

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 540 159**

21 Número de solicitud: 201331726

51 Int. Cl.:

G06F 19/00 (2011.01)

12

PATENTE DE INVENCION

B1

22 Fecha de presentación:

27.11.2013

43 Fecha de publicación de la solicitud:

08.07.2015

Fecha de modificación de las reivindicaciones:

01.03.2016

Fecha de la concesión:

02.03.2016

45 Fecha de publicación de la concesión:

09.03.2016

73 Titular/es:

**UNIVERSIDAD COMPLUTENSE DE MADRID
(100.0%)**

**Avda. de Séneca, nº 2 Vice-Rectorado de
Investigación
28040 Madrid (Madrid) ES**

72 Inventor/es:

**HIDALGO PÉREZ, José Ignacio;
GARNICA ALCÁZAR, Antonio Oscar;
LANCHARES DÁVILA, Juan;
RISCO MARTÍN, José Luis;
COLMENAR VERDUGO, José Manuel;
CUESTA INFANTE, Alfredo;
MAQUEDA VILLAIZÁN, Esther;
BOTELLA SERRANO, Marta y
RUBIO GARCÍA, José Antonio**

74 Agente/Representante:

UNGRÍA LÓPEZ, Javier

54 Título: **Método para el modelado del nivel de glucemia mediante programación genética**

57 Resumen:

Método para el modelado del nivel de glucemia mediante programación genética.

La invención consiste en un método que, aplicando algoritmos evolutivos sobre soluciones aleatorias y datos tomados de un paciente con glucemia, permite establecer un modelado del nivel de glucemia para obtener un modelado del nivel de glucosa en instantes futuros a los momentos de obtención de los datos del paciente. Los datos del paciente son al menos los niveles de glucosa, ingesta e insulina rápida y lenta para un intervalo de tiempo. El algoritmo evolutivo consiste básicamente en aplicar programación genética en su variante de gramáticas evolutivas o evolución gramatical. Es decir, aplicar gramáticas personalizadas en formato BNF, procesos de mapeo personalizados y evaluaciones de error concretas a las soluciones aleatorias o a modelos generados con anterioridad para obtener una expresión que describa y prediga los niveles de glucosa de un paciente. De entre todos los pacientes, el método de la presente invención está especialmente indicado para sujetos con Diabetes Mellitus.

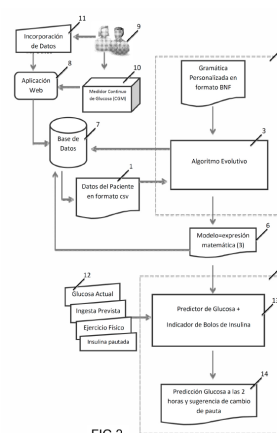


FIG.2

ES 2 540 159 B1

DESCRIPCIÓN

MÉTODO PARA EL MODELADO DEL NIVEL DE GLUCEMIA MEDIANTE PROGRAMACIÓN GENÉTICA

OBJETO DE LA INVENCION

5

La presente invención propone aplicar la programación genética para encontrar un modelo personalizado que describa y prediga los niveles de glucosa de un paciente. De esta forma, la presente invención describe un método que, a partir de los datos históricos de un paciente que consisten en valores previos de glucosa, carbohidratos
10 tomados e insulinas inyectadas, obtiene una expresión que puede usarse para predecir valores de glucosa en un futuro próximo.

El campo de aplicación de la presente invención es la estimación de la glucosa de un paciente a partir de los datos medidos. Debido a la naturaleza de la invención, la
15 estimación de la glucosa de la presente invención permite a dispositivos que operen según el método de la presente invención, predecir los niveles de glucosa de un paciente. De entre todos los pacientes, el método de la presente invención está especialmente indicado para sujetos con *Diabetes Mellitus*.

20

ANTECEDENTES DE LA INVENCION

La Diabetes Mellitus es una enfermedad causada por un defecto en la secreción o en la acción de la insulina, que es esencial para el control de los niveles de glucosa en sangre.

25 En ambos casos el resultado es que las células no asimilan el azúcar y como consecuencia, se produce una subida en los niveles de glucosa en sangre o hiperglucemia. Existen distintos tipos de diabetes dependiendo de su naturaleza. Según la ADA (American Diabetes Association) podemos distinguir cuatro tipos de DM:

- 30
- Diabetes Tipo 1 (T1DM): Las células no producen insulina debido a un proceso autoinmune. Actualmente esto hace necesario que la persona se inyecte insulina exógena, ya sea mediante inyecciones puntuales o que utilice una bomba de insulina.

- Diabetes Tipo 2 (T2DM): El es resultado de una resistencia a la insulina, donde las células no consiguen utilizar la insulina correctamente. En ocasiones se combina con una ausencia, ya sea parcial o total de insulina.
- Diabetes Gestacional: Aparece en los periodos de gestación en una de cada diez mujeres embarazada. El embarazo es un cambio en el metabolismo, ya que el feto utiliza la energía de la madre para obtener alimento, oxígeno y otros recursos. Esto hace que se produzca un descenso en la secreción de insulina por parte de la madre.
- Otros tipos: como por ejemplo problemas en las células , defectos genéticos que afectan a la acción de la insulina, provocados por medicamentos, síndromes genéticos, etc..

En la mayoría de los casos, los pacientes con una evolución larga de la enfermedad requieren la inyección de insulina exógena en varias dosis, o mediante una bomba de insulina. Es importante mantener un buen control glucémico para prevenir tanto las complicaciones agudas de la diabetes (cetoacidosis diabética e hipoglucemia, definida como un valor de glucosa en sangre inferior a 70mg=dl), como todas las complicaciones multi-crónicas asociadas a los pacientes diabéticos (nefropatía, retinopatía, microangiopatía y macroangiopatía).

En los últimos años se ha demostrado que un control estricto del Nivel de glucemia en pacientes críticos mejora su evolución y reduce los costes médicos. El control de los niveles de glucosa es una tarea difícil y que requiere un esfuerzo por ambas partes, pacientes y sus familias. Para mantener buenos niveles de glucosa en sangre el paciente debe tener alguna capacidad de predicción para saber que nivel de glucosa tendría si toma cierta cantidad de comida o si se inyecta cierta cantidad de insulina de un determinado tipo. De hecho el objetivo final es evitar no sólo periodos de hiperglucemia (niveles de glucosa $\geq 180 \text{ mg=dl}$) sino también episodios de Hipoglucemia severa (niveles de glucosa $\leq 40\text{mg=dl}$) que pueden llevar al paciente a la muerte.

Uno de los aspectos que hace difícil el control de glucosa en sangre es la ausencia de un modelo general de respuesta tanto a la insulina como a los diversos factores

mencionados anteriormente, debido principalmente a las particularidades de cada paciente.

5 Los modelos presentes en el estado de la técnica aplican técnicas de modelado clásico que resultan en el uso de ecuaciones lineales, perfiles definidos o modelos con un conjunto limitado de entradas.

10 Otro tipo de técnicas conocidas del estado de la técnica y que nunca han sido utilizadas para realizar modelos de estimación de la glucosa son las técnicas evolutivas. Las técnicas evolutivas como la programación genética (PG) tienen ciertas características que las hacen especialmente útiles para abordar problemas de optimización y modelado complejo. En primer lugar son “simples” conceptualmente hablando y también lo es su aplicación. Sin embargo tienen una base teórica bien definida y ampliamente estudiada. La programación genética ha demostrado su
15 aplicabilidad a multitud de problemas reales y es intrínsecamente paralelizables por trabajar con un conjunto de soluciones. Es más, los algoritmos evolutivos tienen un gran potencial para incorporar conocimiento acerca del dominio en el que trabajan y para incorporar otros mecanismos de búsqueda no necesariamente evolutivos.

20 Una de las aplicaciones más conocidas de la programación genética es la regresión simbólica y la aplicación de una de las variantes de la PG, las gramáticas evolutivas o en inglés Grammatical Evolution (GE) nos permite obtener soluciones para incorporar términos no-lineales. La gramática evolutiva GE es una técnica de computación evolutiva establecida en 1998 por el grupo de Conor Ryan en la Universidad de
25 Limerick (Irlanda). La programación genética trata de encontrar programas ejecutables o funciones que respondan a unos datos de referencia. La principal ventaja es que la gramática evolutiva GE aplica operadores genéticos a una cadena completa lo que simplifica la aplicación de la búsqueda en diferentes lenguajes de programación. Además no tiene problemas de memoria, al contrario que la programación genética
30 básica, donde la representación en árbol puede llevar al conocido problema de bloating (un crecimiento excesivo de las estructuras de datos del computador en la memoria).

Por todo ello, nosotros proponemos aplicar PG para encontrar un modelo personalizado que describa y prediga los niveles de glucosa de un paciente. Nuestro método tomará los datos históricos de un paciente que consisten en valores previos de glucosa carbohidratos tomados e insulinas inyectadas y a partir de ellos obtendrá una
5 expresión que puede usarse para predecir valores de glucosa en un futuro próximo.

Mediante los modelos del estado de la técnica no es posible conocer una estimación de la glucosa en sangre en pacientes ni de forma genérica ni personalizada al paciente. Ello implica que actualmente no es posible aplicar tratamientos adecuados a
10 los pacientes con diabetes. Aunque existen muchos trabajos que usan modelos de control, hasta la fecha el problema del modelado no se ha abordado con técnicas de computación evolutiva. Esta solución no se ha abordado hasta la fecha debido a su complejidad mediante el uso de técnicas clásicas.

