Sistema de Inteligencia de Negocio Entorno a las Enfermedades Cognitivas

Rafael Flores Hernández
Máster en Ingeniería Informática
Business Intelligence

David Amorós Alcaraz
María Isabel Guitart Hormigo

12/06/17

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Sistema de inteligencia de negocio entorno a las enfermedades cognitivas*
Nombre del autor:	*Rafael Flores Hernández*
Nombre del consultor/a:	*David Amorós Alcaraz*
Nombre del PRA:	*María Isabel Guitart Hormigo*
Fecha de entrega (mm/aaaa):	06/2017
Titulación::	*Máster en Ingeniería Informática*
Área del Trabajo Final:	*Business Intelligence*
Idioma del trabajo:	*Español*
Palabras clave	*BI, cognitive, diseases*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.*

El objetivo del presente trabajo es conseguir mediante la aplicación de técnicas de inteligencia de negocio conclusiones claras sobre los efectos que tienen determinadas circunstancias en la evolución de las enfermedades mentales. Disponemos de un fichero fuente en formato Excel que contiene toda la información a analizar. Se implementará un modelo de datos en base a técnicas de diseño dimensional donde albergaremos la información obtenida del Excel de forma estructurada siguiendo los patrones de dimensiones y hechos. Para la consecución de todas las tareas se hará uso de la suite Community de Pentaho. De esta forma los procesos de ETL serán elaborados mediante la herramienta KETTLE, el diseño del cubo OLAP lo realizaremos con el programa Schema Workbench y el análisis de los datos finales lo llevaremos a cabo mediante el servidor de análisis de PENTAHO. Una vez analizados los datos, como conclusiones generales podemos sacar que aquellas actividades que menos esfuerzo físico o mental requieren están más relacionadas con los episodios graves que sufre un paciente, independientemente del tipo de enfermedad. También podemos comprobar como aquellos pacientes que viven en un entorno rural han sufrido menos episodios graves a lo largo del año. Mediante el estudio de la evolución de cada paciente podemos ver las distintas evoluciones que ha sufrido cada uno. También sacamos conclusiones a nivel de fechas, pudiendo establecer que agosto es el mes con más episodios graves.

Abstract (in English, 250 words or less):

The scope of this work is to obtain clear answers about the connection between some circumstances and the evolution of cognitive disorders through business intelligence's technics. Our main data source is an Excel file that has all the information to analyze. We will implement a data model based on dimensional modeling procedures where data from the Excel file will be placed following dimensions and facts patterns. In order to achieve all the tasks, Pentaho Community Suite offers a vast set of applications that covers all of our necesities. This way, the ETL processes will be elaborated by KETTLE, the OLAP cube design with Schema Workbench software and the final data analysis through Business Analytics Platform. Once we have ended all the analysis process, as a general conclusion we can stablish that the activities that need less physical and mental effort are linked to severe episodes that patients suffer, no matter what disorder it is. Furthermore we can realize how patients that live in rural environments had suffered less severe episodes during the year. By the study of each patient's evolution we can see their different progresses. Finally analyzing the data according to the dates information, august is the month with more severe episodes.

Índice

Table of Contents

Lista de figuras

No se encuentran elementos de tabla de ilustraciones.

1. Introducción

1.1 Contexto y justificación del Trabajo

El trabajo nos sitúa en el área de los trastornos cognitivos, donde existen diversas afecciones que pueden llegar a suponer la incapacidad de la persona. Investigar estas enfermedades, a parte de ampliar los conocimientos, puede suponer una mejora en la calidad de vida de los pacientes si estas son tratadas de forma correcta.

Hoy en día con la capacidad de cálculo que ofrecen los sistemas de Business Intelligence, si disponemos de una fuente de datos lo suficientemente extensa podremos analizar la información en busca de patrones de comportamiento, o detectar aquellas medidas que son beneficiosas o no para un paciente.

Tenemos que tener en cuenta que cada vez más, la población alcanza edades que hace un siglo eran impensables, por lo que el estudio de estas enfermedades que afectan en gran medida a personas de avanzada edad se torna de vital importancia.

A día de hoy no existen medios que puedan dar una respuesta efectiva a este tipo de trastornos, por lo que el análisis de los datos proporcionados podría arrojar conclusiones de gran ayuda en el tratamiento de estas enfermedades.

Como resultado de este trabajo se espera obtener una serie de informes que puedan evidenciar las relaciones existentes entre el avance de la enfermedad y los agentes externos a los que los pacientes se ven expuestos en el día a día, tales como la actividad diaria u horas de sueño.

1.2 Objetivos del Trabajo

El objetivo principal del trabajo es dar respuesta a 8 preguntas formuladas en el enunciado.

- ¿Cuál es la relación entre las actividades realizadas y los episodios de crisis graves?
- ¿Se puede establecer algún tipo de relación entre los valores de los diferentes estados de ánimo y los episodios de crisis?
- Estas relaciones son igualas para cualquiera de las enfermedades o en cambio hay relaciones más acusadas por alguna de ellas.
- ¿Se puede establecer alguna relación en nivel geográfico, por ejemplo entorno urbano o rural?
- ¿Cuál sido la evolución de los diferentes pacientes a lo largo del tiempo?

- ¿Se puede establecer alguna relación entre los episodios de crisis y el momento del día o de la semana o del año?
- La realización de actividades físicas mejora o empeora el estado de ánimo de los pacientes.
- ¿Hay algún tipo de actividad que mejore el día a día de los pacientes?

Para poder conseguir esta finalidad, se construirá un sistema de BI donde almacenaremos toda la información obtenida en un DWH y mediante una plataforma de consulta obtener un informe que responda a estas cuestiones.

1.3 Enfoque y método seguido

Dentro de todas las posibilidades que se presentan a la hora de emprender un proyecto de BI podríamos destacar tanto el uso o elección de una herramienta dedicada, así como el planteamiento inicial de diseño.
Tal y como hemos podido aprender en la lectura del libro de Ralph Kimball "The Data warehouse Lifecycle Toolkit", el diseño del modelo que almacenará la información debe realizarse siguiendo un **enfoque dimensional** (*Dimensional Model*), en vez de aplicar la clásica **3ª forma normal** [1].
Este punto es clave, ya que este tipo de modelado proporciona una estabilidad al sistema, tanto a la hora de soportar futuros cambios y ampliaciones al modelo, como ofrecer velocidades de consulta mucho mayores. Más adelante, en los capítulos dedicados de esta memoria, entraremos más en detalle sobre las decisiones a tomar durante el modelado y el por qué de cada una de ellas.
Otro enfoque importante a tomar en este proyecto, es el de la decisión de la herramienta a utilizar. En un proyecto de BI podemos optar por desarrollar una herramienta adhoc que pueda proveernos de aquellas funcionalidades que necesitemos para cada una de las fases del trabajo, pero hay que destacar que el área en el que nos encontramos existen soluciones asentadas y muy aceptadas por la comunidad de BI. Por lo tanto no es necesario reinventar la rueda y en vez de afrontar el desarrollo de estas aplicaciones desde cero, tarea que además podría superar el alcance del trabajo solicitado, la opción más sensata es decantarse por alguna plataforma ya desarrollada que nos proporcione de las herramientas necesarias para la consecución de cada una de las fases del proyecto.

1.4 Planificación del Trabajo

Descripción del problema

Se nos plantea el análisis de información relevante a un conjunto de pacientes clínicos.
El área en el que se enfoca es el de las enfermedades cognitivas.

Mediante la implementación de un sistema de Business Intelligence (BI a partir de ahora) se pretende realizar el estudio e interpretación de los datos para poder dar respuesta a las siguientes preguntas:

- ¿Cuál es la relación entre las actividades realizadas y los episodios de crisis graves?
- ¿Se puede establecer algún tipo de relación entre los valores de los diferentes estados de ánimo y los episodios de crisis?
- Estas relaciones son iguales para cualquiera de las enfermedades o en cambio hay relaciones más acusadas por alguna de ellas.
- ¿Se puede establecer alguna relación en nivel geográfico, por ejemplo entorno urbano o rural?
- ¿Cuál ha sido la evolución de los diferentes pacientes a lo largo del tiempo?
- ¿Se puede establecer alguna relación entre los episodios de crisis y el momento del día o de la semana o del año?
- La realización de actividades físicas mejora o empeora el estado de ánimo de los pacientes.
- ¿Hay algún tipo de actividad que mejore el día a día de los pacientes?

Descripción del trabajo

A alto nivel se pretende desarrollar un entorno de BI que nos permita sacar conclusiones a partir del análisis de los datos clínicos provenientes de una fuente de origen.
Estas conclusiones nos ayudarán a dar respuesta a las preguntas planteadas en el apartado anterior.

Objetivos

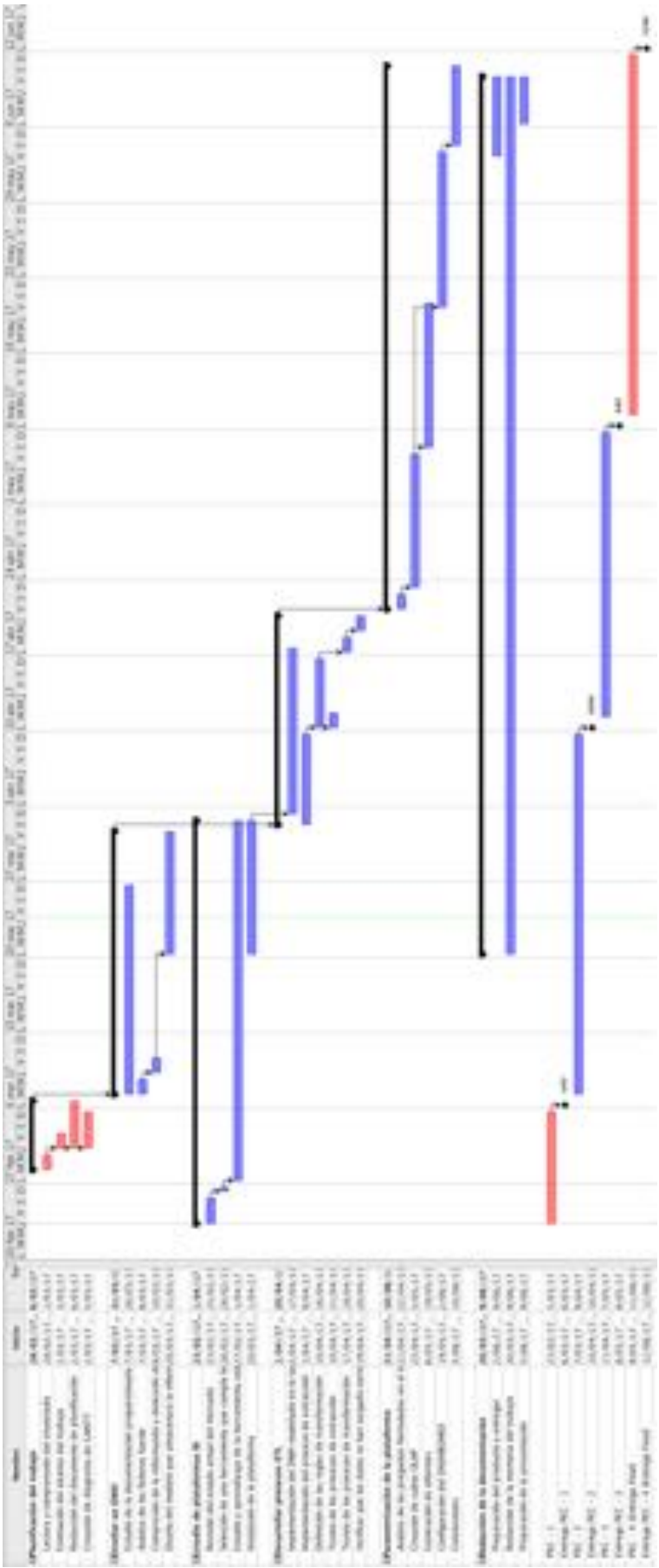
Tal y como hemos comentado, tenemos que desarrollar un entorno de BI que nos ayude a la obtención, carga y análisis de los datos proporcionados.

