraVentricular("*** Short QRS complex (Premature Ventricular Contraction) cycle: "
                , $C.getIdCiclo()));
        end
    end
```

La condición de esta regla es tener un complejo QRS menor que el habitual. En nuestro caso, mirando entre que rangos puede oscilar el valor normal y fijándonos en los ficheros relacionados con esta enfermedad, hemos decidido que la condición que se tiene que cumplir es que el complejo QRS sea menor que 80 ms. Como consecuencia se inserta un objeto de tipo ContraccionPrematuraVentricular en la base de hechos.

- **Infarto agudo de miocardio**

```
rule "Detectar InfartoAgudoMiocardio"
    agenda-group "deteccion"
        when
            $T : T(getPeak() > 1)
        then
            insert(new InfartoAgudoMiocardio("*** High T wave peak (Myocardial Infarction) cycle: "
                , $T.getIdCiclo()));
        end
    end
```


La condición de esta regla es tener un pico en la onda T superior a los valores normales que ésta suele presentar. En nuestro caso, fijándonos en los ficheros en los valores típicos del pico de esta onda y fijándonos también en sus valores, pero de los ficheros de la enfermedad, hemos decidido que la regla salte cuando se dé la condición de que el pico sea mayor que 1. Como consecuencia se inserta un objeto de tipo InfartoAgudoMiocardio en la base de hechos.

- **Isquemia coronaria**

```
rule "Detectar IsquemiaCoronaria"
    agenda-group "deteccion"
    when
        $T : T(getPeak() < 0)
    then
        insert(new IsquemiaCoronaria("*** Negative T wave peak (Coronary ischemia) cycle: "
            , $T.getIdCiclo()));
    end
```

La condición de esta regla es tener un pico en la onda T de valor negativo. Como consecuencia se inserta un objeto IsquemiaCoronaria en la base de hechos.

- **Hipocalcemia**

```
rule "Detectar Hipocalcemia"
    agenda-group "deteccion"
    when
        $sST : SegmentoST(getDuracionReal() > 140)
    then
        insert(new Hipocalcemia("*** Long ST present (Hipocalcemia)
            cycle: ", $sST.getIdCiclo()));
    end
```

Esta regla tiene como condición que el segmento ST tenga una duración mayor a 140 milisegundos. Como consecuencia se inserta un objeto Hipocalcemia en la base de hechos.

- **Hipopotasemia**

```
rule "Detectar Hipopotasemia 1"  
  agenda-group "deteccion"  
  when  
    $sST : SegmentoST(getDuracionReal() < 140)  
  then  
    insert(new Hipopotasemia("** Short ST present  
      (Hipopotasemia) cycle: ", $sST.getIdCiclo()));  
  end
```

```
rule "Detectar Hipopotasemia 2"  
  agenda-group "deteccion"  
  when  
    $T : T(getPeak() < 0)  
  then  
    insert(new Hipopotasemia("** Negative T wave peak  
      (Hipopotasemia) cycle: ", $T.getIdCiclo()));  
  end
```

```
rule "Detectar Hipopotasemia 3"  
  agenda-group "deteccion"  
  when  
    $iPQ : IntervaloPQ(getDuracionReal() > 190)  
  then  
    insert(new Hipopotasemia("** Long PQ present  
      (Hipopotasemia) cycle: ", $iPQ.getIdCiclo()));  
  end
```

En nuestro caso, esta enfermedad se puede dar por tres anomalías que no dependen entre sí: segmento ST corto (duración menor de 140 milisegundos), inversión de la onda T (pico negativo) o intervalo PQ ancho (con una duración mayor de 190 milisegundos).

5.3.3 Mostrar enfermedades detectadas

Las reglas para mostrar enfermedades tienen como condición que la enfermedad exista en la base de hechos. Si es así se muestra por la salida la enfermedad, su causa y el ciclo en el que fue detectada. La estructura de estas reglas es la siguiente:

```
rule "Mostrar IsquemiaCoronaria"
    agenda-group "mostrar"
    when
        $Enfermedad : IsquemiaCoronaria()
        not (IsquemiaCoronaria(ciclo<$Enfermedad.ciclo))
    then
        System.out.println($Enfermedad.getCausa() + $Enfermedad.getCiclo());
    end
```

Por otro lado, si el paciente no presenta anomalías en su ECG y, por lo tanto, no se han añadido enfermedades en la base de hechos, entonces se presenta a continuación la regla para indicar que el paciente está sano:

```
rule "Mostrar CorazonSano"
    agenda-group "mostrar"
    when
        not (Enfermedad())
    then
        System.out.println("**No sickness detected");
    end
```

6 Manual de uso

Para facilitar su ejecución se exportará el programa en un fichero ejecutable .jar. Se ejecutará escribiendo en un terminal de comandos el nombre del fichero ejecutable y a continuación como argumento el fichero .ecg.fix que queramos analizar.

Ejemplo de salida de la ejecución del fichero hipopotasemia.ecg.fix:

Fichero: ficheros\hipopotasemia.ecg.fix

SLF4J: Failed to load class "org.slf4j.impl.StaticLoggerBinder".
SLF4J: Defaulting to no-operation (NOP) logger implementation
SLF4J: See <http://www.slf4j.org/codes.html#StaticLoggerBinder> for further details.

** Negative T wave peak (Coronary ischemia) cycle: 1
** Negative T wave peak (Hipopotasemia) cycle: 1

7 Referencias

- [1] Jaakko Malmivuo. Bioelectromagnetism, chapter 19 The Basis of ECG Diagnosis [online]. Available: <http://www.bem.fi/book/19/19.htm>
- [2] AMF. alteración de una onda, complejo o intervalo [online]. Available: http://amf-semfyc.com/web/article_ver.php?id=1083
- [3] <https://www.my-ekg.com/>. Intercalo PQ [online]. Available: <https://www.my-ekg.com/como-leer-ekg/intervalo-pr.html>
- [4] AMF. alteración de una onda, complejo o intervalo [online]. Available: http://amf-semfyc.com/web/article_ver.php?id=1083
- [5] AMF. alteración de una onda, complejo o intervalo [online]. Available: <https://www.my-ekg.com/como-leer-ekg/alteraciones-ondas-intervalos.html>
- [6] Jaakko Malmivuo. Bioelectromagnetism, chapter 19 The Basis of ECG Diagnosis [online]. Available: <http://www.bem.fi/book/19/19.htm>
- [7] Electrocardiography. From Wikipedia, the free encyclopedia. Available: <https://en.wikipedia.org/w/index.php?title=Electrocardiography&oldid=513556137#>