15

DESCRIPCIÓN DE LA INVENCIÓN

La presente invención propone una técnica nueva que implica obtener el modelo del paciente particularizado utilizando programación genética "GP" (del inglés "Genetic Programming"). La programación genética "GP" elimina las barreras relacionadas con
20 la construcción del modelo, tales como la linealidad o la limitación en los parámetros de entrada. Mediante la aplicación de la programación genética "GP" en la forma divulgada por la presente invención, es posible encontrar un modelo personalizado que describa y prediga los niveles de glucosa de un paciente. El método de la presente invención toma los datos históricos de un paciente que consisten en valores
25 previos de glucosa carbohidratos tomados e insulinas inyectadas y, a partir de ellos, obtiene una expresión que puede usarse para predecir valores de glucosa en un futuro próximo.

Un primer aspecto de la invención es un método para el modelado del nivel de glucemia que permite predecir la glucosa de un individuo. El método comprende los
30 siguientes pasos:

- i) obtener de un individuo unos datos que comprendan, para un tiempo k , al menos:

- unos niveles de glucosa \widehat{GL} ;
- unos niveles de ingesta CH ;
- unos niveles de insulina inyectada de efecto rápido IS e insulina inyectada de efecto lento IL ;

5 ii) aplicar un Algoritmo Evolutivo a un conjunto de soluciones y a los datos anteriormente obtenidos;

iii) calcular una función \widehat{GL} de predicción de la glucosa para un tiempo posterior $(k+1)$ al tiempo k , tal que:

$$\widehat{GL}(k + 1) = f(\widehat{GL}, CH, IS, IL).$$

10

El paso ii) adicionalmente comprende llevar a cabo los siguientes sub-pasos:

a) generar el conjunto de soluciones con N-soluciones aleatorias donde cada solución está formada por una cadena de caracteres (cromosoma);

15 b) calcular N-expresiones \widehat{GL}_k para $k=1, \dots, N$ obtenidas mediante la decodificación del conjunto de las N-soluciones aleatorias aplicando una gramática BNF y una función de mapeo;

c) calcular el error E_k que conlleva: calcular e_k como la diferencia entre los datos obtenidos del paciente y las N-expresiones \widehat{GL}_k ; y, aplicar una función de fitness a cada uno de los errores anteriormente calculados e_k ; de tal forma que se obtiene para cada expresión \widehat{GL}_k , un error asociado E_k ;

20

d) seleccionar N-1 soluciones resultado de: tomar las N-soluciones y apartar la solución de menor error E_k de las N-soluciones; enfrentar las N-1 soluciones tomadas de dos en dos, seleccionando la solución de menor error E_k ;

e) cruzar las N-1 soluciones anteriores mediante un algoritmo de probabilidad de cruce;

25

f) mutar un carácter de las N-1 soluciones anteriores mediante un algoritmo de probabilidad de mutación;

g) añadir la solución de menor error E_k apartada en el paso d) a las N-1 soluciones anteriores;

30

h) repetir los pasos c) a g) hasta cumplir una condición de parada predefinida.

El algoritmo de probabilidad de cruce comprende:

- i) tomar las N-1 soluciones de dos en dos;

- ii) establecer una probabilidad de cruce entre 0 y 1;
- iii) generar un número aleatorio entre 0 y 1;
- iv) si el número aleatorio generado es mayor que la probabilidad de cruce, no hay cruce;
- 5 v) si el número aleatorio generado es menor o igual que la probabilidad de cruce, se cruza parte de una solución con parte de la otra solución de tal forma que se mantiene la longitud de la cadena de caracteres.

Por otro lado, el algoritmo de probabilidad de mutación comprende:

- 10 i) tomar las N-1 soluciones de una en una;
- ii) establecer una probabilidad de mutación entre 0 y 1;
- iii) generar un número aleatorio entre 0 y 1;
- iv) si el número aleatorio generado es mayor que la probabilidad de mutación, no hay mutación;
- 15 v) si el número aleatorio generado es menor o igual que la probabilidad de mutación, se muta uno o más caracteres de la cadena de caracteres de tal forma que se mantiene la longitud de la cadena de caracteres.

Adicionalmente, la condición de parada predefinida es al menos una de las siguientes condiciones:

- 20 • máximo número de iteraciones;
- convergencia: no mejora en un número "p" de iteraciones;
- estar cerca de un óptimo teórico.

Por otro lado, la función de fitness es una de las siguientes funciones (ver tabla 2):

- 25 • mínimos cuadrados F_1 (*Least Squares*);
- error medio F_2 (*Average Error*);
- error máximo F_3 (*Maximum Error*);
- error cuadrático medio F_4 (*RSME*);
- desviación absoluta media F_5 (*MAD*).

30

Objective	Fitness Function
Least Squares	$F_1 = \sum_{k=1}^N e_k^2$
Average Error	$F_2 = \frac{1}{N} \sum_{k=1}^N e_k$
Maximum Error	$F_3 = \max(e_k), 1 \leq k \leq N$
RSME	$F_4 = \sqrt{\frac{1}{N} \sum_{k=1}^N e_k^2}$
MAD	$F_5 = \frac{1}{N} \sum_{k=1}^N \frac{e_k}{GL(k)}$

Tabla 2.- Funciones de Fitness.

La predicción de la glucosa anteriormente calculada se obtenía a partir de cuatro datos de entrada tomado del paciente. No obstante, el número de datos de entrada puede ser variable hasta un total de veinticinco variables de entrada. Así, la solución $\widehat{GL}(k + 1)$ de partida se ampliaría hasta tener una expresión como la siguiente:

$$\widehat{GL}(k + 1) = f(P, E, k, SI, IG, PG, CI, M, IC, IP, DI, OC, TI, FV, VI, PC, TE, FA, G(\cdot), C(\cdot), IS(\cdot), IL(\cdot), F(\cdot), E(\cdot), Z)$$

donde el operador \cdot representa cualquier instante actual o anterior en el tiempo y donde f es una función que se calcula mediante una gramática BNF que comprende la siguiente forma:

$\langle f \rangle ::= \langle f \rangle \langle OP \rangle \langle f \rangle$
 $| (\langle f \rangle \langle OP \rangle \langle f \rangle)$
 $| \langle PREOP \rangle \langle f \rangle$
 $| \langle V \rangle$
 $| \langle Z \rangle$
 $\langle OP \rangle ::= +$
 $| -$
 $| *$


```

|/
<PREOP> ::= SIN
|COS
|EXP
5 |TAN
|LOG
|ABS
|LN
|TANH
10 |COSH
|SENH
|DERIVADA_CON_RESPECTO_AL_TIEMPO
|DERIVADA_CUADRADO_CON_RESPECTO_AL_TIEMPO
|DERIVADA_TERCERA_CON_RESPECTO_AL_TIEMPO
15 <V> ::= <P>
|<A>
|#{k}
|<SI>
|<IG>
20 |<PG>
|<CI>
|<M>
|<IC>
|<IP>
25 |<DI>
|<OC>
|<TI>
|<FV>
|<VI>
30 |<PC>
|<TE>
|<FA>
|#{G[k-<J>]}

```

- |#{C[k-<J>]}
- |#{IS[k-<J>]}
- |#{IL[k-<J>]}
- |#{F[k-<J>]}
- 5 |#{E[k-<J>]}
- |#<Z>
- <P> ::= “el peso del individuo en kg.”;
- <E> ::= “edad del individuo en años”;
- <SI> ::= “unidades de glucemia que baja 1 unidad de insulina en mg/dl”;
- 10 <IG> ::= “índice glucémico de la ingesta” (el de la glucosa es 1, y puede ser mayor o menor;)
- <PG> ::= <D>.<D>
- <CI> ::= <D>.<D>
- <M> ::= 0
- 15 <IC> ::= 5
- <IP> ::= 0
- <DI> ::= 2
- <OC> ::= 3
- <TI> ::= 12
- 20 <FV> ::= 0.2
- <VI> ::= 0.8
- <PC> ::= 0.6
- <FTSI> ::= +<D>.<D>
- |-<D>.<D>
- 25 <J> ::= 0
- |1
- |2
- |3
- | ...
- 30 | “instantes de tiempo” (por ejemplo: 96 → 24 horas a intervalos de 15 minutos = 96 instantes de tiempo. En el caso de usar otro horizonte de predicción u otro periodo de muestreo habría que modificar esta regla)
- <Z> ::= <D>.<D>

<D> ::= 0

|1

|2

|3

5 |4

|5

|6

|7

|8

10 |9

donde los distintos parámetros de entrada son:

P: peso actual en kg;

A: edad actual en años;

15 k: instante actual;

FSI: sensibilidad a la insulina o mg/dl que baja la glucemia por unidad de insulina;

IG: es el índice glucémico;

PG: unidades proteína/grasa: cantidad de alimento que aporta 100Kcal en forma de
grasa y/o proteínas, se mide en unidades;

20 CI: ratio Hidratos de carbono / insulina;

M: menstruación , Sí =1 ó no=0;

IC: insulina Circulante, unidades de insulina;

IP: intervalo entre comienzo de la infusión de insulina prandial y comienzo de la
ingesta, en intervalos de 15 minutos;

25 DI: tiempo de duración de la última ingesta en intervalos de 15 minutos;

OC: orden en que se consumen los alimentos; (número natural correspondiente a una
fila en una tabla de posibles ordenaciones)

TI: tiempo desde la última ingesta en intervalos de 15 minutos;

FV: factor de variabilidad de absorción, valor entre 0 y 1;

30 VI: variabilidad de la insulina, valor entre 0 y 1;

PC: preparación de la comida (fritos, asados, condimentos..), valor entre 0 y 1;

FTSI: factor transitorio de sensibilidad a la insulina, vendrá determinado por el efecto
de distintos tipos de enfermedades que tiene el paciente y por los fármacos que toma

el paciente, que aumentan o disminuyen la glucosa, y el efecto de la insulina. Lo definimos como un valor que va de -1 a 1;

Z: es una constante;

en cuanto a históricos:

- 5 G(·) es el histórico de la glucemia (o parte de él);
- C(·) es el histórico de carbohidratos;
- IS(·) es la insulina de acción corta;
- IL(·) es la insulina de acción larga;
- F(·) es el ejercicio físico; y,
- 10 E(·) es el nivel de estrés.