Como proyecto de BI podemos destacar los siguientes objetivos clave:

1. **Diseñar un almacén de datos** (Data Warehouse, DWH a partir de ahora), que recopile toda la información obtenida desde las distintas fuentes.
2. Implementar el DWH y **desarrollar los procesos ETL** (extracción, transformación y carga), que serán los encargados de introducir en el DWH los datos.
3. Realizar un **estudio de las diferentes plataformas BI** existentes en el mercado y que nos permitan realizar la explotación de la información almacenada. Finalmente se debe elegir una y configurarla.
4. **Parametrización de la plataforma** elegida para que se adapte a las exigencias del proyecto en cuanto a la generación de informes, configuración del Dashboard, cubos OLAP etc…

Objetivo	Tareas	Fecha Inicio	Fecha Fin
Planificación del trabajo	Lectura y comprensión del enunciado	28/02/17	01/03/17
	Estimación del alcance del trabajo	02/03/17	03/03/17
	Redacción del documento de planificación	02/03/17	06/03/17
	Creación de diagrama de GANTT	02/03/17	05/03/17
Diseñar un DWH	Estudio de la documentación proporcionada	07/03/17	26/03/17
	Análisis de los ficheros fuente (donde vendrán los datos iniciales)	07/03/17	08/03/17
	Compresión de la información y detección de posibles campos sujetos a una posible normalización	09/03/17	10/03/17
	Diseño del modelo que almacenará la información	20/03/17	31/03/17
Desarrollar	Implementación del DWH modelado	02/04/17	17/04/17

procesos ETL	en la tarea anterior		
	Implementación del proceso de extracción	01/04/17	09/04/17
	Definición de las reglas de transformación	10/04/17	16/04/17
	Testeo de los procesos de extracción	10/04/17	11/04/17
	Testeo de los procesos de transformación	17/04/17	18/04/17
	Verificar que los datos se han cargado correctamente en el DWH	19/04/17	20/04/17
Estudio de plataformas BI	Revisión del estado actual del mercado	23/02/17	25/02/17
	Selección de una herramienta que cumpla los requisitos del proyecto	26/02/17	26/02/17
	Estudio y aprendizaje de la herramienta seleccionada	27/02/17	01/04/17
	Instalación de la plataforma	20/03/17	01/04/17
Parametrización de la plataforma	Análisis de las preguntas formuladas en el trabajo	21/04/17	22/04/17
	Creación de cubos OLAP	23/04/17	05/05/17
	Generación de informes	06/05/17	19/05/17
	Configuración del DASHBOARD	19/05/17	02/06/17
	Conclusiones	03/06/17	10/06/17
Redacción de la documentación	Preparación del producto a entregar	02/06/17	09/06/17
	Redacción de la memoria del trabajo	20/03/17	09/06/17
	Preparación de la presentación	05/06/17	09/06/17



Teniendo en cuenta las fechas de entrega establecidas para cada “PEC”, pasamos a detallar los elementos del trabajo que serán entregados en cada uno de estos hitos:

Objetivo	Tareas	Fecha de Entrega
PEC 1	Planificación del trabajo	06/03/17
	Revisión actual del mercado	
	Selección de una herramienta que cumpla los requisitos del proyecto	
PEC 2	Correcciones de la entrega anterior	10/04/17
	Análisis de los ficheros proporcionados	
	Diseño del DWH	
	Primera versión de la memoria	
PEC 3	Correcciones de la entrega anterior	08/05/17
	Desarrollar los procesos ETL	
	Análisis de las preguntas formuladas en el trabajo	
	Creación de cubos OLAP	
	Segunda versión de la memoria	
PEC 4	Correcciones de la entrega anterior	12/06/17
	Generación de los informes	
	Configuración del DASHBOARD	
	Conclusiones	
	Preparación del producto a entregar	
	Preparación de la presentación	
	Entrega de la memoria final	

1.5 Breve sumario de productos obtenidos

Como resultado del trabajo, hemos creado un servidor dentro de una máquina virtual donde albergamos tanto la BBDD (nuestro DWH) como la plataforma de análisis BI de Pentaho.
A parte tenemos un paquete de scripts tanto de creación como de backup del contenido de las BBDD.
En el área relativa a los procesos de ETL, tenemos los distintos archivos que hemos generado mediante la herramienta Kettle para ejecutar los distintos procesos de extracción, transformación y carga.
Mediante la herramienta Schema Workbench hemos generado el CUBO OLAP, que se compone de un archivo XML que contiene los metadatos necesarios para definir la instancia en nuestro servidor de análisis.
Finalmente, dentro de la presente memoria se pueden consultar los resultados obtenidos que dan respuesta a las preguntas formuladas en el apartado 2.7.

1.6 Breve descripción de los otros capítulos de la memoria

Selección de una herramienta de BI

En este apartado trataremos muy por encima las razones por las cuales nos hemos decantado por la suite Communnity de Pentaho para realizar las tareas del proyecto.

DWH: Diseño del modelo

Comprende todos los pasos y medidas tomadas para diseñar el modelo de datos en el que se basará nuestro Data Warehouse. Es la base sobre la cual construiremos nuestro sistema de BI

Diseño físico del Datawarehouse

Explicación a alto nivel de los pasos realizados para levantar físicamente el diseño de nuestro modelo.

ETL: Desarrollos de procesos ETL (Extract Transform Load)

Realizaremos un repaso a alto nivel de los procesos (trabajos y transformaciones) que entran en juego en la fase de ETL y que han sido desarrollados mediante la herramienta KETTLE. Se mencionará también el anexo correspondiente donde se explica con mayor nivel de detalle aquellas etapas que se han considerado de interés.

Diseño del cubo OLAP

Mostraremos los elementos principales de la herramienta Schema Workbench de la suite Pentaho y como estos deben configurarse para adaptarse a nuestro modelo dimensional.

Instalación de una plataforma de consulta

Asentaremos las nociones básicas de la plataforma Business Analytics de Pentaho para poder operar con cubos. Mostraremos también dos posibles herramientas de análisis.

Informes y respuestas a las preguntas formuladas

Finalmente, en este apartado justificaremos mediante la herramienta de análisis cada una de las respuestas a las preguntas planteadas objeto del presente proyecto.

2. Resto de capítulos

2.1. Selección de una herramienta de BI

La opción seleccionada para la consecución de este trabajo será la suite Community de Pentaho.

Después de haber realizado una pequeña labor de investigación sobre las distintas aplicaciones, dimos con estos interesantes diagramas de Gartner [2] [3]:

Podemos observar como Pentaho pasa de estar en el cuadrante de “NICHE PLAYERS” al de “VISIONARIES”.

También destacamos la cantidad de información didáctica que se puede encontrar en la red, lo cual supone una gran ayuda para comenzar en un proyecto de BI.
Finalmente, el conjunto de herramientas que compone la suite PENTAHO nos proporciona las utilidades necesarias para poder completar el proyecto de inicio a fin [4].

Podemos destacar los siguientes componentes:

- **Business Analytics Platform**: Lo cual compone una aplicación que puede arrancarse a modo de servidor y donde podremos realizar entre otras tareas, la explotación de datos o la revisión de informes.
- **Data Integration**: O también conocida como Spoon o Kettle, es la aplicación estrella de Pentaho. Se centra en las tareas de ETL.
- **Report Designer**: Como su propio nombre indica, nos servirá para el diseño de los informes

- **Schema Workbench**: Mediante esta aplicación de la suite podremos, entre otras cosas, diseñar los cubos OLAP que luego podremos explotar mediante el **Business Analytics Platform**.

Enlace a la web de Pentaho: http://community.pentaho.com/

A parte, hemos investigado:

- RapidMiner: https://rapidminer.com/
- BIRT: http://www.eclipse.org/birt/
- Jasper Reports: http://community.jaspersoft.com/
- SpagoBI: http://www.spagoworld.org/

Los tres últimos (BIRT, Jasper y SpagoBI) son herramientas basadas en Eclipse.

A modo de resumen exponemos la siguiente tabla comparativa entre estos productos, donde podemos visualizar claramente por qué Pentaho es la opción más recomendada para afrontar este proyecto.

	PENTAHO	RapidMiner	BIRT	Jasper Reports	SpagoBI
Versión gratuita	✔	✖	✔	✖	✔
Herramienta ETL	✔	✔	✖	✔	✖
Diseño de informes	✔	✖	✔	✔	✔
Herramienta para la creación de esquemas	✔	✖	✖	✔	✖
Plataforma de explotación de datos	✔	✔	✖	✔	✔

Hemos expuesto estos 5 indicadores ya que serán claves en el desempeño del proyecto y aunque podríamos haber mezclado productos, como por ejemplo, usar el diseño de informes de SpagoBI y por otro lado la herramienta de ETL de Pentaho (Kettle), creemos que homogeneizar la plataforma puede redundar en una mayor simpleza del resultado final.

Por todo ello y volviendo a recalcar que la suite “community” de Pentaho nos ofrece todo lo necesario para la consecución de las distintas fases del trabajo, concluimos que esta es la alternativa que mejor se adapta a nuestras necesidades.

2.2. DWH: Diseño del modelo

Tal y como hemos podido aprender de la lectura del libro "*The Data Warehouse Lifecycle Toolkit*" los elementos clave de esta etapa del trabajo son [1]:

- **Bus Matrix**: Una representación de las distintas dimensiones de consulta orientadas al negocio y las áreas en las que intervienen.
- **Modelado dimensional a alto nivel**: Donde reflejaremos gráficamente las dimensiones que se han detectado así como la tabla de hechos.
- **Diseño detallado de las tablas de dimensiones y de hechos**: Se representará la información tanto gráficamente como un resumen de las características de cada tabla y sus componentes.

Bus Matrix

Para una correcta representación del bus, debemos centrar nuestra atención en las preguntas formuladas en el enunciado, y una a una podremos ir detectando así las dimensiones y su relación.

Q1:"¿Cuál es la relación entre las actividades realizadas y los episodios de crisis graves?"

Para dar respuesta a esta pregunta necesitaremos realizar una consulta donde intervengan tanto los datos de las actividades (ACTIVITY VALUES) así como el del nivel de los episodios sufridos (EPISODE VALUES). De este análisis extraemos dos dimensiones a tener en cuenta: **actividad** y **episodio**.

Q2:"¿Se puede establecer algún tipo de relación entre los valores de los diferentes estados de ánimo y los episodios de crisis? "

Es imposible poder dar una respuesta a esta pregunta mediante los datos fuente proporcionados, pues carecemos de la información relativa a los estados de ánimo de un paciente.

Q3:"Estas relaciones son iguales para cualquiera de las enfermedades o en cambio hay relaciones más acusadas por alguna de ellas."

Debido a que la segunda pregunta planteada no se puede responder, enfocamos esta tercera cuestión a la primera relación, actividad/episodio. En este caso, debido a que necesitamos comparar los resultados con los tipos de enfermedad, deducimos la necesidad de una nueva dimensión **enfermedad** que nos permita realizar este cruce de información. Esta dimensión la podremos construir a partir de los datos contenidos en la pestaña PATIENTS del fichero fuente, donde encontramos la columna COGNITIVE DISORDER.

Q4:"¿Se puede establecer alguna relación en nivel geográfico, por ejemplo entorno urbano o rural?"

Efectivamente, debido a que poseemos el dato del tipo de entorno donde vive cada paciente, podremos realizar consultas mediante esa variable combinándola con el resto de información disponible.
Por ello, detectamos en esta pregunta una nueva dimensión: **entorno**. Al igual que la dimensión enfermedad, esta nueva tabla la generaremos a partir de los datos contenidos en la pestaña PATIENTS del fichero fuente, donde encontramos la columna ENVIRONMENT.

Incluso podríamos crear del mismo modo una dimensión más relacionada con la ciudad, ya que disponemos de ese dato (columna CITY).

Q5:"¿Cuál sido la evolución de los diferentes pacientes a lo largo del tiempo? "

Curiosamente, aun siendo la dimensión más usada con diferencia, no ha sido hasta la quinta pregunta donde nos hemos encontrado con la necesidad de realizar consultas mediante el atributo de tipo fecha: **fecha**. Igualmente, teniendo en cuenta que debemos realizar las comparaciones por paciente, la creación de una dimensión que pueda extraer información relacionada por individuo es necesaria, por ello nos encontramos con la dimensión **paciente**.

Q6:"¿Se puede establecer alguna relación entre los episodios de crisis y el momento del día o de la semana o del año?"

La granularidad del dato que se nos ofrece como fuente de información es a nivel de día, por lo que es imposible obtener respuestas relacionadas con el momento del día. Sin embargo, sí podríamos procesar la dimensión **fecha**, para que esta guarde tanto el día de la semana (Lunes, Martes, Miércoles...) como el mes o la estación.
Por ejemplo, podría ser interesante conocer si el invierno afecta a los pacientes, o si los fines de semana reportan alguna mejoría.

Q7:"La realización de actividades físicas mejora o empeora el estado de ánimo de los pacientes."

Al igual que nos pasaba con la segunda pregunta planteada, es imposible poder dar una respuesta mediante los datos fuente proporcionados, pues carecemos de la información relativa a los estados de ánimo de un paciente.

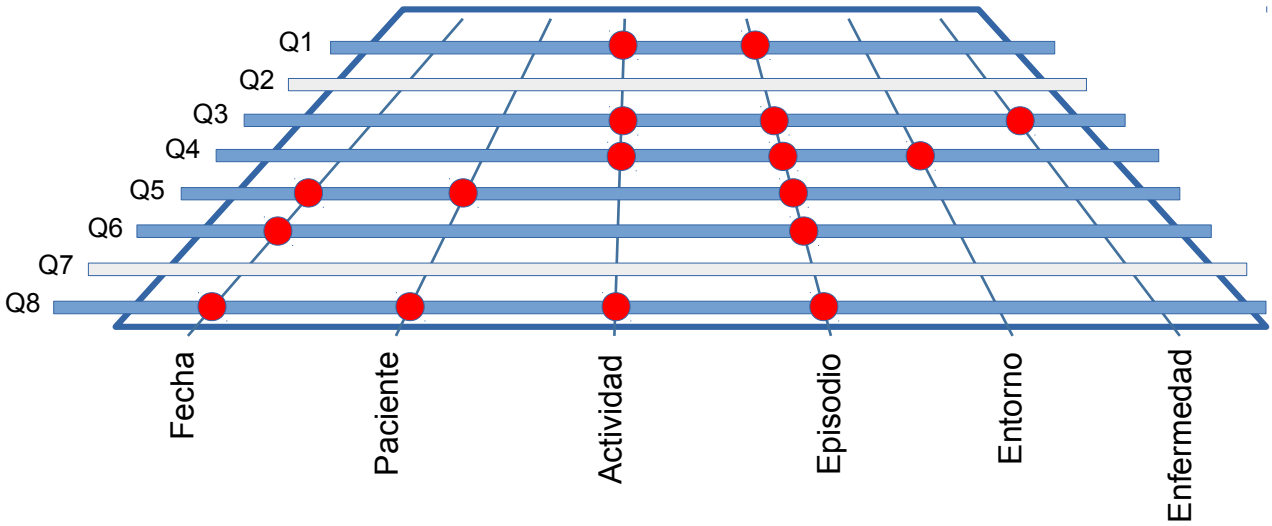
Q8:"¿Hay algún tipo de actividad que mejore el día a día de los pacientes?"

Partimos de la base de que interpretamos como "mejora del día a día" el hecho de que un paciente sufra cada vez menos episodios graves. Con este enfoque, las dimensiones que entran en juego son: **actividad**, **fecha**, **episodio**, **paciente**.

Como conclusión del análisis realizado atendiendo a las preguntas de la propuesta, listamos las siguientes dimensiones:

- Fecha
- Paciente
- Actividad
- Episodio
- Entorno
- Enfermedad

Si entendemos cada una de estas preguntas como un área de negocio, podríamos definir el Bus Matrix de la siguiente forma:

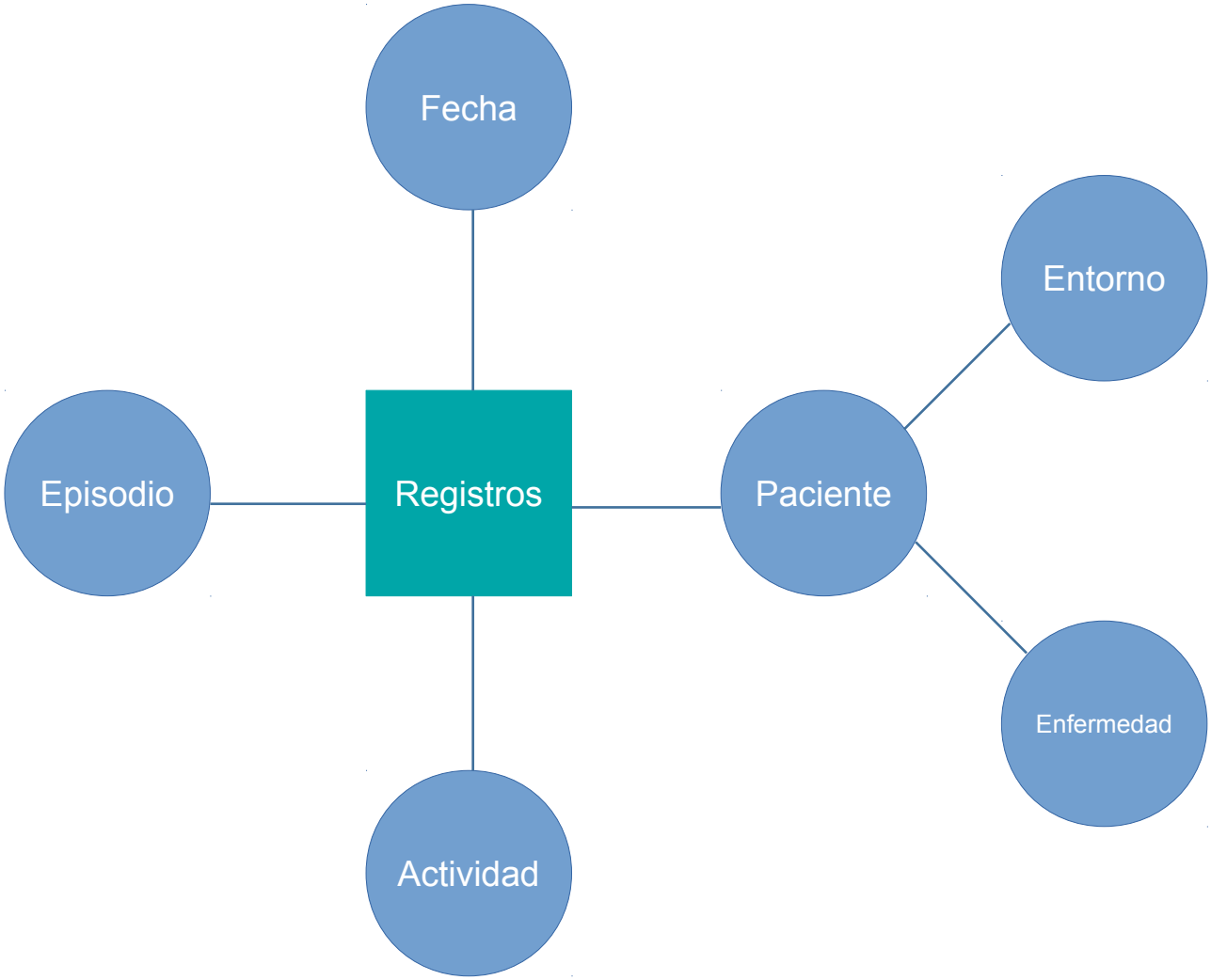


	Fecha	Paciente	Actividad	Episodio	Entorno	Enfermedad
Q1:"¿Cuál es la relación entre las actividades realizadas y los episodios de crisis graves?"			✖	✖		
Q2:"¿Se puede establecer algún tipo de relación entre los valores de los diferentes estados de ánimo y los episodios de crisis? "						
Q3:"Estas relaciones son iguales para cualquiera de las enfermedades o en cambio hay relaciones más acusadas por alguna de ellas."			✖	✖		✖
Q4:"¿Se puede establecer alguna relación en nivel geográfico, por ejemplo entorno urbano o rural?"			✖	✖	✖	
Q5:"¿Cuál sido la evolución de los diferentes pacientes a lo largo del tiempo?"	✖	✖		✖		
Q6:"¿Se puede establecer alguna relación entre los episodios de crisis y el momento del día o de la semana o del año?"	✖			✖		
Q7:"La realización de actividades físicas mejora o empeora el estado de ánimo de los pacientes."						
Q8:"¿Hay algún tipo de actividad que mejore el día a día de los pacientes?"	✖	✖	✖	✖		

Bus Matrix

Modelado dimensional a alto nivel

Una vez detectadas las distintas dimensiones que participarán en el sistema, es el momento de describir gráficamente el modelado a alto nivel.
Siguiendo el patrón de diseño aplicado en en el modelado dimensional, este tendrá una forma de estrella como el que pasamos a mostrar a continuación:



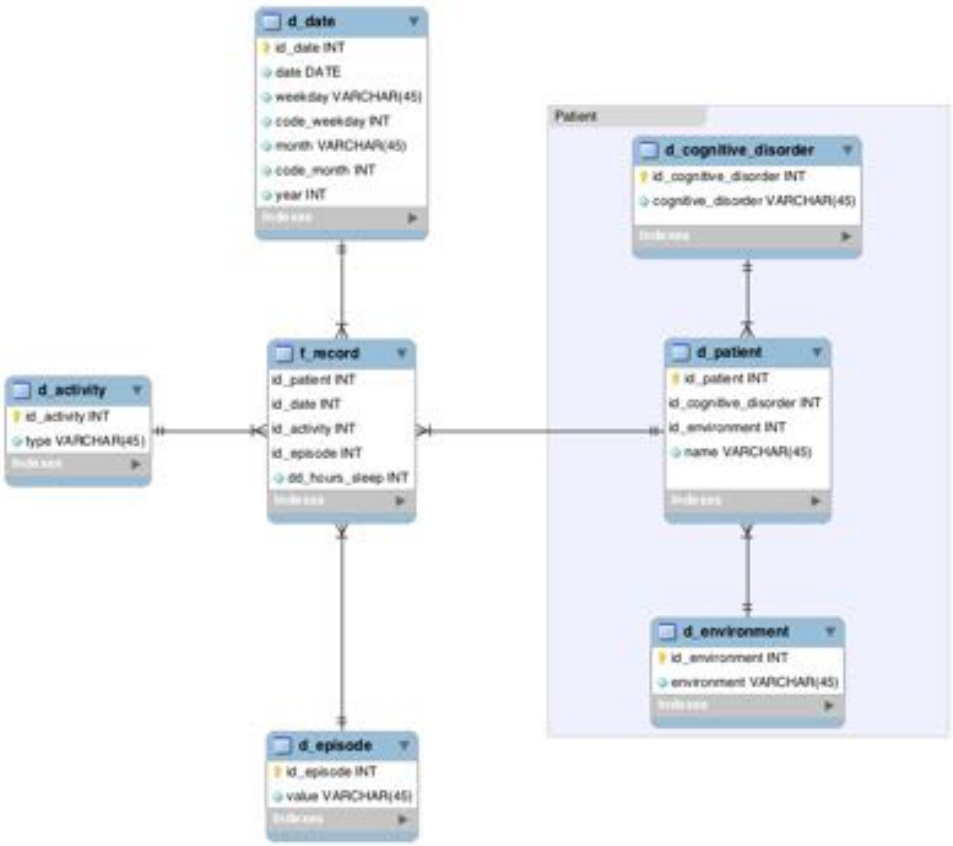
Modelado dimensional a alto nivel

Podemos observar la tabla de hechos llamada **Registros** en el centro del gráfico representada mediante un cuadrado y las distintas dimensiones a su alrededor.
Aunque el modelo se representa mediante un tipo estrella típico de modelos dimensionales, hay que destacar las dimensiones "Entorno" y "Enfermedad". Estas dos dimensiones no estarán relacionadas directamente con la tabla de hechos (**Registros**), sino que son

dependientes de la dimensión "Paciente". De este modo obtenemos un modelo de tipo copo de nieve (Snowflake).
Aunque se deben evitar en lo posible este tipo de soluciones, sacar los valores recogidos de las enfermedades y entornos en dimensiones propias ahorrará un espacio considerable de la tabla "Paciente", ya que estos valores se repetirán constantemente. También, aunque nos acerquemos a un esquema propio de 3FN y nos alejemos de una propuesta dimensional, hay que decir que en este caso no se pierde la legibilidad del modelo y sigue siendo fácil de entender con el objetivo de poder mostrarlo finalmente al área de negocio.

Diseño detallado de las tablas de dimensiones y de hechos

En esta fase pasamos a describir en detalle las tablas que serán creadas en el modelo. Para ello representaremos la tabla de hechos y cada una de las dimensiones, explicando todos los atributos que las componen.



Nombre de la Tabla	f_record
Tipo	Hechos
Nombre de la Vista	Registros
Descripción	Tabla de hechos donde se almacenará la información registrada cada día de los pacientes
Esquemas en los que es usada	cognitive_disorders

	Destino							Fuente					
Nombre Columna	Descripción	Tipo de dato	Key?	FK To	NULL?	Valor por defecto	Ejemplos	Sistema de entrada	Esquema origen	Tabla origen	Nombre del campo origen	Tipo de dato origen	Reglas ETL
id_patient	Identifica al paciente con los datos	int	FK	d_patient	N		1,2,3,4...	Derivado					Se extrae de la dimensión referente
id_date	Identifica la fecha con los datos	int	FK	d_date	N		1,2,3,4...	Derivado					Se extrae de la dimensión referente
id_activity	Identifica la actividad con los datos	int	FK	d_activity	N		1,2,3,4...	Derivado					Se extrae de la dimensión referente
id_episode	Identifica el episodio con los datos	int	FK	d_episode	N		1,2,3,4...	Derivado					Se extrae de la dimensión referente
dd_hours_sleep	Dimensión degenerada. Registra las horas que duerme un paciente	int			N	-1	1,2,3,4...	Excel	DATA SHEET COGNITIVE	HOURS SLEEP VALUES	P1,P2...P20	Number	Convertir a int

Nombre de la Tabla	d_date
Tipo	Dimensión
Nombre de la Vista	Fecha
Descripción	Esta tabla almacenará la información relevante a los datos de fechas
Esquemas en los que es usada	cognitive_disorders

	Destino							Fuente					
Nombre Columna	Descripción	Tipo de dato	Key?	FK To	NULL?	Valor por defecto	Ejemplos	Sistema de entrada	Esquema origen	Tabla origen	Nombre del campo origen	Tipo de dato origen	Reglas ETL
id_date	Clave primaria de la fecha	int	PK		N		1,2,3,4...	ETL					
date	Fecha obtenida directamente de la fuente	date			N		2016-01-01, 2016-12-31	Excel	DATA SHEET COGNITIVE	HOURS SLEEP VALUES, ACTIVITY VALUES, EPISODE VALUES	FECHA	DATE (DD/MM/AAAA)	Hay que pasar el formato al tipo date de MySQL (YYYY-MM-DD)
weekday	Día de la semana asociada a la fecha	string			N		Mon, Tue, Wed...	ETL					Hay que calcular el día de la semana que corresponde a la fecha de entrada
code_weekday	Código que hace referencia al día de la semana	int			N		1,2...7	ETL					Se realiza el mismo cálculo que con weekday
month	Mes	string			N		Jan, Feb...	ETL					Hay que calcular el mes que corresponde a la fecha de entrada
code_month	Código que hace referencia al mes	int			N		1,2,3...12	ETL					Se realiza el mismo cálculo que con month
year	Año	int			N		2016, 2017...	ETL					Se extrae el año de la fecha

Nombre de la Tabla	d_patient
Tipo	Dimensión
Nombre de la Vista	Paciente
Descripción	Esta tabla almacenará la información relevante a los pacientes
Esquemas en los que es usada	cognitive_disorders

	Destino							Fuente					
Nombre Columna	Descripción	Tipo de dato	Key?	FK To	NULL?	Valor por defecto	Ejemplos	Sistema de entrada	Esquema origen	Tabla origen	Nombre del campo origen	Tipo de dato origen	Reglas ETL
id_patient	Clave primaria del paciente	int	PK		N		1,2,3,4...	ETL					
id_cognitive_disorder	Identifica la enfermedad con el paciente	int	FK	d_cognitive_disorder	N		1,2,3,4...	Derivado					Se extrae de la dimensión referente
id_environment	Identifica la ciudad con el paciente	int	FK	d_environment	N		1,2,3,4...	Derivado					Se extrae de la dimensión referente
name	Nombre del paciente	string			N		P1, P2, ... P20	Excel	DATA SHEET COGNITIVE	PATIENTS	PATIENT	Number	Convertir a VARCHAR

Nombre de la Tabla	d_activity
Tipo	Dimensión
Nombre de la Vista	Actividad
Descripción	Esta tabla almacenará la información relevante a las actividades
Esquemas en los que es usada	cognitive_disorders

	Destino							Fuente					
Nombre Columna	Descripción	Tipo de dato	Key?	FK To	NULL?	Valor por defecto	Ejemplos	Sistema de entrada	Esquema origen	Tabla origen	Nombre del campo origen	Tipo de dato origen	Reglas ETL
id_activity	Clave primaria de la actividad	int	PK		N		1,2,3,4...	ETL					
type	Nombre del tipo de actividad	string			N	NO ACTIVITY	SIN ACTIVIDAD CONCRETA, REUNIONES FAMILIARES, RADIO/TV, DORMIR/SOFA, ACTIVIDAD FISICA	Excel	DATA SHEET COGNITIVE	ACTIVITY VALUES	P1, P2, ...	Number	Hay que procesar los distintos tipos, y convertirlos a string

Nombre de la Tabla	d_episode
Tipo	Dimensión
Nombre de la Vista	Episodio
Descripción	Esta tabla almacenará la información relevante a las actividades
Esquemas en los que es usada	cognitive_disorders