Respecto de la función de mapeo indicada anteriormente, ésta comprende la siguiente expresión:

$$Choice_i = (CIV)MOD (\# \text{ of choices}_i)$$

- 15 donde $Choice_i$ es la elección seleccionada para el no-terminal i, CIV es el codón que estamos decodificando, MOD es la función módulo y $(\# \text{ of choices}_i)$ es el número de opciones posibles para la regla en el terminal i.

- 20 Hasta aquí queda descrito el método de la presente invención cuando partimos de cero. Es decir, no existe un modelo anterior de la presente invención aplicado al paciente. En caso de existir un modelo anterior, método de la presente invención, entre los pasos i) y ii), opcionalmente comprendía evaluar si es o no correcta la función \widehat{GL} (=modelo) de predicción de la glucosa.

- 25 En otro aspecto de la presente invención, la presente invención comprende un programa de ordenador para la ejecución de un método para el modelado del nivel de glucemia la predicción de la glucosa según una cualquiera de la realizaciones descritas anteriormente o en el apartado "descripción de una o varias formas de realización de la invención".

30

En otro aspecto de la presente invención, la presente invención comprende un medio de almacenamiento que contiene un programa de ordenador para la ejecución de un

método para el modelado del nivel de glucemia la predicción de la glucosa según una cualquiera de la realizaciones descritas anteriormente o en el apartado “descripción de una o varias formas de realización de la invención”.

El último aspecto de la presente invención está comprendido por un sistema
5 computacional en el que se carga el programa de ordenador anteriormente descrito.

BREVE DESCRIPCIÓN DE LAS FIGURAS

Figura 1.- Representa un diagrama de flujo para calcular el modelo de predicción de la
10 glucosa.

Figura 2.- Representa un diagrama de flujo para calcular el modelo de predicción de la glucosa en una implementación web.

15 Figura 3.- Es el ejemplo de gramática en BNF diseñada para regresión simbólica.

Figura 4.- Es un primer ejemplo de gramática BNF.

Figura 5.- Es un segundo ejemplo de gramática BNF.
20

Figura 6.- Es un tercer ejemplo de gramática BNF.

Figura 7.- Es un cuarto ejemplo de gramática BNF.

DESCRIPCIÓN DE UNA O VARIAS FORMAS DE REALIZACIÓN DE LA 25 INVENCION

Seguidamente se realiza una descripción de una o varias formas de realización de la invención que, de forma no limitativa, ayudan a la mejor comprensión de la presente
30 invención.

La figura 1 muestra un diagrama de flujo del método de acuerdo con la presente invención. Primeramente, se realiza una recogida de datos 1 durante un periodo de

tiempo determinado k , por ejemplo siete días. Los datos recogidos son al menos los niveles de glucosa GL , los niveles de ingesta CH y los niveles de insulina de efecto rápido IS y de efecto lento IL . Posteriormente, se evalúa **2** si el modelo (=función \widehat{GL}) es correcto (si ya existe previamente un modelo). En caso afirmativo, se continúa
 5 recogiendo datos. En caso contrario, se aplica un algoritmo evolutivo **3** consistente, básicamente, en personalizar una gramática en formato BNF **4** mediante la aplicación de una función de mapeo **5**. Como resultado de la aplicación del algoritmo evolutivo **3**, se obtiene un nuevo modelo caracterizado por la función **6**:

$$\widehat{GL}(k + 1) = f(\widehat{GL}, CH, IS, IL)$$

10

que depende de los datos recogidos que son al menos GL , CH , IS e IL .

Los datos recogidos se pueden ampliar con una o varias variables de entrada hasta obtener un modelo caracterizado por una función con la siguiente expresión:

15

$$\widehat{GL}(k + 1) = f(P, E, k, SI, IG, PG, CI, M, IC, IP, DI, OC, TI, FV, VI, PC, TE, FA, G(\cdot), C(\cdot), IS(\cdot), IL(\cdot), F(\cdot), E(\cdot), Z)$$

donde:

P: peso actual en kg;

20

A: edad actual en años;

k: instante actual;

FSI: sensibilidad a la insulina o mg/dl que baja la glucemia por unidad de insulina;

IG: es el índice glucémico u;

PG: unidades proteína/grasa: cantidad de alimento que aporta 100Kcal en forma de

25

grasa y/o proteínas, se mide en unidades;

CI: ratio Hidratos de carbono / insulina;

M: menstruación (días desde la última menstruación), Sí =1 ó no=0;

IC: insulina Circulante, unidades de insulina;

30

IP: intervalo entre comienzo de la infusión de insulina prandial y comienzo de la ingesta, en intervalos de 15 minutos;

DI: tiempo de duración de la última ingesta en intervalos de 15 minutos;

OC: orden en que se consumen los alimentos; (número natural correspondiente a una

fila en una tabla de posibles ordenaciones)

TI: tiempo desde la última ingesta en intervalos de 15 minutos;

FV: factor de variabilidad de absorción, valor entre 0 y 1;

VI: variabilidad de la insulina, valor entre 0 y 1;

5 PC: preparación de la comida (fritos, asados, condimentos..), valor entre 0 y 1;

FTSI: factor transitorio de sensibilidad a la insulina, vendrá determinado por el efecto de distintos tipos de enfermedades que tiene el paciente y por los fármacos que toma el paciente, que aumentan o disminuyen la glucosa, y el efecto de la insulina. Lo definimos como un valor que va de -1 a 1;

10 Z: es una constante;

en cuanto a históricos:

G(·) es el histórico de la glucemia (o parte de él);

C(·) es el histórico de carbohidratos;

IS(·) es la insulina de acción corta;

15 IL(·) es la insulina de acción larga;

F(·) es el ejercicio físico; y,

E(·) es el nivel de estrés.

La figura **2** muestra un diagrama de flujo donde el método de la presente invención se encuentra implementado dentro de un sistema computacional que comprende una base de datos **7** y una aplicación web **8**, a través de la cual, un paciente **9** puede introducir **11** en la base de datos **7** los datos correspondientes a los niveles de glucosa medidos por un medidor continuo de glucosa **10**. El sistema computacional adicionalmente comprende unos medios computacionales **15** tales como memorias, microprocesadores, unidades de entrada/salida. Los medios computacionales **15** están encargados de almacenar las gramáticas y procesar el algoritmo evolutivo para obtener el modelo o expresión matemática para $GL(k+1)$ **6** con el que obtener la predicción de la glucosa **13**. Los modelos computacionales **15** pueden ampliarse con medios computacionales opcionales **15'** que servirían para predecir la glucosa **14** e indicar los bolos de insulina en un tiempo futuro (por ejemplo 2 horas) combinación la función $GL(k+1)$ con los datos "instantáneos" de glucosa, ingesta prevista, ejercicio físico e insulina pautada **12**.

A continuación se detalla cómo se aplica el algoritmo evolutivo para modelar el nivel de glucemia en individuos mediante Programación Genética, y especialmente en sujetos con *Diabetes Mellitus*.

5 1. Aproximación Evolutiva

El objetivo de la presente invención es encontrar una expresión que modele el nivel de glucosa de un paciente diabético. Esta expresión debe ser obtenida, al menos, a partir de datos previos de glucosa, carbohidratos e insulina almacenados en un sistema o
10 base de datos. Por lo tanto, estamos ante un problema similar al problema de la regresión simbólica. La regresión simbólica trata de obtener expresiones matemáticas que reproduzcan un conjunto discreto de datos.

La Programación Genética (PG) ha demostrado ser efectiva en un alto número de
15 problemas de regresión simbólica, aunque tiene algunas limitaciones, que a menudo vienen del modo de representación como por ejemplo el “bloating”. Otro punto que hay que considerar es que en Programación Genética (PG), la evolución se produce en el fenotipo del individuo y no en su representación (genotipo). Durante los últimos años, variantes de la programación genética, las gramáticas evolutivas han aparecido para
20 proponer diferentes aproximaciones a la evolución. La Gramática Evolutiva (GE) permite la generación de programas de computador en un lenguaje arbitrario. Esto se consigue mediante la utilización de gramáticas para especificar las reglas de obtención de programas. Específicamente, la presente invención utiliza gramáticas expresadas en la forma de BNF (Backus Naur form).

25 En contraste con los algoritmos genéticos, que trabajan con representación de soluciones, la Gramática Evolutiva GE trabaja (evoluciona) con un código genético que determina el proceso de producción de la solución. El proceso de traducción de código se determina por las Gramáticas representadas como BNF.

30 BNF es una notación técnica para expresar gramáticas libres de contexto. Una representación en BNF puede ser cualquier especificación de un lenguaje completo o

un subconjunto de un lenguaje orientado al problema. Una especificación BNF es un conjunto de reglas de derivación, expresadas en la forma:

$$\langle \text{símbolo} \rangle ::= \langle \text{expresión} \rangle$$

5

Las reglas están compuestas de secuencias de terminales y no terminales. Los símbolos que aparecen a la izquierda son no-terminales, mientras que los terminales nunca aparecen al lado izquierdo. En este caso, se puede afirmar que $\langle \text{símbolo} \rangle$ es no-terminal y, aunque no es una especificación BNF completa, también se puede afirmar para la presente invención que $\langle \text{expresión} \rangle$ será un no-terminal ya que siempre aparecen entre el par $\langle \rangle$. Por tanto, en este caso, el no-terminal $\langle \text{símbolo} \rangle$ será remplazado por una expresión. La operación de remplazo se representa con “ $::=$ ”. El resto de la gramática debe indicar las diferentes posibilidades.

15 Una gramática está representada por una 4-tupla $\{N, T, P, S\}$, siendo N el conjunto de no-terminales, T es el conjunto de terminales, P son las reglas de producción para la asignación de elementos de N a T, y S es un símbolo de comienzo que debería aparecer en N. Las opciones dentro de una regla de producción están separadas por el símbolo “|”.

20

La Figura 3 representa un ejemplo de gramática en BNF diseñado para regresión simbólica. El código que representa una expresión consistirá de elementos de un conjunto de terminales T. Estos han sido combinados con las reglas de la gramática como se explica más adelante.

25

Además las gramáticas pueden ser adaptadas para sesgar la búsqueda del proceso evolutivo porque hay un número finito de opciones de reglas de producción, la cual limita el espacio de búsqueda.

30 2. Proceso de Mapeo.

Como se ha mencionado anteriormente, la presente invención aplica un algoritmo evolutivo para evolucionar genotipos, los cuales se representan como una cadena de

valores enteros. Cada genotipo mapea el símbolo de comienzo a símbolos terminales leyendo los codones, los cuales, en este ejemplo de realización, tienen una longitud de 8 bits. El proceso es similar al explicado en la sección previa pero, en vez de hacer elecciones aleatorias, tomaremos las decisiones leyendo el genotipo. Cada codón es por tanto un valor entero del genotipo, el cual es procesado mediante la siguiente función de mapeo:

$$Choice_i = (CIV) \text{MOD } (\# \text{ of choices}_i)$$

donde $Choice_i$ es la elección seleccionada para el no-terminal i , CIV es el codón que se está decodificando, MOD es la función módulo y $(\# \text{ of choices}_i)$ es el número de opciones posibles para la regla en el terminal i .

10

La función de mapeo toma el valor entero del cromosoma, calcula el módulo respecto al número de posibilidades de la regla y selecciona la opción de acuerdo con el resultado. Dado que la función módulo devuelve valores entre 0 y $(\# \text{ of choices}_i) - 1$, la primera opción corresponderá al primer valor, 0, la segunda a 1, y así sucesivamente.

15

Por tanto, si una regla tiene sólo una posibilidad, ésta siempre será seleccionada.

En el presente ejemplo de realización, el proceso de mapeado usa la gramática de la Figura 3, diseñada para resolver un problema de regresión simbólica y los problemas del modelo de la glucosa, a falta de particularizar el conjunto terminal, lo cual se describe en la siguiente sección. Un individuo se compone de una serie de genes (valores enteros). Cada gen puede tomar un valor entre 0 y 255 ya que los codones son de 8 bits. Supongamos por ejemplo el siguiente individuo de 7 genes:

20

$$12 - 55 - 23 - 47 - 38 - 254 - 2$$

25

El símbolo comienzo es $S = \{ \text{expr} \}$, por tanto la expresión de la solución comenzará con este no-terminal:

$$\text{Solución} = \langle \text{expr} \rangle$$

30

Para obtener el fenotipo aplicamos la función de mapeo al primer gen ($CIV = 12$) usando la regla del primer no-terminal de la expresión. En este punto sólo aparece un

no-terminal, <expr>, el cual se corresponde con la regla I de la Figura 3. El número de opciones en esa regla es tres. Por tanto, la función mapeo aplicada es:

$$12 \text{ MOD } 3 = 0$$

5

Por lo que se selecciona la primera opción, <expr> <op> <expr>, y se continúa con el proceso de mapeo. La opción seleccionada sustituye los no-terminales decodificados. Como resultado la expresión es, en este punto,:

10

$$\text{Solución} = \langle \text{expr} \rangle \langle \text{op} \rangle \langle \text{expr} \rangle$$

El proceso continuará con el siguiente gen, 55, el cual se emplea en decodificar el primer no-terminal de la expresión en su estado de desarrollo actual. A continuación se aplica de nuevo el mapeo, ahora a la regla I:

15

$$55 \text{ MOD } 3 = 1$$

La segunda opción, <pre_op> (<expr>), es entonces seleccionada. La expresión pasa a ser:

20

$$\text{Solución} = \langle \text{pre op} \rangle (\langle \text{expr} \rangle) \langle \text{op} \rangle \langle \text{expr} \rangle$$

El siguiente gen, 23, pasa a ser descodificado. En este punto del proceso, el primer no-terminal que aparece en la expresión es <pre_op>. Por tanto, se puede aplicar la función de mapeo a la regla III, la cual tiene tres posibilidades:

25

$$23 \text{ MOD } 3 = 2$$

Al tomar el valor 2 se selecciona la tercera opción, el símbolo terminal *Abs*.

La expresión pasa a ser:

30

$$\text{Solución} = \text{Abs}(\langle \text{expr} \rangle) \langle \text{op} \rangle \langle \text{expr} \rangle$$

El siguiente gen, 47, descodifica <expr> con la regla I:

$$47 \text{ MOD } 3 = 2$$

Por lo que se selecciona la tercera opción, <var>. La expresión pasa a ser:

5

$$\text{Solución} = \text{Abs}(\langle \text{var} \rangle) \langle \text{op} \rangle \langle \text{expr} \rangle$$

El gen 38 descodifica <var> con la regla IV:

10

$$38 \text{ MOD } 2 = 0$$

Dicho valor selecciona la primera opción, el no-terminal X.

$$\text{Solution} = \text{Abs}(X) \langle \text{op} \rangle \langle \text{expr} \rangle$$

15 El no-terminal <op> se descodifica con 254 y la regla II:

$$254 \text{ MOD } 4 = 2$$

Este valor selecciona la tercera opción, el terminal *.

20

$$\text{Solución} = \text{Abs}(X) * \langle \text{expr} \rangle$$

El siguiente gen, 2, descodifica <expr> con la regla I:

25

$$2 \text{ MOD } 3 = 2$$

Este valor selecciona la tercera opción, el no-terminal <var>.

$$\text{Solución} = \text{Abs}(X) * \langle \text{var} \rangle$$

30

En este punto el proceso de transformación de genotipo a fenotipo ha agotado los codones. Es decir, se han utilizado todos los genes o codones pero no se ha llegado a una expresión con terminales en todos sus componentes.

La solución es reusar los codones comenzando por el primero otra vez. Esto representa una novedad respecto de los algoritmos evolutivos distintos de las Gramáticas Evolutivas, del estado de la técnica. De hecho es posible reutilizar los
 5 codones varias veces. Esta técnica se conoce como “wrapping” y simula el fenómeno de solapamiento de genes que se da en muchos organismos. Reusar codones no es un problema ya que en Gramática Evolutiva (GE), un codón siempre genera el mismo valor entero y, si se aplica a la misma regla, genera la misma solución. Sin embargo, si los usamos con reglas diferentes obtendremos diferentes partes de fenotipos. Así, las
 10 Gramáticas Evolutivas (GE) aseguran que un genotipo individual siempre produce el mismo fenotipo. Por tanto, el “wrapping” no supone un problema. De esta manera, aplicando el “wrapping”, el proceso vuelve atrás, al primer gen, 12, el cual se utiliza para decodificar <var> con la regla IV:

15
$$12 \text{ MOD } 2 = 0$$

Este valor selecciona la primera opción, el no-terminal X, dando lugar a la expresión definitiva del fenotipo:

20
$$\text{Solución} = \text{Abs}(X) * X$$

En la siguiente sección se describen cuatro gramáticas correspondientes a cuatro ejemplos de realización para representar diferentes espacios de búsqueda para obtener expresiones que modelen los niveles de glucosa en sangre.

25 3. Descripción del modelo.

Para que un modelo sea completo, dicho modelo, para los niveles de glucosa, debería basarse en factores observables así como en otras características ocultas e intrínsecas al organismo del paciente. Los factores observables son aquellos datos
 30 que ha recogido el paciente manualmente o un dispositivo automático, mientras que los no observables deben ser inferidos. Por estos motivos, la presente invención propone un modelo que considera todos estos factores, aplicando Gramáticas

Evolutivas (GE) para inferir una expresión que caracterice el comportamiento de la glucosa en pacientes diabéticos.

3.1. Datos disponibles y modelo general de la glucosa.

5

El nivel de la glucosa en un instante dado depende de varios factores, algunos de ellos intrínsecos al funcionamiento del organismo. En el caso de un paciente de diabetes, los más relevantes son el nivel de glucosa que había hasta la última medida, los carbohidratos ingeridos y la insulina inyectada. Estos factores están incluidos en los
10 conjuntos de datos de nuestros pacientes. Es importante destacar que estos datos son fáciles de recopilar para un paciente real. El valor de la glucosa se mide con analizadores de sangre, los carbohidratos se miden en unidades ingeridas a partir de las comidas diarias, y la cantidad de insulina así como el tipo, son datos que el paciente obviamente conoce.