	Destino							Fuente					
Nombre Columna	Descripción	Tipo de dato	Key?	FK To	NULL?	Valor por defecto	Ejemplos	Sistema de entrada	Esquema origen	Tabla origen	Nombre del campo origen	Tipo de dato origen	Reglas ETL
id_episode	Clave primaria del episodio	int	PK		N		1,2,3,4...	ETL					
value	Valor del episodio	string			N	NO EPISODE	BAJO, LEVE, MODERADO, GRAVE	Excel	DATA SHEET COGNITIVE	EPISODE VALUES	P1, P2, ...	Number	Hay que procesar los distintos tipos, y convertirlos a string

Nombre de la Tabla	d_cognitive_disorder
Tipo	Dimensión
Nombre de la Vista	Enfermedad
Descripción	Esta tabla almacenará la información relevante a las enfermedades
Esquemas en los que es usada	cognitive_disorders

	Destino							Fuente					
Nombre Columna	Descripción	Tipo de dato	Key?	FK To	NULL?	Valor por defecto	Ejemplos	Sistema de entrada	Esquema origen	Tabla origen	Nombre del campo origen	Tipo de dato origen	Reglas ETL
id_cognitive_disorder	Clave primaria de la enfermedad	int	PK		N		1,2,3,4...	ETL					
cognitive_disorder	Nombre de la enfermedad	string			N		DELIRIO, DEMENCIA, AMNESIA	Excel	DATA SHEET COGNITIVE	Patients	COGNITIVE DISORDERS	Number	Hay que procesar las distintas enfermedades, y convertirlas a string

Nombre de la Tabla	d_environment
Tipo	Dimensión
Nombre de la Vista	Entorno
Descripción	Esta tabla almacenará la información relevante a los entornos donde viven los pacientes
Esquemas en los que es usada	cognitive_disorders

	Destino							Fuente					
Nombre Columna	Descripción	Tipo de dato	Key?	FK To	NULL?	Valor por defecto	Ejemplos	Sistema de entrada	Esquema origen	Tabla origen	Nombre del campo origen	Tipo de dato origen	Reglas ETL
id_environment	Clave primaria del entorno	int	PK		N		1,2,3,4...	ETL					
environment	Tipo de entorno	string			N		RURAL, SEMIRURAL, URBANO	Excel	DATA SHEET COGNITIVE	Patients	ENVIRONMENT	Number	Hay que procesar los distintos entornos, y convertirlos a string

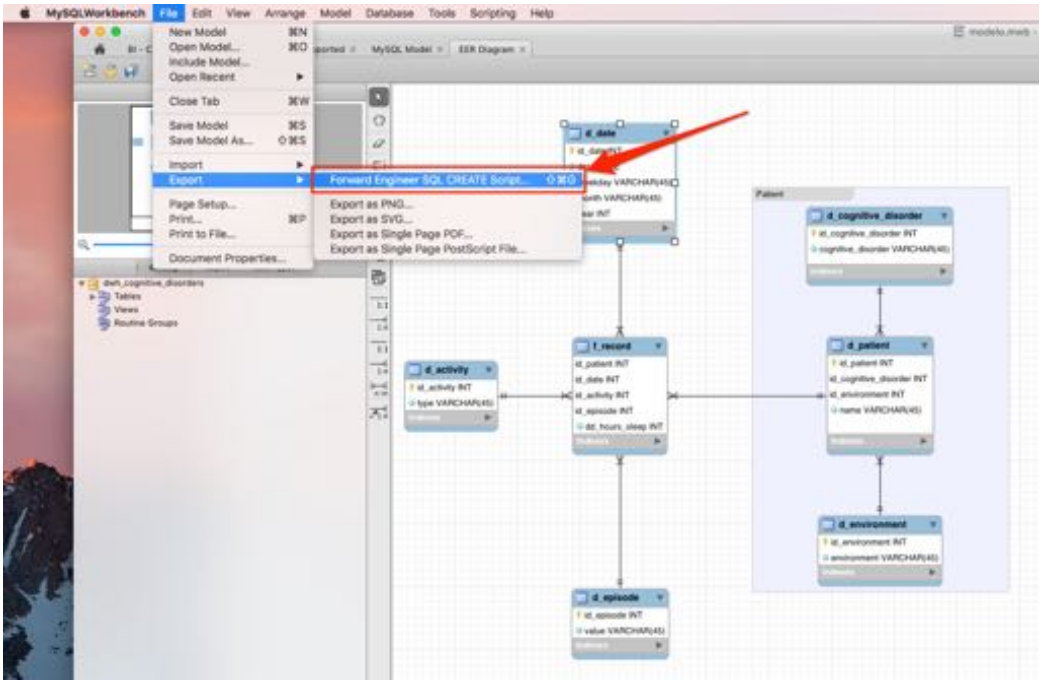
2.3. Diseño físico del Data Warehouse

Una vez completado el diseño de la base de datos, pasamos a implementar físicamente los requisitos detallados en el apartado anterior.
Para llevar a cabo este proceso hemos creado una máquina virtual mediante VirtualBox [5], donde hemos instalado la versión 16.04 de Ubuntu Server [6].

```
rflores@ubuntu-bi:~$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 16.04.2 LTS
Release:        16.04
Codename:       xenial
```

Por otro lado, hemos tomado la decisión de albergar la información de nuestro sistema en MariaDB [7], firme sucesor de MySQL y que nos va a brindar una completa compatibilidad con el proyecto así como con nuestro servidor Ubuntu.

Para la ejecución de esta tarea nos hemos apoyado en la herramienta MySQL Workbench [8], que entre otras características, nos permite la posibilidad de exportar a código SQL el modelo diseñado mediante su interfaz gráfica (File>Export>Forward Engineer SQL CREATE Script...).



El resultado del fichero exportado tendrá un contenido parecido al que mostramos en el siguiente extracto:

```
-- MySQL Script generated by MySQL Workbench
-- Sat Apr 22 16:01:49 2017
-- Model: New Model    Version: 1.0
-- MySQL Workbench Forward Engineering

SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECKS=0;
SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FOREIGN_KEY_CHECKS=0;
SET @OLD_SQL_MODE=@@SQL_MODE,
SQL_MODE='TRADITIONAL,ALLOW_INVALID_DATES';

-- -----------------------------------------------------
-- Schema dwh_cognitive_disorders
-- -----------------------------------------------------

-- -----------------------------------------------------
-- Schema dwh_cognitive_disorders
-- -----------------------------------------------------
CREATE SCHEMA IF NOT EXISTS `dwh_cognitive_disorders` DEFAULT CHARACTER
SET utf8 ;
USE `dwh_cognitive_disorders` ;

-- -----------------------------------------------------
-- Table `dwh_cognitive_disorders`.`d_cognitive_disorder`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS
`dwh_cognitive_disorders`.`d_cognitive_disorder` (
  `id_cognitive_disorder` INT NOT NULL AUTO_INCREMENT,
  `cognitive_disorder` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`id_cognitive_disorder`))
ENGINE = InnoDB;


-- -----------------------------------------------------
-- Table `dwh_cognitive_disorders`.`d_environment`
-- -----------------------------------------------------
CREATE TABLE IF NOT EXISTS `dwh_cognitive_disorders`.`d_environment` (
  `id_environment` INT NOT NULL AUTO_INCREMENT,
  `environment` VARCHAR(45) NOT NULL,
  PRIMARY KEY (`id_environment`))
ENGINE = InnoDB;
...
```

Este fichero deberemos trasladarlo a nuestro servidor donde ejecutaremos el siguiente comando para su importación:

```
$sudo mysql -u root < create_dwh.sql
```

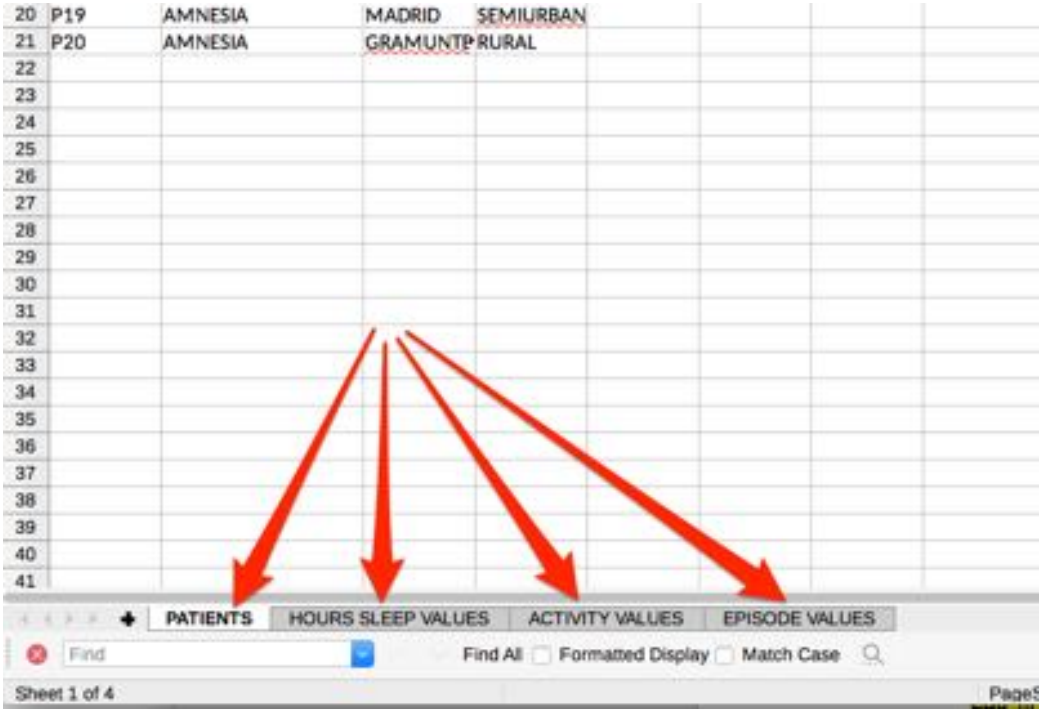
De esta forma ya disponemos de la base de nuestro DWH.
Debido a que durante el proceso de ETL que abordaremos a continuación podemos encontrarnos varias veces con la necesidad de "resetear" nuestra base de datos al estado inicial (vacía), es recomendable hacer un "dump" de esta primera instancia:

```
$sudo mysqldump -u root dwh_cognitive_disorders >
VACIA_dwh_cognitive_disorders.sql
```

2.4. ETL: Desarrollo de procesos ETL (Extract Transform Load)

En este apartado detallaremos los procesos principales y más interesantes que se han llevado a cabo para acometer la carga de la información contenida en el Excel origen y su almacenamiento final en la base de datos. Asimismo analizaremos las opciones que la herramienta Kettle de la suite Pentaho nos proporciona con el objetivo de poder seguir de guía para otros proyectos.

Una de las premisas de las que partimos antes de comenzar este proceso es que el origen de nuestros datos es un Excel que contiene una cuatro pestañas (DATASHEETSCOGNITIVE.xlsx):



La primera de ellas (PATIENTS) nos servirá para obtener:

- El nombre de los pacientes, que en el caso que nos ocupa están "anonimizados" mediante la codificación P1, P2...P20.
- Los tipos de enfermedad en los cuales se basa el estudio.
- Los distintos entornos en los que viven los pacientes.

La pestaña "HOURS SLEEP VALUES" nos proporciona las horas de sueño de cada paciente para cada día del año. Cabe destacar que tal y como hemos montado el diseño de nuestro DWH, esta pestaña no nos servirá para la construcción de las dimensiones, sino que la tendremos en cuenta una vez comencemos el proceso de carga de hechos.

La pestaña "ACTIVITY VALUES" contiene todas las actividades que un paciente puede realizar. En este caso y mediante nuestra herramienta de ETL deberemos sacar el conjunto total de actividades que se presentan en la hoja de cálculo. Es importante tener en cuenta que no debemos centrarnos en un paciente/columna únicamente pues no podemos asegurar que este presente en su información todas las actividades posibles. Más adelante detallaremos la solución final optada para resolver este problema.

Por último, la pestaña "EPISODE VALUES" tiene una estructura similar a la de "ACTIVITY VALUES", pero en este caso se registran los grados de los episodios sufridos por cada paciente por día del año. Debido a la similitud en cuanto a estructura con al pestaña anterior, el proceso de extracción, transformación y carga de la información será muy parecido.

Para la ejecución de estos pasos hemos separado el proceso en dos trabajos:

- **j_etl.kjb**: Compondrá las distintas dimensiones
- **j_etl_facts.kjb**: Rellenará la tabla de hechos

Cada uno de estos trabajos llevará a cabo una serie de transformaciones que serán los encargados de procesas la información.

Para el trabajo que construirá las dimensiones (j_etl.kjb) se han creado:

- **t_d_activities.ktr**: Encargado de procesar la información sobre las actividades provenientes del Excel.
- **t_d_activities_load.ktr**: Carga final de los datos obtenido en la transformación anterior (t_d_activities.ktr).
- **t_d_cognitive_environment.ktr**: Encargado de procesar la información sobre los entornos provenientes del Excel y almacenarlos directamente en la tabla correspondiente.
- **t_d_date.ktr**: Generación de una tabla dimensión fecha para los días del año 2016.
- **t_d_episodes.ktr**: Encargado de procesar la información sobre los episodios provenientes del Excel.
- **t_d_episodes_load.ktr**: Carga final de los datos obtenido en la transformación anterior (t_d_episodes.ktr).
- **t_d_patients.ktr**: Encargado de procesar la información sobre los pacientes provenientes del Excel.
- **t_d_patients_load.ktr**: Carga final de los datos obtenido en la transformación anterior (t_d_patients.ktr).

Podemos observar como hay dimensiones que se cargan en dos transformaciones (hemos añadido el sufijo "_load" a estos procesos). Esto es necesario debido al carácter transaccional de la herramienta, donde a veces necesitamos guardar la información en una base de datos de "staging" en una primera transformación, para poder valernos de su estructura en una segunda carga.

Para el trabajo que construirá los hechos (j_etl_facts.kjb) se han creado:

- **t_f_getHours.ktr**: Transformación que obtendrá las horas de sueño del Excel y almacenará en la base de datos de "staging" con el objetivo de cargarlas en la transformación final (t_f_records.ktr).
- **t_f_getPatients.ktr**: Transformación que nos sirve para pasar los distintos pacientes como parámetros a la siguiente transformación (t_f_records.ktr).
- **t_f_records.ktr**: Transformación que se ejecutará tantas veces como pacientes le lleguen de la transformación anterior (t_f_getPatients.ktr) y que irá componiendo en cada ejecución la tabla de hechos.

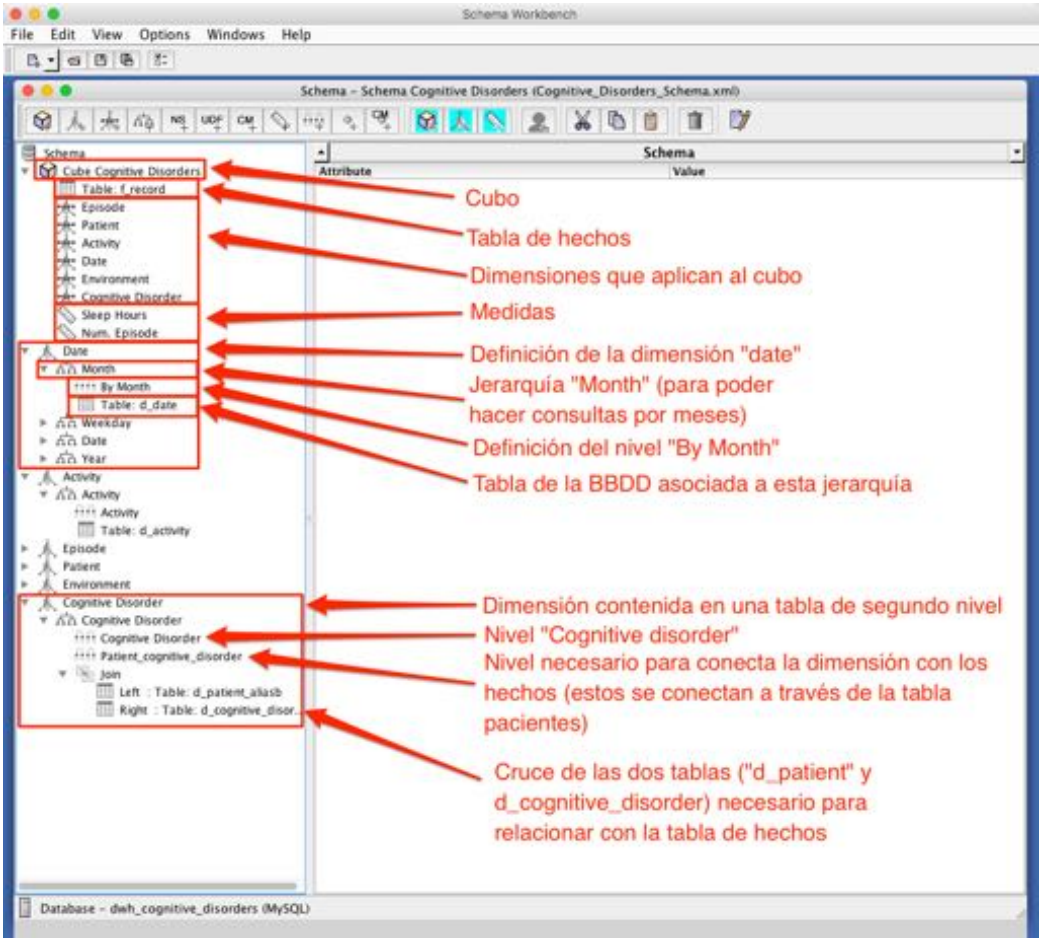
En los anexos de esta memoria (apartados 6.1, 6.2 y 6.3) se puede consultar con mayor detalle los pasos llevados a cabo para la creación de las transformaciones más interesantes de este proyecto, donde explicamos también algunas peculiaridades de la plataforma Pentaho, que creemos puede ser de utilidad al lector.

2.5. Diseño del cubo OLAP

El siguiente paso para construir nuestra plataforma de BI es diseñar el cubo OLAP que procesará la información que contiene nuestra base de datos. Una vez hemos ejecutado todos los pasos definidos en el proceso de ETL y constatamos que nuestra BBDD se ha rellenado con la información haremos uso de la aplicación "Schema Workbench" de la suite Pentaho.

Mediante esta aplicación podremos ir configurando las opciones del cubo de forma visual.
El resultado de esta operación es un fichero XML que define las propiedades de las dimensiones, las medidas con las que vamos a operar así como las características generales del cubo. Aunque estos diseños pueden llegar a ser mucho más complejos, el caso que nos ocupa es un planteamiento sencillo.

Mostramos a continuación el resultado final.



El primer paso para la creación de un cubo es crear un nuevo esquema. Hemos llamado a nuestro esquema "Schema Cognitive Disorders".

Luego los siguientes pasos son crear el cubo y sus dimensiones. Como hemos podido apreciar en la captura anterior, hemos definido las seis dimensiones y dos medidas.

Dimensiones:

1. Date
2. Activity
3. Episode
4. Patient
5. Environment
6. Cognitive Disorder

Medidas:

1. Sleep Hours
2. Num. Episode

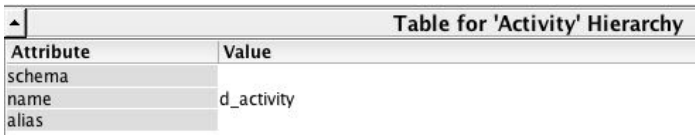
Las seis dimensiones han sido definidas como "Shared Dimensions", esto quiere decir que podrían ser usadas en otros cubos si así fuese necesario. Caben destacar las siguientes:

Dimensión "Activity"

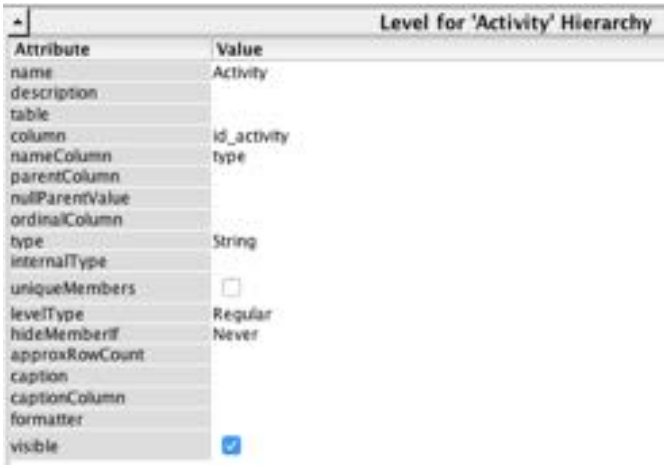
Esta **dimensión** es una de las más sencillas, pero por su carácter genérico merece la pena explicar su estructura. Luego, el resto de dimensiones cumplirá esta base, aunque añadiendo algunas variaciones para cumplir ciertas condiciones.

Como base, una dimensión contendrá una **jerarquía**, este caso "Activity" (coincidiendo con el nombre de la dimensión) donde definiremos el nombre de la misma y si queremos que sea visible o no.

Cada jerarquía tiene **asociada una tabla de la base de datos**, esta tabla puede ser un cruce de varias tablas como veremos más adelante, pero para esta dimensión basta con definir la tabla de la dimensión actividades "d_activity".



Dentro de la jerarquía tendremos que definir también un **nivel**, que será donde configuremos los datos que deben extraerse de la tabla definida en la jerarquía y como deben extraerse.

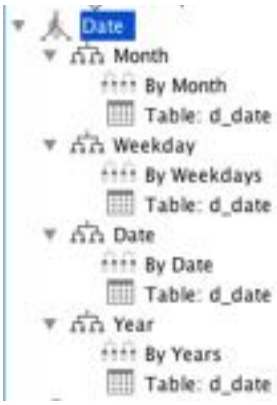


En esta captura podemos ver como definimos el nombre del nivel "name", la columna que contiene la clave primaria de esta tabla "column", la columna que contiene el valor que queremos representar "nameColumn" y el tipo de dato "type".

Dimensión "Date"

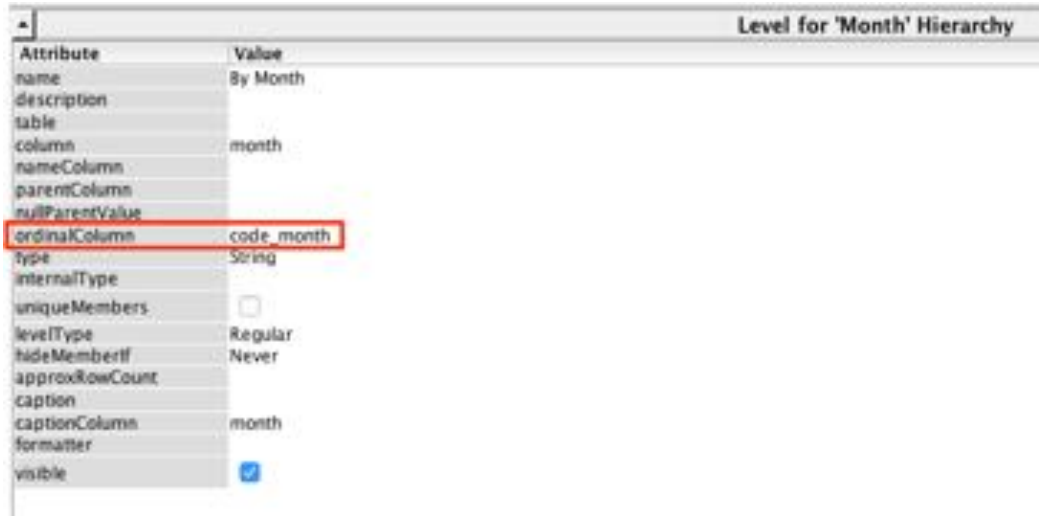
En el caso de la dimensión "Date", con el objetivo de poder realizar consultas tanto a nivel de fecha como de año, mes o día de la semana, hemos definido cuatro jerarquías dentro de la dimensión:

- Month
- Weekday
- Date
- Year



Lo único que variará de cada definición es la columna de consulta de la tabla así como el tipo de dato.
Cabe destacar los casos de las jerarquías "Month" y "Weekday". Debido a que vamos a mostrar los nombres de los meses (Jan, Feb, Mar....Dec) y los nombres de los días de la semana (Mon, Tue, Wed...Sun),

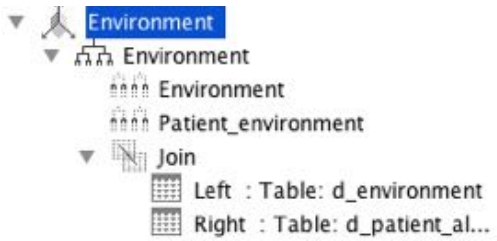
debemos hacer uso de las columnas "code_month" y "code_weekday" que creamos durante el diseño del modelo de datos. Estas columnas contienen el número del mes y del día de la semana respectivamente, de esta forma, si tenemos el valor "Mar" en nuestra columna "month" en la columna "code_month" tendremos el valor "3", por otro lado si tenemos el valor "Thu" en la columna "weekday", tendremos un "5" en la columna "code_weekday" (IMPORTANTE: para Pentaho los días de la semana comienzan el domingo, por lo que el orden será "Sun, Mon, Tue...Sat").
Una vez explicada la situación, cuando estemos diseñando el nivel correspondiente, deberemos indicar cuál es la columna por la que queremos que ordene la información ("code_month", "code_weekday"). Para ello usaremos el atributo "ordinalColumn".



Dimensión "Environment"

La dimensión "Environment", al igual que la dimensión "Cognitive Disorder", depende directamente de la dimensión "Patient" para poder ser relacionada con los hechos. Esto quiere decir que necesitamos relacionar primero la tabla de entornos "d_environment" con la tabla pacientes "d_patient" para poder sacar un nexo con los hechos "f_record".

Para poder acometer esta relación, en este caso, en vez de definir una tabla para nuestra jerarquía, deberemos definir un elemento "join", donde configuraremos las dos tablas que queremos cruzar ("d_patient" y "d_environment").



Es importante señalar que debido a que tenemos una dimensión paciente en nuestro diseño, donde ya hemos hecho uso de la tabla "d_patient" debemos definir un alias para este nuevo uso ("d_patient_alias").

Table for Join

Attribute	Value
schema	
name	d_patient
alias	d_patient_alias

Luego finalmente, en la configuración de nuestro "join" tenemos que indicar cuales son las claves que vamos a usar para relacionar ambas tablas ("id_environment" en el caso que nos ocupa).

Join for 'Environment' Hierarchy

Attribute	Value
leftAlias	
leftKey	id_environment
rightAlias	d_patient_alias
rightKey	id_environment

Finalmente definimos los niveles necesarios para esta dimensión. La configuración de estos es muy parecida a los definidos anteriormente, sólo que necesitamos dos, puesto que estamos haciendo una relación entre ambos datos y tenemos que prestar especial atención al origen de cada uno de los elementos ya que hemos definido dos tablas.