15

Para el primer ejemplo de realización, se han obtenido datos con un periodo de medición entre dos consecutivos de 15 minutos a lo largo de 24 horas. La Tabla 1 muestra el conjunto de datos obtenido para un paciente "X1". En cada fila de la tabla k se representa el tiempo, donde $k = 1$ son las 00:00AM, GL es el nivel de glucosa en
20 dicho instante, CH las unidades de carbohidratos ingeridas, IS la insulina de efecto rápido inyectada e IL la de efecto lento.

k	GL	CH	IS	IL	k	GL	CH	IS	IL	k	GL	CH	IS	IL
1	209.1453	0	0	0	34	174.09018	0	0	0	66	272.82541	0	0	0
2	209.1453	0	0	0	35	176.68898	0	0	0	67	271.4209	0	0	0
3	205.79354	0	0	0	36	184.17896	0	0	0	68	272.60383	0	0	0
4	202.56395	0	0	0	37	195.61325	0	0	0	69	271.58261	0	0	0
5	199.43946	0	0	0	38	209.11478	0	0	0	70	268.30335	0	0	0
6	196.41567	0	0	0	39	223.51534	0	0	0	71	263.63006	0	4	0
7	193.49677	0	0	0	40	237.54628	0	0	0	72	258.18738	0	0	0
8	190.69246	0	0	0	41	247.25104	20	0	0	73	252.2917	30	0	0
9	188.01547	0	0	0	42	250.72465	0	0	0	74	246.13977	0	0	0
10	185.47971	0	0	0	43	251.90543	0	0	0	75	240.98451	0	0	0
11	183.09885	0	0	0	44	255.86031	0	0	0	76	241.19843	0	0	0
12	180.88533	0	0	0	45	262.91532	0	0	0	77	246.19998	0	0	0
13	178.84963	0	0	0	46	271.63911	0	0	0	78	254.31449	0	0	0
14	176.9999	0	0	0	47	277.89942	0	0	0	79	264.42406	0	0	0
15	175.34167	0	0	0	48	278.70943	0	0	0	80	275.21267	0	0	0
16	173.87786	0	0	0	49	275.38173	40	0	0	81	282.61864	0	0	0
17	172.60884	0	0	0	50	269.9759	0	0	0	82	284.42698	0	0	0
18	171.53263	0	0	0	51	264.80613	0	0	0	83	282.33176	0	0	0
19	170.6451	0	0	0	52	264.5835	0	0	0	84	277.80323	0	0	0
20	169.94031	0	0	0	53	269.03811	0	0	0	85	271.88307	0	0	0
21	169.41076	0	0	0	54	276.56921	0	0	0	86	265.29768	0	0	0
22	169.04771	0	0	0	55	286.01609	0	0	0	87	258.54258	0	0	0
23	168.84148	0	0	0	56	296.54932	0	0	0	88	251.94544	0	0	0
24	168.78173	0	0	0	57	307.58483	0	0	0	89	245.71299	0	0	0
25	168.85766	0	0	0	58	317.78442	0	0	0	90	239.91877	0	0	0
26	169.05827	0	0	0	59	322.9634	0	0	0	91	234.68683	0	0	18
27	169.37254	0	0	0	60	322.05895	0	0	0	92	230.03136	0	0	0
28	169.78957	0	0	0	61	317.01482	0	0	0	93	225.93797	0	0	0
29	170.29872	0	0	0	62	309.34854	0	0	0	94	222.23379	0	0	0
30	170.88974	0	0	0	63	300.16725	0	0	0	95	218.78458	0	0	0
31	171.55425	0	0	0	64	290.26909	10	0	0	96	215.4919	0	0	0
32	172.27976	0	3	12	65	280.21928	0	0	0	97	212.29133	0	0	0
33	173.05923	30	0	0										

Tabla 1.- Ejemplo de datos tomados para un paciente “X1” para los valores *GL*, *CH*, *IS* *IL*.

- 5 El modelo propuesto nos proporciona los valores estimados de la glucosa, representados por \widehat{GL} . En cada iteración, la glucosa estimada se obtiene usando los valores estimados anteriores, y los valores de los carbohidratos e insulina en dicho instante, formalmente:

10
$$\widehat{GL}(k + 1) = f(\widehat{GL}; CH; IS; IL); \quad 1 \leq k \leq N \quad (\text{ec.1})$$

donde $\widehat{GL}(k + 1)$ es el futuro valor estimado para la glucosa, un instante posterior al actual, y el resto de variables tiene la misma interpretación que la anteriormente descrita. De esta manera el motor de la Gramática Evolutiva (GE) debería ser capaz de decidir el aspecto de *f*. Para guiar la búsqueda necesitamos una gramática que limite el espacio de búsqueda y capture la dinámica de los niveles de glucosa en sangre. A continuación detallamos las gramáticas empleadas en la presente invención.

3.2. Gramáticas BNF para el modelado de los niveles de glucosa.

Siguiendo el modelo general mostrado en (ec.1), se han diseñado cuatro gramáticas donde la glucosa estimada depende únicamente de factores observables. La incorporación de conocimiento del problema en la gramática mejora el rendimiento de la exploración. Así, las gramáticas diseñadas siguen los consejos de los médicos en el equipo de investigación. Según sus indicaciones, el comportamiento esperado de la glucosa puede variar a lo largo del día, por lo que se han añadido las posibles influencias en forma de ventanas de tiempo. En definitiva, hay cuatro gramáticas que mezclan las diferentes observaciones.

10

Gramática 1

Es bien conocido que la ingesta de carbohidratos eleva la glucosa mientras que las inyecciones la disminuyen. Por este motivo en la primera gramática los carbohidratos siempre suman mientras que la insulina siempre resta. Además permite utilizar cualquier valor anterior de la glucosa, los carbohidratos o la insulina al construir la expresión.

15

$$\widehat{GL}(k + 1) = f_{gl}(\widehat{GL}(k-m)) + f_{ch}(CH(k-m)) - f_{in}(IS(k-m); IL(k-m)); 0 \leq m \leq k; \quad (\text{gr.1})$$

20

La forma concreta de f_{gl} , f_{ch} y f_{in} se obtiene mediante GE con la Gramática 1, mostrada en la Figura 4. Los tres términos <exprgluc>, <exprch> y <exprins> se corresponden con f_{gl} , f_{ch} y f_{in} , respectivamente y son las expresiones que pueden usar prefijos (operandos) como aquellos de la regla IX, variables para cada uno de los términos o combinaciones de ellos a través de la regla VIII.

25

Gramática 2

Esta gramática es un caso particular de la anterior que limita el histórico de valores de glucosa, carbohidratos e insulina al instante actual k y el anterior $k-1$. El modelo resultante tiene la expresión general:

30

$$\widehat{GL}(k + 1) = f_{gl}(\widehat{GL}(k-m)) + f_{ch}(CH(k-m)) - f_{in}(IS(k-m); IL(k-m)); \quad 0 \leq m \leq 1; \quad (\text{gr.2})$$

La Figura 5 muestra la gramática empleada. Simplemente hay que limitar a 00 y 01 los índices en las reglas III, V y VII.

5 Gramática 3

Para dotar de más libertad a la búsqueda, se dejan libres las operaciones conectivas, ampliándolas a las operaciones aritméticas básicas. La expresión general del modelo es (gr.3), donde f es la función que conecta las tres subexpresiones para la glucosa, carbohidratos e insulina.

$$\widehat{GL}(k + 1) = f(f_{gl}(\widehat{GL}(k - m)); f_{ch}(CH(k - m)); f_{in}(IS(k - m); IL(k - m))); 0 \leq m \leq k; \text{ (gr.3)}$$

La gramática 3 es la encargada de definir este modelo, el cual presenta una ligera modificación en la regla I de la gramática 1. Se trata de cambiar las operaciones fijas + y - por el no-terminal <op> y a continuación se define en la regla VIII. La figura 6 muestra la gramática 3.

20 Gramática 4

El modelo general es similar a la gramática 2 pero añade libertad a función f que relaciona las subexpresiones de la glucosa, los carbohidratos y la insulina.

$$\widehat{GL}(k + 1) = f(f_{gl}(\widehat{GL}(k - m)); f_{ch}(CH(k - m)); f_{in}(IS(k - m); IL(k . m))); 0 \leq m \leq 1; \text{ (gr. 4)}$$

25 Para ello la gramática 4 es similar a la gramática 2 pero dotando de libertad a las operaciones en la regla I. La gramática se muestra en la Figura 7.

30 3.3. Evaluación fitness o aptitud (error e_k)

Si las gramáticas limitan el espacio de búsqueda de la Gramática Evolutiva GE, la misión de las funciones de "fitness" (también conocidas como *aptitud*) es guiar la evolución hacia una buena solución. Para calcular el "fitness", primero se obtiene la

serie temporal de la glucosa completa, \widehat{GL} , a partir del fenotipo generado por el genotipo del individuo y la gramática empleada. En este paso la glucosa estimada en k se realimenta para estimar la glucosa en los siguientes k .

- 5 A continuación se mide el error e_k , definido como la diferencia absoluta entre el valor real, en el conjunto de entrenamiento, y el valor estimado para cada k .

$$e_k = |GL(k) - \widehat{GL}(k)|, \quad 1 \leq k \leq N$$

- 10 Finalmente se emplean las cinco funciones de fitness diferentes, basadas en el error e_k , recogidas en la Tabla 2. Las funciones de fitness contenidas en la Tabla 2 son: F_1 (mínimos cuadrados), F_2 (error medio), F_3 (error máximo), F_4 (error cuadrático medio) y F_5 (desviación absoluta media).

Objective	Fitness Function
Least Squares	$F_1 = \sum_{k=1}^N e_k^2$
Average Error	$F_2 = \frac{1}{N} \sum_{k=1}^N e_k$
Maximum Error	$F_3 = \max(e_k), 1 \leq k \leq N$
RSME	$F_4 = \sqrt{\frac{1}{N} \sum_{k=1}^N e_k^2}$
MAD	$F_5 = \frac{1}{N} \sum_{k=1}^N \frac{e_k}{GL(k)}$

15

Tabla 2.- Ejemplos de funciones de fitness.

REIVINDICACIONES

1.- Método para el modelado del nivel de glucemia mediante programación genética que comprende los siguientes pasos:

- 5 i) obtener de un individuo unos datos que comprendan, para un tiempo k , al menos:
- o unos niveles de glucosa \widehat{GL} ;
 - o unos niveles de ingesta CH ;
 - o unos niveles de insulina inyectada de efecto rápido IS e insulina
- 10 inyectada de efecto lento IL ;
- ii) aplicar un Algoritmo Evolutivo a un conjunto de soluciones y a los datos anteriormente obtenidos;
- iii) calcular una función \widehat{GL} de predicción de la glucosa para un tiempo posterior $(k+1)$ al tiempo k , tal que:

$$\widehat{GL}(k + 1) = f(\widehat{GL}, CH, IS, IL)$$

15 caracterizado por que el paso ii) adicionalmente comprende llevar a cabo los siguientes pasos:

- a) generar el conjunto de soluciones con N-soluciones aleatorias donde cada solución está formada por una cadena de caracteres (cromosoma);
 - b) calcular N-expresiones \widehat{GL}_k para $k=1, \dots, N$ obtenidas mediante la
- 20 decodificación del conjunto de las N-soluciones aleatorias aplicando una gramática BNF y una función de mapeo;
- c) calcular el error E_k que conlleva: calcular e_k como la diferencia entre los datos obtenidos del paciente y las N-expresiones \widehat{GL}_k ; y, aplicar una función de fitness a cada uno de los errores anteriormente calculados e_k ; de tal forma que
- 25 se obtiene para cada expresión \widehat{GL}_k , un error asociado E_k ;
- d) seleccionar N-1 soluciones resultado de: tomar las N-soluciones y apartar la solución de menor error E_k de las N-soluciones; enfrentar las N-1 soluciones tomadas de dos en dos, seleccionando la solución de menor error E_k ;
 - e) cruzar las N-1 soluciones anteriores mediante un algoritmo de probabilidad de
- 30 cruce;
- f) mutar un carácter de las N-1 soluciones anteriores mediante un algoritmo de probabilidad de mutación;

- g) añadir la solución de menor error E_k apartada en el paso d) a las N-1 soluciones anteriores;
- h) repetir los pasos c) a g) hasta cumplir una condición de parada predefinida.

5 2.- Método según la reivindicación 1, caracterizado por que el algoritmo de probabilidad de cruce comprende:

- i) tomar las N-1 soluciones de dos en dos;
- ii) establecer una probabilidad de cruce entre 0 y 1;
- iii) generar un número aleatorio entre 0 y 1;
- 10 iv) si el número aleatorio generado es mayor que la probabilidad de cruce, no hay cruce;
- v) si el número aleatorio generado es menor o igual que la probabilidad de cruce, se cruza parte de una solución con parte de la otra solución de tal forma que se mantiene la longitud de la cadena de caracteres.

15

3.- Método según la reivindicación 1, caracterizado por que el algoritmo de probabilidad de mutación comprende:

- i) tomar las N-1 soluciones de una en una;
- ii) establecer una probabilidad de mutación entre 0 y 1;
- 20 iii) generar un número aleatorio entre 0 y 1;
- iv) si el número aleatorio generado es mayor que la probabilidad de mutación, no hay mutación;
- v) si el número aleatorio generado es menor o igual que la probabilidad de mutación, se muta un carácter de la cadena de caracteres de tal forma que se
- 25 mantiene la longitud de la cadena de caracteres.

4.- Método según la reivindicación 1, caracterizado por que la condición de parada predefinida es al menos una de las siguientes condiciones:

- máximo número de iteraciones;
- 30 • convergencia: no mejora en un número "p" de iteraciones;
- estar cerca de un óptimo teórico.

5.- Método según la reivindicación 1, caracterizado por que la función de fitness es una de las siguientes funciones:

- mínimos cuadrados;
- error medio;
- 5 • error máximo;
- error cuadrático medio;
- desviación absoluta media.

6.- Método según la reivindicación 1, caracterizado por que la solución

10 $\widehat{GL}(k + 1)$ tiene la siguiente expresión:

$$\widehat{GL}(k + 1) = f(P, E, k, SI, IG, PG, CI, M, IC, IP, DI, OC, TI, FV, VI, PC, TE, FA, G(\cdot), C(\cdot), IS(\cdot), IL(\cdot), F(\cdot), E(\cdot), Z)$$

donde el operador \cdot representa cualquier instante actual o anterior en el tiempo y

15 donde f es una función que se calcula mediante una gramática BNF que comprende la siguiente forma:

<f> ::= <f><OP><f>
 | (<f><OP><f>)
 20 | <PREOP>(<f>)
 | <V>
 | <Z>
 <OP> ::= +
 |-
 25 |*
 |/
 <PREOP> ::= SIN
 |COS
 |EXP
 30 |TAN
 |LOG
 |ABS
 |LN

|TANH
 |COSH
 |SENH
 |DERIVADA_CON_RESPECTO_AL_TIEMPO
 5 |DERIVADA_CUADRADO_CON_RESPECTO_AL_TIEMPO
 |DERIVADA_TERCERA_CON_RESPECTO_AL_TIEMPO
 <V> ::= <P>
 |<A>
 |#{k}
 10 |<SI>
 |<IG>
 |<PG>
 |<CI>
 |<M>
 15 |<IC>
 |<IP>
 |<DI>
 |<OC>
 |<TI>
 20 |<FV>
 |<VI>
 |<PC>
 |<TE>
 |<FA>
 25 |#{G[k-<J>]}
 |#{C[k-<J>]}
 |#{IS[k-<J>]}
 |#{IL[k-<J>]}
 |#{F[k-<J>]}
 30 |#{E[k-<J>]}
 |#{Z>
 <P> ::= “el peso del individuo en kg.”;
 <E> ::= “edad del individuo en años”;

<SI> ::= “unidades de glucemia que baja 1 unidad de insulina en mg/dl”;

<IG> ::= “índice glucémico de la ingesta” (el de la glucosa es 1, y puede ser mayor o menor;)

<PG> ::= <D>.<D>

5 <CI> ::= <D>.<D>

<M> ::= 0

<IC> ::= 5

<IP> ::= 0

<DI> ::= 2

10 <OC> ::= 3

<TI> ::= 12

<FV> ::= 0.2

<VI> ::= 0.8

<PC> ::= 0.6

15 <FTSI> ::= +<D>.<D>

|-<D>.<D>

<J> ::= 0

|1

|2

20 |3

|...

| “instantes de tiempo” (por ejemplo: 96 → 24 horas a intervalos de 15 minutos = 96 instantes de tiempo. En el caso de usar otro horizonte de predicción u otro periodo de muestreo habría que modificar esta regla);

25 <Z> ::= <D>.<D>

<D> ::= 0

|1

|2

|3

30 |4

|5

|6

|7

|8

|9

donde los distintos parámetros de entrada son:

- 5 P: peso actual en kg;
A: edad actual en años;
k: instante actual;
FSI: sensibilidad a la insulina o mg/dl que baja la glucemia por unidad de insulina;
IG: es el índice glucémico;
- 10 PG: unidades proteína/grasa: cantidad de alimento que aporta 100Kcal en forma de
grasa y/o proteínas, se mide en unidades;
CI: ratio Hidratos de carbono / insulina;
M: menstruación , Sí =1 ó no=0;
IC: insulina Circulante, unidades de insulina;
- 15 IP: intervalo entre comienzo de la infusión de insulina prandial y comienzo de la
ingesta, en intervalos de 15 minutos;
DI: tiempo de duración de la última ingesta en intervalos de 15 minutos;
OC: orden en que se consumen los alimentos; (número natural correspondiente a una
fila en una tabla de posibles ordenaciones)
- 20 TI: tiempo desde la última ingesta en intervalos de 15 minutos;
FV: factor de variabilidad de absorción, valor entre 0 y 1;
VI: variabilidad de la insulina, valor entre 0 y 1;
PC: preparación de la comida (fritos, asados, condimentos..), valor entre 0 y 1;
FTSI: factor transitorio de sensibilidad a la insulina: valor de -1 a 1;
- 25 Z: es una constante;
en cuanto a históricos:
G(·) es el histórico de la glucemia (o parte de él);
C(·) es el histórico de carbohidratos;
IS(·) es la insulina de acción corta;
- 30 IL(·) es la insulina de acción larga;
F(·) es el ejercicio físico; y,
E(·) es el nivel de estrés.

7.- Método según la reivindicación 1, caracterizado por que la función de mapeo comprende la siguiente expresión:

$$Choice_i = (CIV)MOD (\# \text{ of choices}_i)$$

donde $Choice_i$ es la elección seleccionada para el no-terminal i , CIV es el codón que estamos decodificando, MOD es la función módulo y $(\# \text{ of choices}_i)$ es el número de opciones posibles para la regla en el terminal i .

5

8.- Método según la reivindicación 1, caracterizado por que el método, entre los pasos i) y ii), opcionalmente comprende evaluar la función \widehat{GL} de predicción de la glucosa anteriormente calculada cuando existe un modelo de glucosa anteriormente calculado.

10

9.- Programa de ordenador caracterizado por que comprende la ejecución de un método definido de acuerdo con una cualquiera de las reivindicaciones 1 a 8.

10.- Medio de almacenamiento caracterizado por que comprende un programa de ordenador definido según la reivindicación 9.

15

11.- Sistema computacional caracterizado por que en dicho sistema computacional se carga el programa de ordenador definido en la reivindicación 9.