Level for 'Environment' Hierarchy

Attribute	Value
name	Environment
description	
table	d_environment
column	id_environment
nameColumn	environment
parentColumn	
nullParentValue	
ordinalColumn	
type	String
internalType	
uniqueMembers	
levelType	Regular
hideMemberIf	Never
approxRowCount	
caption	
captionColumn	environment
formatter	
visible	

Level for 'Environment' Hierarchy

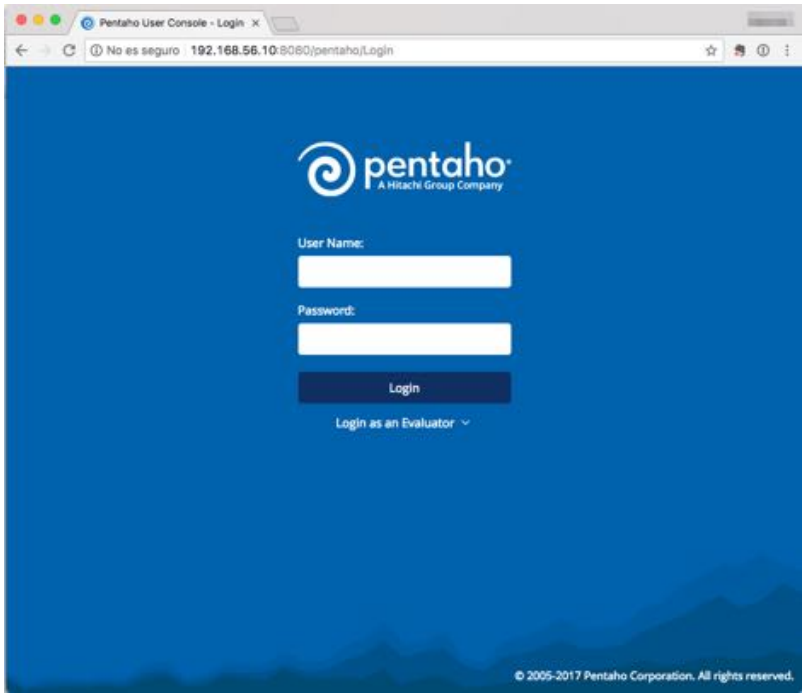
Attribute	Value
name	Patient_environment
description	
table	d_patient_alias
column	id_patient
nameColumn	id_patient
parentColumn	
nullParentValue	
ordinalColumn	
type	Integer
internalType	
uniqueMembers	
levelType	Regular
hideMemberIf	Never
approxRowCount	
caption	
captionColumn	id_patient
formatter	
visible	

2.6. Instalación de una plataforma de consulta

Una vez que hemos finalizado el diseño de nuestro cubo OLAP, el siguiente paso es poder subirlo a una herramienta que nos permita realizar consultas, haciendo uso de este esquema en conjunto con nuestro DWH.

Para ello vamos a utilizar la herramienta/plataforma web que ofrece la suite Community de Pentaho: **Business Analytics Platform**.

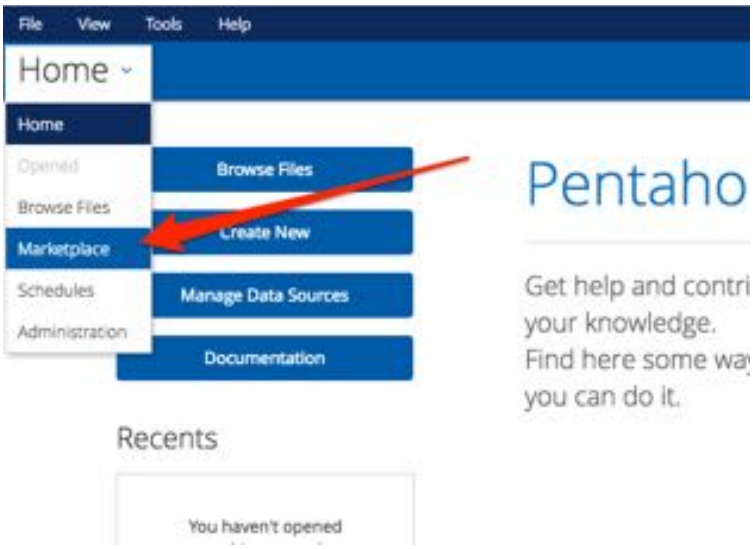
Esta aplicación está desarrollada al igual que las anteriores en Java, pero cuenta con la característica distintiva de ser una aplicación web. Es decir, la utilizaremos a través del navegador. Por ello cuando ejecutamos el script de arranque, estamos levantando un servidor tomcat.
Finalmente, poniendo la dirección IP de nuestro servidor, o su dominio si lo hemos definido, podremos acceder a la aplicación.

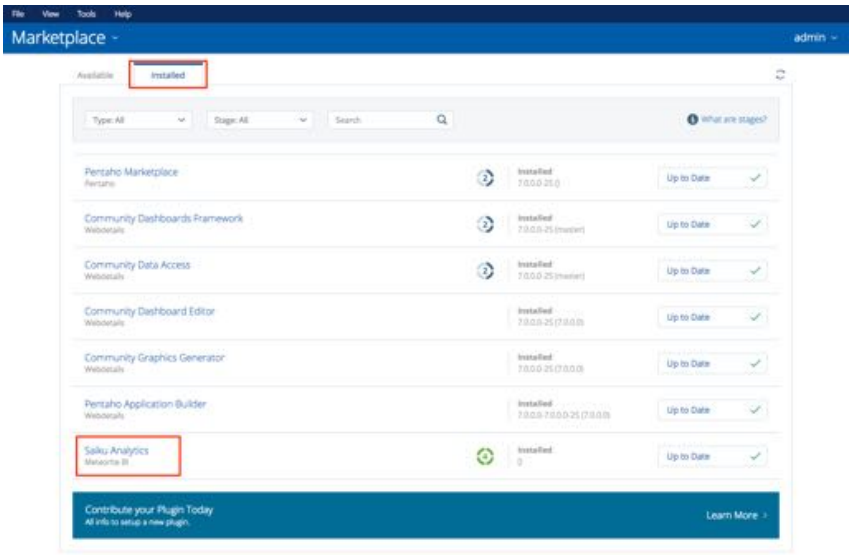


Una vez introduzcamos los credenciales de acceso, veremos la siguiente pantalla, donde destacamos las siguientes opciones:

- Home > Marketplace
- Manage Data Sources
- Tools > Refresh > Mondrian Schema Cache
- Create New

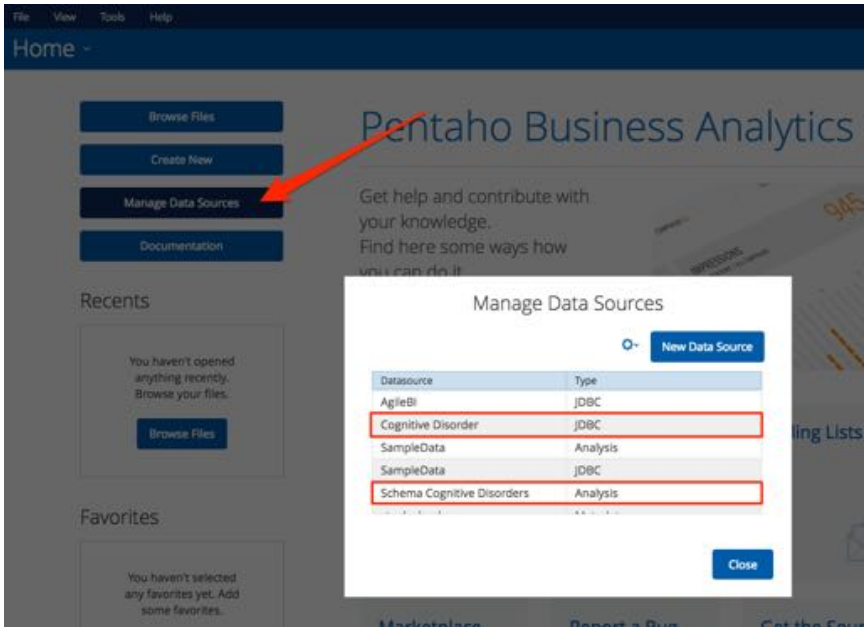
Mediante la sección Marketplace podremos añadir plugins a nuestra aplicación. En nuestro caso veremos que hemos descargado la versión gratuita de Saiku Analytics.





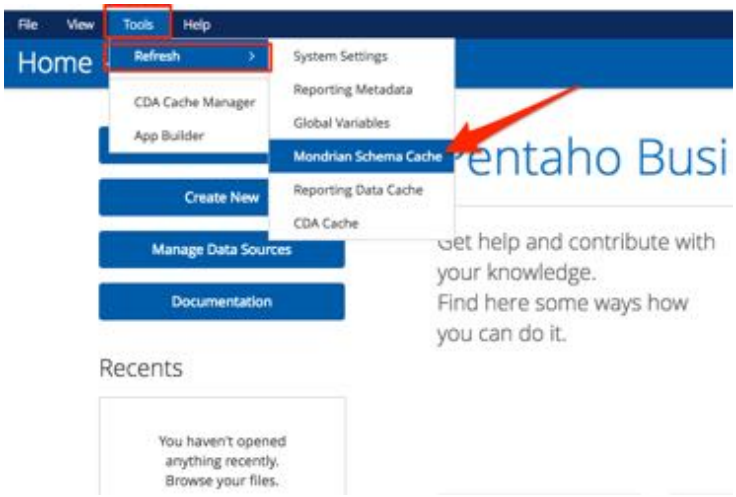
En la sección Manage Data Sources, podremos configurar la conexión con nuestro DWH así como comprobar que se ha subido correctamente nuestro "schema" generado mediante Schema Workbench.

La subida del esquema se puede realizar directamente desde Schema Workbench, pero es importante que el servidor esté levantado.



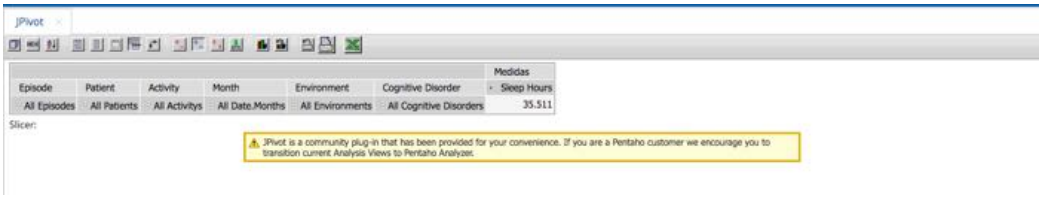
En la captura anterior podemos ver como tenemos una conexión a nuestra BBDD mediante JDBC "Cognitive Disorder", así como el esquema del cubo "Schema Cognitive Disorders".

Es muy normal encontrarnos errores una vez empecemos a trabajar con nuestro cubo, por lo que siempre que subamos nuevas versiones del esquema es muy importante limpiar la caché. Para ello haremos uso de la opción "Mondrian Schema Cache".

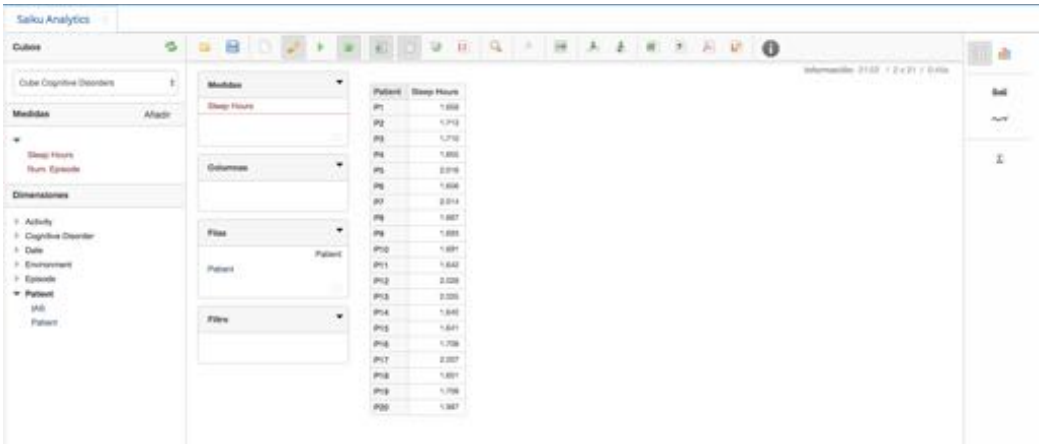


Por último dentro de la opción "Create New" podremos encontrar las herramientas para realizar las consultas. Por defecto Pentaho trae la aplicación JPivot, pero debido a que es un proyecto que ya no tiene mantenimiento y su diseño es algo tosco y poco intuitivo, hemos optado, como mencionamos anteriormente, por hacer uso de otra aplicación (Saiku Analytics) [9].

En la siguiente captura podemos observar el "Look&Feel" de la aplicación JPivot para nuestro cubo.



Ahora mostramos un ejemplo de la aplicación Saiku Analytics para una consulta cualquiera:



Como nota adicional es interesante, sobretodo en el comienzo, tener en un terminal el log del servidor tomcat, ya que este nos puede lanzar cualquier mensaje de aviso sobre problemas que podamos tener, no sólo con la aplicación sino con nuestro cubo.

```
$pentaho-server/tomcat/logs/catalina.out
```

2.7. Informes y respuestas a las preguntas formuladas

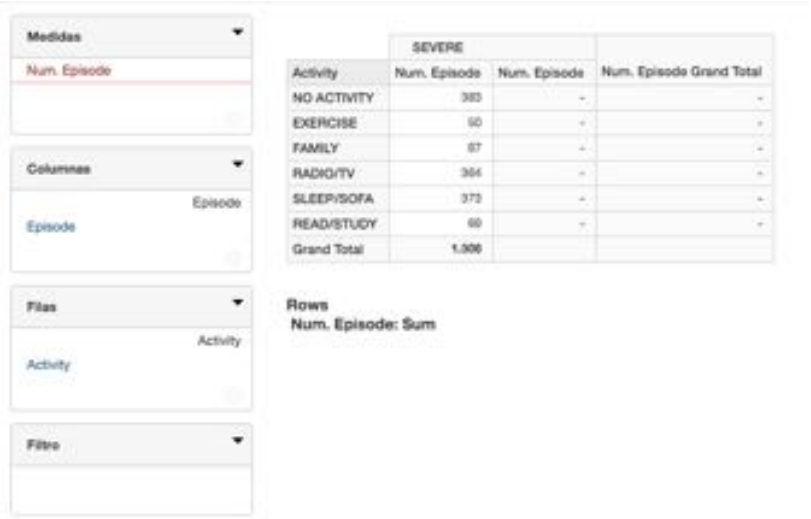
Recordemos que el enunciado nos plantea una serie de cuestiones que debemos responder y que a su vez fueron claves para el diseño de nuestro modelo.
Será en este apartado donde analizaremos esas preguntas y daremos respuesta haciendo uso de la herramienta del servidor de Pentaho y de la aplicación Saiku, con la que "jugaremos" con las dimensiones y medidas con el objetivo de sacar conclusiones.

Pregunta 1

¿Cuál es la relación entre las actividades realizadas y los episodios de crisis graves?

Para dar respuesta a esta pregunta, debemos realizar una consulta donde podamos observar las distintas actividades (dimensión actividad) realizadas por el conjunto de pacientes y el número de episodios (medida número de episodios) que sufren con carácter grave.