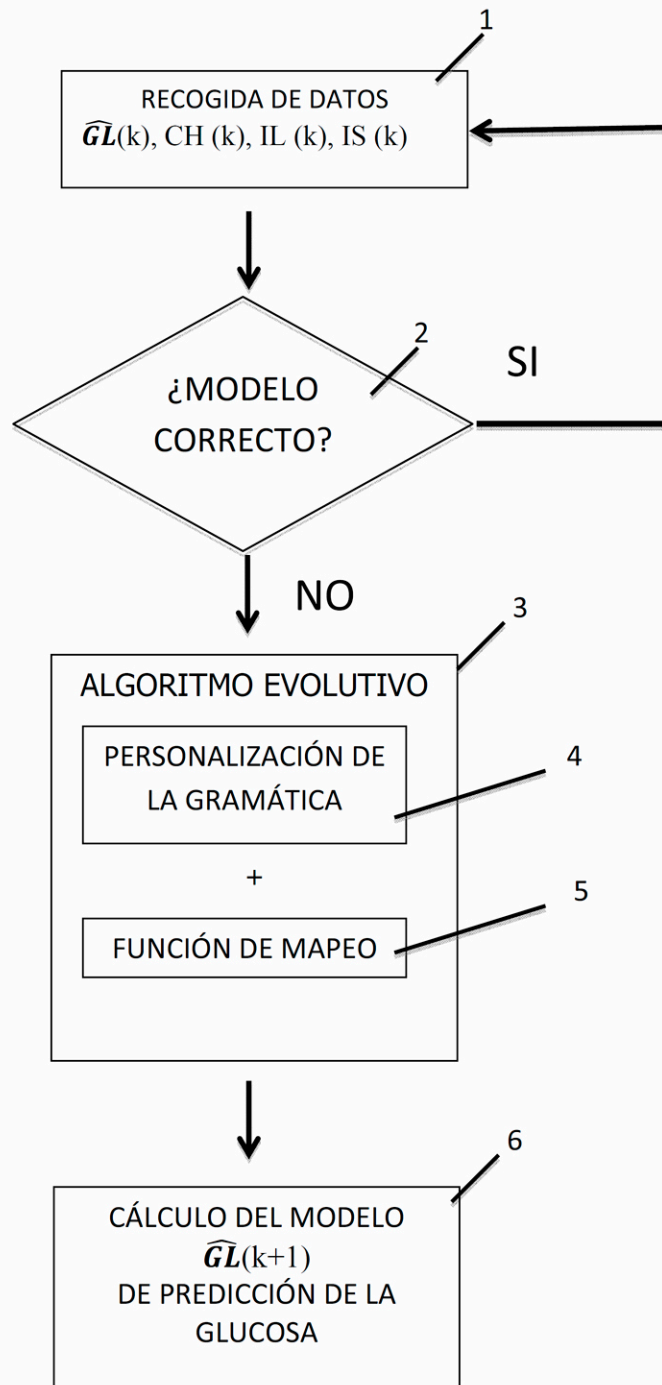


FIG. 1

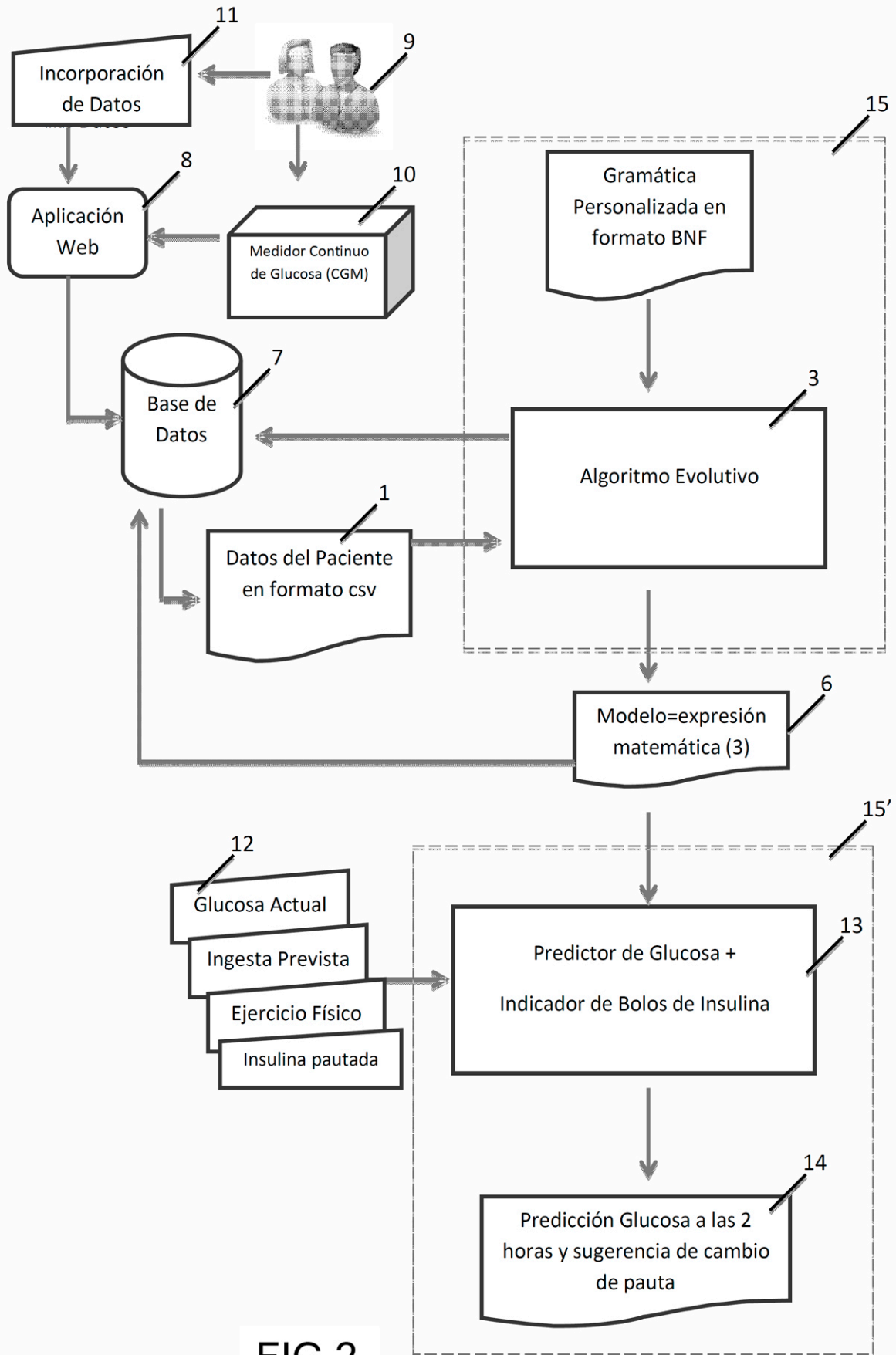


FIG.2

```

N = { expr, op, pre_op, var, num, dig }
T = { +, -, *, /, Sin, Cos, Abs, X, 0, 1, 2, 3, 4, 5, (, ), . }
S = { expr }
P = { I, II, III, IV, V, VI }

I   <expr>      ::= <expr> <op> <expr>
                       | <pre_op> (<expr>)
                       | <var>

II  <op>        ::= + | - | * | /

III <pre_op>    ::= Sin | Cos | Abs

IV  <var>       ::= X | <num>

V   <num>       ::= <dig>.<dig> | <dig>

VI  <dig>       ::= 0 | 1 | 2 | 3 | 4 | 5

```

FIG. 3

```

N = {func, exprgluc, gluc, exprch, varch, exprins, varins, op, preop, idx,
     cte, dgt}
T = { +, -, *, /, sin, cos, tan, exp, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, GL, CH,
     IS, IL, K}
S = {func}
P = {I, II, III, IV, V, VI, VII, VIII, IX, X, XI, XII}

I   <func>      ::= <exprgluc> + <exprch> - <exprins>

II  <exprgluc> ::= <preop> (<gluc>)
                       | (<cte> <op> <gluc>)
                       | <gluc>

III <gluc>      ::= #{GL[k_<idx>]}|#{K}

IV  <exprch>    ::= <exprch> <op> <exprch>
                       | <preop> (<exprch>)
                       | <varch>

V   <varch>     ::= #{CH[k_<idx>]}|#{K}|<cte>

VI  <exprins>   ::= <exprins> <op> <exprins>
                       | <preop> (<exprins>)
                       | <varins>

VII <varins>    ::= #{IS[k_<idx>]}|#{IL[k_<idx>]}|#{K}|<cte>

VIII <op>       ::= +|-|/|*
IX  <preop>     ::= sin|cos|tan|exp
X   <idx>       ::= <dgt><dgt>
XI  <cte>       ::= <dgt><dgt>.<dgt><dgt>
XII <dgt>       ::= 0|1|2|3|4|5|6|7|8|9

```

FIG. 4

```

N = {func, exprgluc, gluc, exprch, varch, exprins, varins, op, preop, cte,
dgt}
T = { +, -, *, /, sin, cos, tan, exp, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, GL, CH,
IS, IL, K}
S = {func}
P = {I, II, III, IV, V, VI, VII, VIII, IX, X, XI}

I   <func> ::= <exprgluc> + <exprch> - <exprins>

II  <exprgluc> ::= <preop> (<gluc>)
      |(<cte> <op> <gluc>)
      |<gluc>

III <gluc> ::= #{GL[k_00]}|#{GL[k_01]}|#{K}

IV  <exprch> ::= <exprch> <op> <exprch>
      |<preop> (<exprch>)
      |<varch>

V   <varch> ::= #{CH[k_00]}|#{CH[k_01]}|#{K}|<cte>

VI  <exprins> ::= <exprins> <op> <exprins>
      |<preop> (<exprins>)
      |<varins>

VII <varins> ::= #{IS[k_00]}|#{IS[k_01]}|#{IL[k_00]}|#{IL[k_01]}|#{K}|<cte>

VIII <op> ::= +|-|/|*
IX   <preop> ::= sin|cos|tan|exp
X    <cte> ::= <dgt><dgt>.<dgt><dgt>
XI   <dgt> ::= 0|1|2|3|4|5|6|7|8|9

```

FIG. 5

```

N = {func, exprgluc, gluc, exprch, varch, exprins, varins, op, preop, cte,
     dgt}
T = { +, -, *, /, sin, cos, tan, exp, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, GL, CH,
     IS, IL, K}
S = {func}
P = {I, II, III, IV, V, VI, VII, VIII, IX, X, XI}

I   <func> ::= <exprgluc> <op> <exprch> <op> <exprins>

II  <exprgluc> ::= <preop> (<gluc>)
      |(<cte> <op> <gluc>)
      |<gluc>

III <gluc> ::= #{GL[k_00]}|#{GL[k_01]}|#{K}

IV  <exprch> ::= <exprch> <op> <exprch>
      |<preop> (<exprch>)
      |<varch>

V   <varch> ::= #{CH[k_00]}|#{CH[k_01]}|#{K}|<cte>

VI  <exprins> ::= <exprins> <op> <exprins>
      |<preop> (<exprins>)
      |<varins>

VII <varins> ::= #{IS[k_00]}|#{IS[k_01]}|#{IL[k_00]}|#{IL[k_01]}|#{K}|<cte>

VIII <op> ::= +|-|/|*
IX   <preop> ::= sin|cos|tan|exp
X    <cte> ::= <dgt><dgt>.<dgt><dgt>
XI   <dgt> ::= 0|1|2|3|4|5|6|7|8|9

```

FIG. 6

```

N = {func, exprgluc, gluc, exprch, varch, exprins, varins, op, preop, cte,
dgt}
T = { +, -, *, /, sin, cos, tan, exp, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, GL, CH,
IS, IL, K}
S = {func}
P = {I, II, III, IV, V, VI, VII, VIII, IX, X, XI}

I   <exprgluc> ::= <exprgluc> <op> <exprch> <op> <exprins>

II  <exprgluc> ::= <preop> (<gluc>)
      |(<cte> <op> <gluc>)
      |<gluc>

III <gluc> ::= #{GL[k_00]}|#{GL[k_01]}|#{K}

IV  <exprch> ::= <exprch> <op> <exprch>
      |<preop> (<exprch>)
      |<varch>

V   <varch> ::= #{CH[k_00]}|#{CH[k_01]}|#{K}|<cte>

VI  <exprins> ::= <exprins> <op> <exprins>
      |<preop> (<exprins>)
      |<varins>

VII <varins> ::= #{IS[k_00]}|#{IS[k_01]}|#{IL[k_00]}|#{IL[k_01]}|#{K}|<cte>

VIII <op> ::= +|-|/|*
IX   <preop> ::= sin|cos|tan|exp
X    <cte> ::= <dgt><dgt>.<dgt><dgt>
XI   <dgt> ::= 0|1|2|3|4|5|6|7|8|9

```

FIG. 7



②① N.º solicitud: 201331726

②② Fecha de presentación de la solicitud: 27.11.2013

③② Fecha de prioridad:

INFORME SOBRE EL ESTADO DE LA TECNICA

⑤① Int. Cl.: **G06F19/00** (2011.01)

DOCUMENTOS RELEVANTES

Categoría	⑤⑥ Documentos citados	Reivindicaciones afectadas
X	WO 2010019919 A1 (UNIV TOLEDO et al.) 18.02.2010, párrafos [0008-0102].	1,9-12
A		2-8
X	US 2011077930 A1 (ALFERNESS CLIFTON A et al.) 31.03.2011, párrafos [0013-0163].	1
X	US 2008154513 A1 (KOVATCHEV BORIS P et al.) 26.06.2008, párrafos [0013-0141].	1
A	WO 2005113036 A1 (UNIV CALIFORNIA et al.) 01.12.2005, párrafos [0051-0227].	1-12

Categoría de los documentos citados

X: de particular relevancia

Y: de particular relevancia combinado con otro/s de la misma categoría

A: refleja el estado de la técnica

O: referido a divulgación no escrita

P: publicado entre la fecha de prioridad y la de presentación de la solicitud

E: documento anterior, pero publicado después de la fecha de presentación de la solicitud

El presente informe ha sido realizado

para todas las reivindicaciones

para las reivindicaciones n.º:

Fecha de realización del informe
20.10.2014

Examinador
A. Casado Fernández

Página
1/4

Documentación mínima buscada (sistema de clasificación seguido de los símbolos de clasificación)

G06F

Bases de datos electrónicas consultadas durante la búsqueda (nombre de la base de datos y, si es posible, términos de búsqueda utilizados)

INVENES, EPODOC

Fecha de Realización de la Opinión Escrita: 20.10.2014

Declaración

Novedad (Art. 6.1 LP 11/1986)	Reivindicaciones 2-8	SI
	Reivindicaciones 1, 9-12	NO
Actividad inventiva (Art. 8.1 LP11/1986)	Reivindicaciones 2-8	SI
	Reivindicaciones 1, 9-12	NO

Se considera que la solicitud cumple con el requisito de aplicación industrial. Este requisito fue evaluado durante la fase de examen formal y técnico de la solicitud (Artículo 31.2 Ley 11/1986).

Base de la Opinión.-

La presente opinión se ha realizado sobre la base de la solicitud de patente tal y como se publica.

1. Documentos considerados.-

A continuación se relacionan los documentos pertenecientes al estado de la técnica tomados en consideración para la realización de esta opinión.

Documento	Número Publicación o Identificación	Fecha Publicación
D01	WO 2010019919 A1 (UNIV TOLEDO et al.)	18.02.2010
D02	US 2011077930 A1 (ALFERNES CLIFTON A et al.)	31.03.2011
D03	US 2008154513 A1 (KOVATCHEV BORIS P et al.)	26.06.2008
D04	WO 2005113036 A1 (UNIV CALIFORNIA et al.)	01.12.2005

2. Declaración motivada según los artículos 29.6 y 29.7 del Reglamento de ejecución de la Ley 11/1986, de 20 de marzo, de Patentes sobre la novedad y la actividad inventiva; citas y explicaciones en apoyo de esta declaración

El documento D01 se considera el más próximo del estado de la técnica a la invención solicitada. Entre paréntesis se cita las partes relevantes del documento D01.

Reivindicación 1:

Método para el modelado del nivel de glucemia mediante programación genética que comprende los siguientes pasos ([0039]-[0084]):

i) obtener de un individuo unos datos que comprendan, para un tiempo k, al menos ([0008], [0090]):

unos niveles de glucosa GL;

unos niveles de ingesta CH;

unos niveles de insulina inyectada de efecto rápido IS e insulina inyectada de efecto lento IL;

ii) aplicar un Algoritmo Evolutivo a un conjunto de soluciones y a los datos anteriormente obtenidos ([0145], [0147]);

iii) calcular una función GL de predicción de la glucosa para un tiempo posterior (k+1) al tiempo k ([0079], [0102]), tal que:

$GL(k + 1) = f(GL, CH, IS, IL)$

A la vista de lo indicado se concluye que la reivindicación 1 no es nueva (Artículo 6 LP.).

Reivindicación 2:

La diferencia principal entre el documento D01 y la reivindicación 2 es que D01 no menciona un algoritmo evolutivo que genere el conjunto de soluciones con N-soluciones aleatorias donde cada solución está formada por una cadena de caracteres, que calcule N-expresiones obtenidas mediante la decodificación del conjunto de las N-soluciones aleatorias aplicando una gramática BNF y una función de mapeo, que calcule el error E_k , que seleccionar N-1 soluciones resultado de: tomar las N-soluciones y apartar la solución de menor error E_k de las N-soluciones; enfrentar las N-1 soluciones tomadas de dos en dos, seleccionando la solución de menor error E_k ; y que cruce las N-1 soluciones anteriores mediante un algoritmo de probabilidad de cruce;

El efecto técnico de esta diferencia es aplicar un algoritmo evolutivo a los datos obtenidos del individuo para modelar el nivel de glucemia.

El problema técnico es cómo modelar el nivel de glucemia de un individuo a partir de un algoritmo evolutivo.

D01 no implementa un algoritmo equivalente para modelar el nivel de glucemia de un individuo.

A la vista de lo mencionado anteriormente se concluye que la reivindicación 2 es nueva e implica actividad inventiva (Artículos 6 y 8 LP.).

Reivindicaciones 3-8:

Las reivindicaciones 3 a 8, dependientes de la reivindicación 2, son por tanto nuevas e implican actividad inventiva (Artículos 6 y 8 LP.).

Reivindicación 9:

La diferencia entre D01 y la reivindicación 9 es que en la reivindicación 9 se incluye opcionalmente evaluar la función GL de predicción de la glucosa anteriormente calculada cuando existe un modelo de glucosa anteriormente calculado.

Esta diferencia es una mera variante constructiva que se considera dentro del alcance de la práctica habitual seguida por el experto en la materia.

Consecuentemente, el objeto de la reivindicación 9 carece de actividad inventiva (Artículo 8 LP.).

Reivindicaciones 10-12:

Las reivindicaciones 10 a 12 son reivindicaciones de tipo declarativo. No se consideran nuevas en la medida en que la reivindicación 1 no implica novedad.