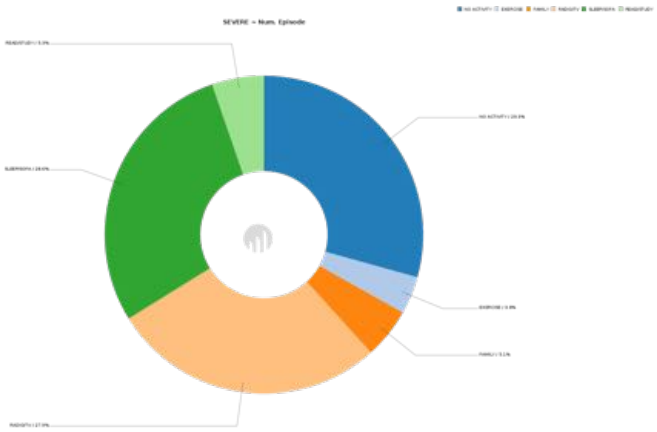
Seleccionaremos como **medida**: Num. Episode
Seleccionaremos como **columna**: Episode (filtrando por "Severe")
Seleccionaremos como **fila**: Activity



Podemos ver fácilmente como las actividades "SLEEP/SOFA", "RADIO/TV y "NO ACTIVITY" son las que más episodios graves generan:

Activity	Num. Episode
SLEEP/SOFA	373
RADIO/TV	364
NO ACTIVITY	383

De igual forma podemos ver mediante la siguiente gráfica los mismos resultados.



Como conclusión de este análisis, podemos sacar que las actividades que menos esfuerzo requieren son aquellas que conllevan más episodios graves.

Pregunta 2

¿Se puede establecer algún tipo de relación entre los valores de los diferentes estados de ánimo y los episodios de crisis?

No, no se puede sacar ningún tipo de relación, pues no disponemos de los datos relacionados con los diferentes estados de ánimo.

Pregunta 3

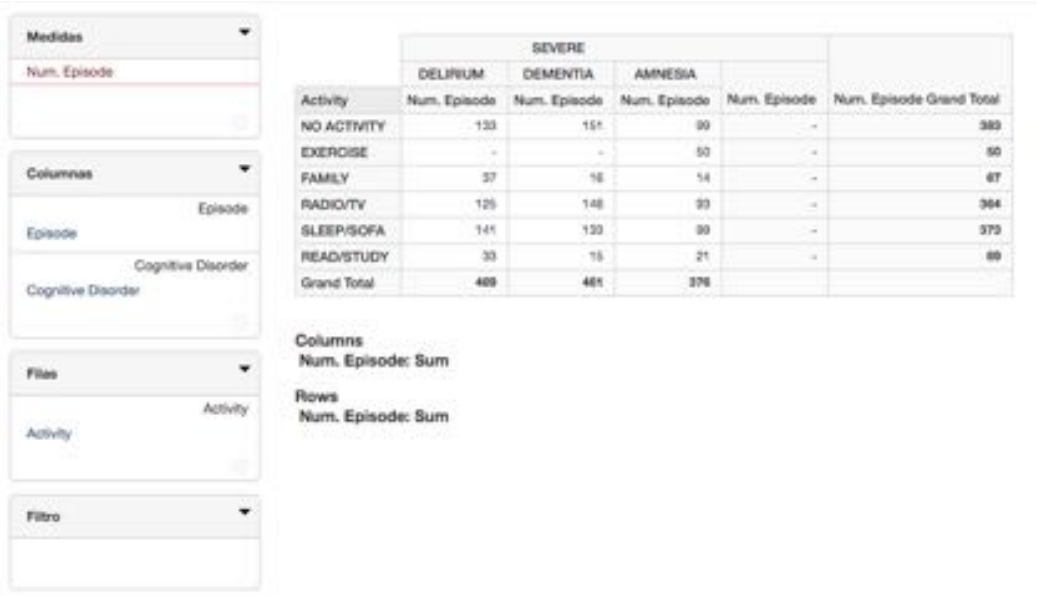
Estas relaciones son iguales para cualquiera de las enfermedades o en cambio hay relaciones más acusadas por alguna de ellas.

Ya que no podemos establecer una relación basada en las condiciones de la pregunta 2, vamos a plantear esta pregunta según las condiciones de la pregunta 1, es decir ¿podemos extraer conclusiones parecidas de los episodios graves y las actividades que realizan los pacientes por cada una de las enfermedades?

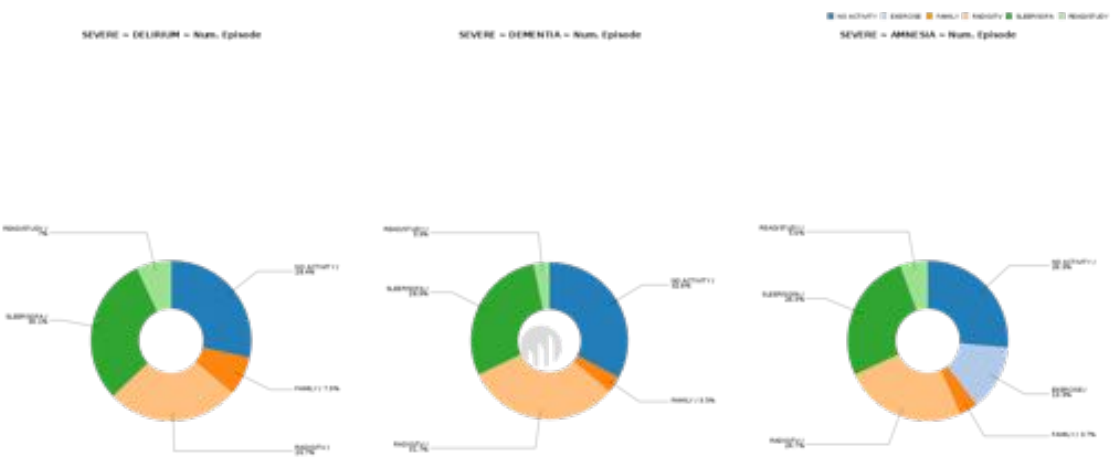
Para ello realizaremos la siguiente consulta:

Seleccionaremos como **medida**: Num. Episode
Seleccionaremos como **columna**: Episode (filtrando por "Severe") y Cognitive Disorder
Seleccionaremos como **fila**: Activity

Podemos comprobar observando el resultado, que los valores de episodios graves, para las tres enfermedades, son siempre mayores en los casos de "NO ACTIVITY", "SLEEP/SOFA" y "RADIO/TV".



Activity	Delirium	Dementia	Amnesia
NO ACTIVITY	133	151	99
RADIO/TV	125	146	93
SLEEP/SOFA	141	133	99



Pregunta 4

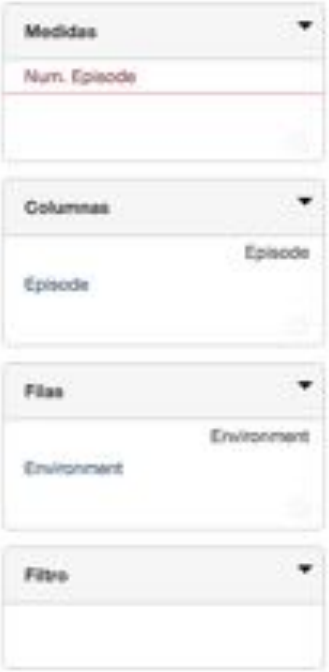
¿Se puede establecer alguna relación en nivel geográfico, por ejemplo entorno urbano o rural?

Sí, al haber creado una dimensión entorno, podemos realizar una consulta donde podamos ver si existe alguna relación entre los entornos y el número de episodios grave que sufre un paciente.

Realizando la siguiente consulta:

Seleccionaremos como **medida**: Num. Episode
Seleccionaremos como **columna**: Episode (filtrando por "Severe")
Seleccionaremos como **fila**: Environment

Podemos observar que los pacientes que se encuentran en entorno "RURAL" son los que menos episodios graves sufren.

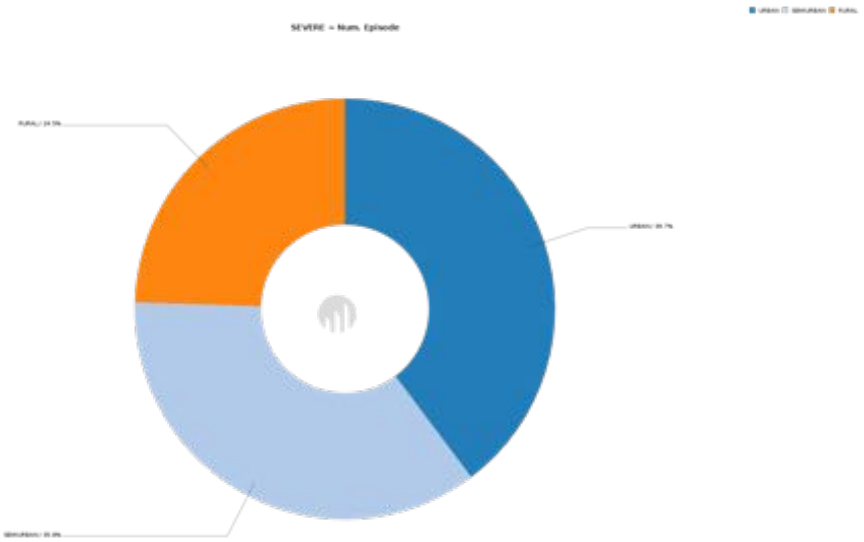


	SEVERE		
Environment	Num. Episode	Num. Episode	Num. Episode Grand Total
URBAN	519	-	519
SEMIURBAN	467	-	467
RURAL	320	-	320
Grand Total	1.306		

Columns
Num. Episode: Sum

Rows
Num. Episode: Sum

Environment	Num. Espisode
URBAN	519
SEMIURBAN	467
RURAL	320



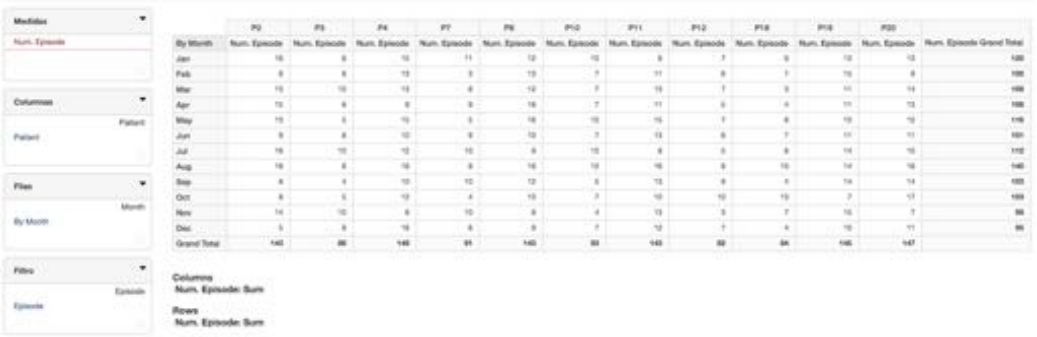
Pregunta 5

¿Cuál ha sido la evolución de los diferentes pacientes a lo largo del tiempo?

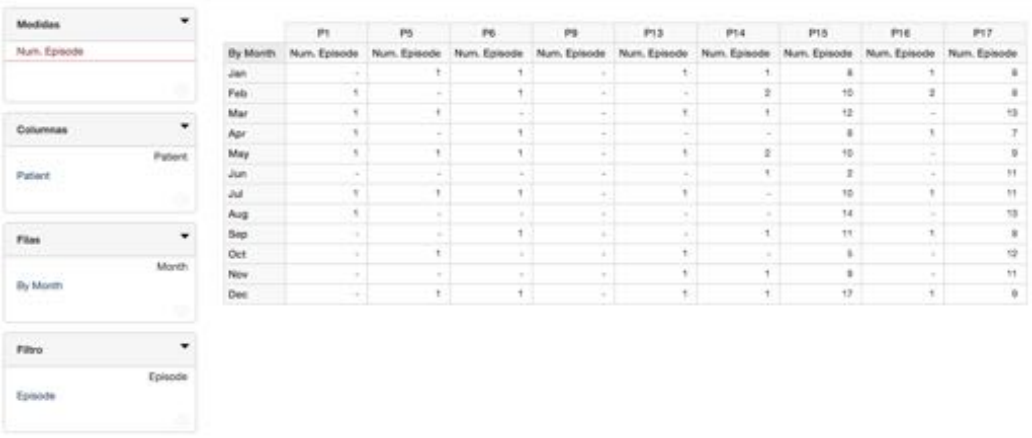
Si nos ceñimos a los episodios graves que sufren los pacientes, y como estos evolucionan a lo largo del año, debemos hacer la siguiente consulta.

Seleccionaremos como **medida**: Num. Episode
Seleccionaremos como **columna**: Patient
Seleccionaremos como **fila**: Date - By Month
Seleccionamos como **filtro**: Episode (**Severe**)

Hemos optado por usar una distribución de fechas por meses, ya que de esta forma podemos ver claramente la evolución a lo largo del año.

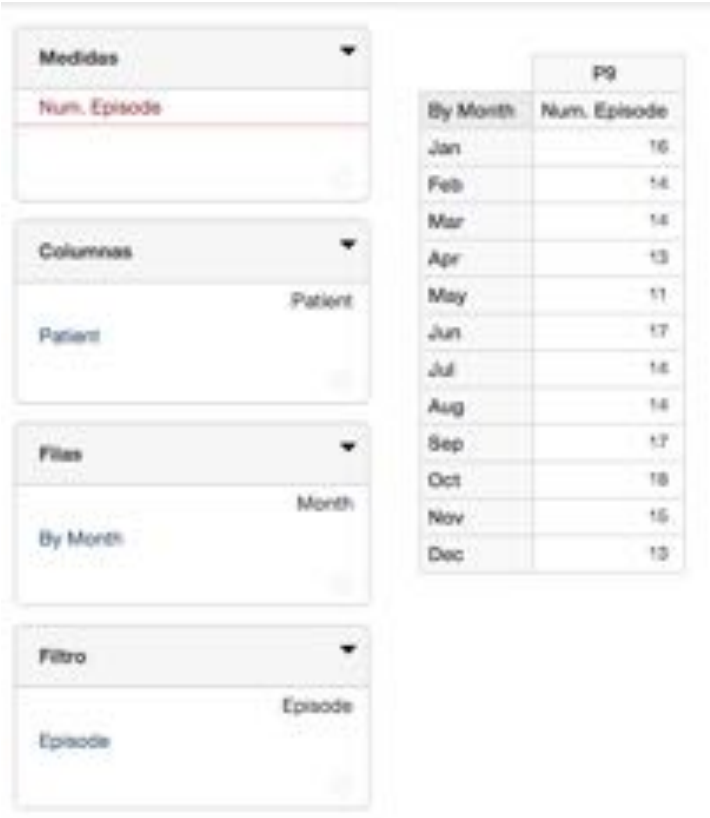


El primer dato curioso que podemos extraer es que los pacientes P1, P5, P6, P9, P13, P14, P15, P16, P17, no han sufrido episodios graves en todo el año y por ello no aparece en nuestra tabla. Si queremos sacar su evolución, podemos cambiar el filtro del tipo de episodio a "**MODERATE**" y seleccionando igualmente esos usuarios.



	P1	P5	P6	P9	P13	P14	P15	P16	P17
By Month	Num. Episode	Num. Episode	Num. Episode	Num. Episode	Num. Episode	Num. Episode	Num. Episode	Num. Episode	Num. Episode
Jan	-	1	1	-	1	1	8	1	8
Feb	1	-	1	-	-	2	10	2	8
Mar	1	1	-	-	1	1	12	-	13
Apr	1	-	1	-	-	-	8	1	7
May	1	1	1	-	1	2	10	-	9
Jun	-	-	-	-	-	1	2	-	11
Jul	1	1	1	-	1	-	10	1	11
Aug	1	-	-	-	-	-	14	-	13
Sep	-	-	1	-	-	1	11	1	8
Oct	-	1	-	-	1	-	5	-	12
Nov	-	-	-	-	1	1	8	-	11
Dec	-	1	1	-	1	1	17	1	8

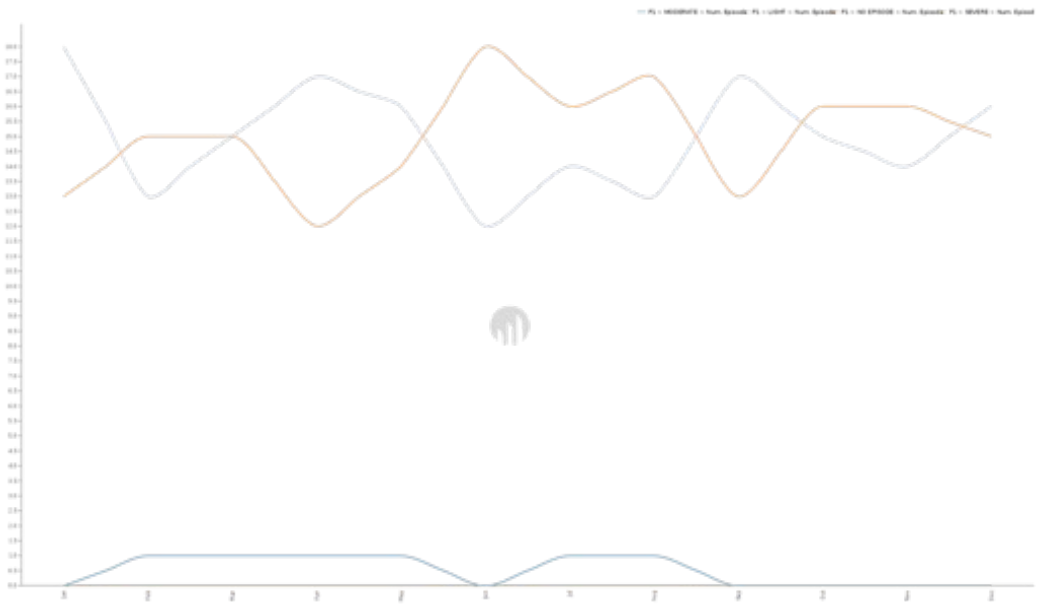
Aún así, también podemos ver como el paciente 9 (P9) tampoco ha sufrido ningún episodio moderado. Realizaremos una última consulta para ver su evolución con respecto a los **episodios leves**.



	P9
By Month	Num. Episode
Jan	16
Feb	14
Mar	14
Apr	13
May	11
Jun	17
Jul	14
Aug	14
Sep	17
Oct	18
Nov	15
Dec	13

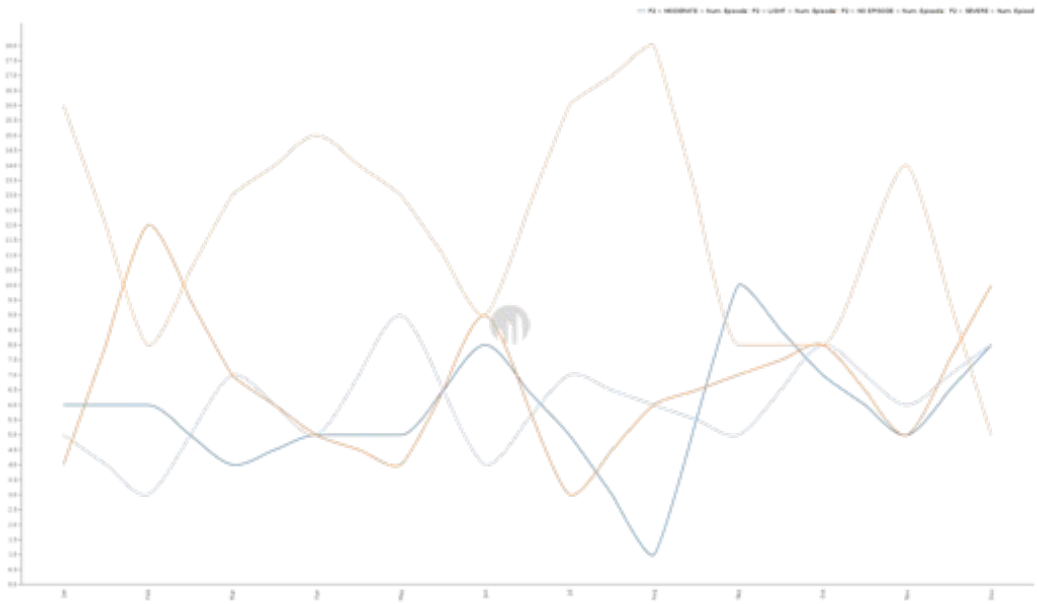
Para mayor claridad en la información, pasaremos a representar gráficamente la evolución de cada uno de los pacientes para cada uno de sus episodios.

Paciente 1 (P1)



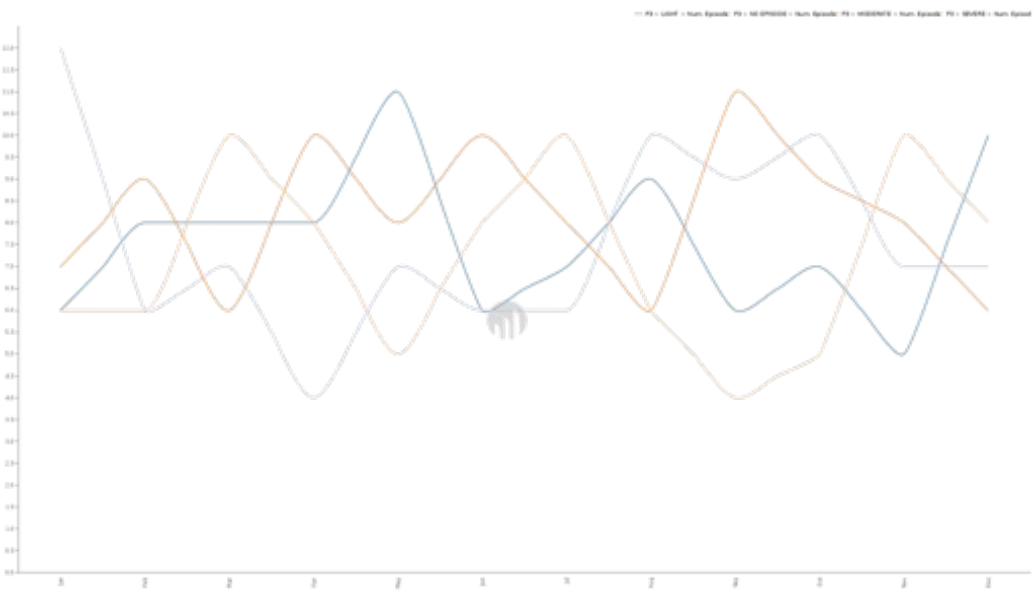
Podemos ver como este paciente no sufre episodios graves en todo el año, así como que ha sufrido muy pocos episodios moderados, desapareciendo estos a partir de septiembre. Por otro lado, los episodios leves y la ausencia de ellos vemos que fluctúan a lo largo del año, siendo el periodo de verano (junio a agosto) la etapa en la que el paciente ha sufrido un menor número de episodios.

Paciente 2 (P2)



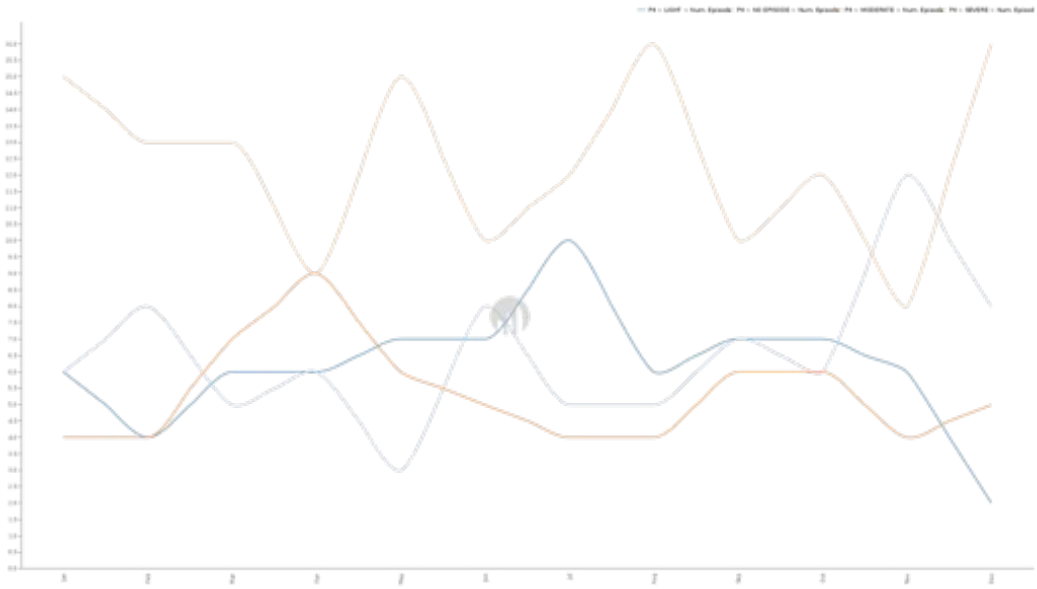
El paciente 2 destaca por haber tenido un mayor brote de episodios graves en el periodo de verano, julio y agosto, aunque podemos observar como este experimenta un mejoría hacia final de año, quedando por debajo del resto de niveles.
En el lado opuesto, podemos ver como los meses en los que ha experimentado una mejoría, son los meses de febrero y diciembre, donde claramente la ausencia de episodios destaca sobre el resto.

Paciente 3 (P3)



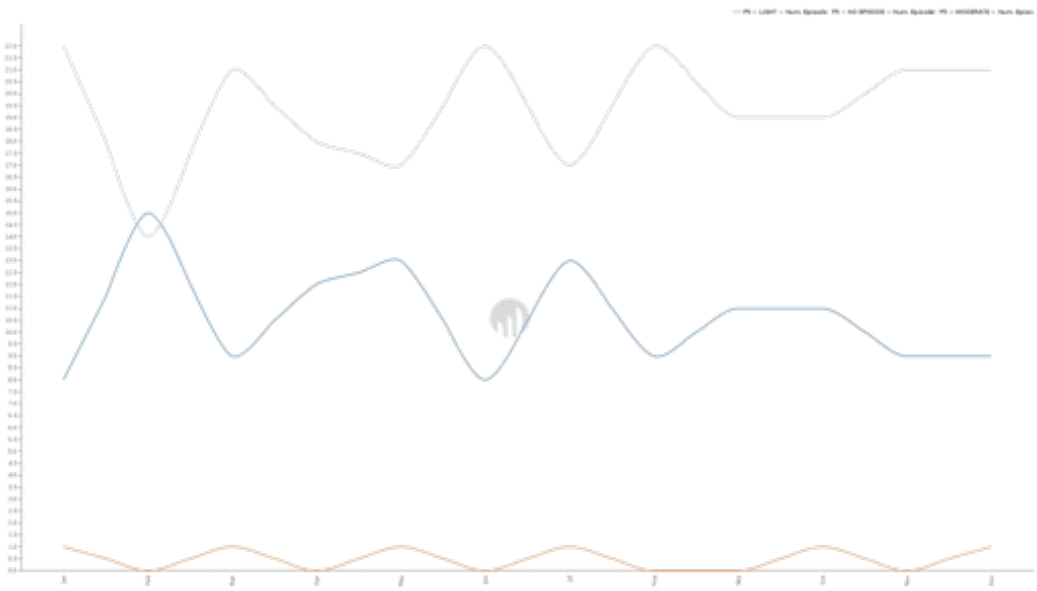
Para el caso del paciente 3, podemos ver como la evolución es muy irregular, no pudiendo esclarecer la tendencia de su estado.
En septiembre se puede apreciar como es el mes en el que menos episodios graves ha sufrido, pero por contra es en el que más episodios moderados ha experimentado. Igualmente al inicio del estudio se aprecia un mayor número de días sin episodios, no alcanzándose nunca más esta cifra.

Paciente 4 (P4)



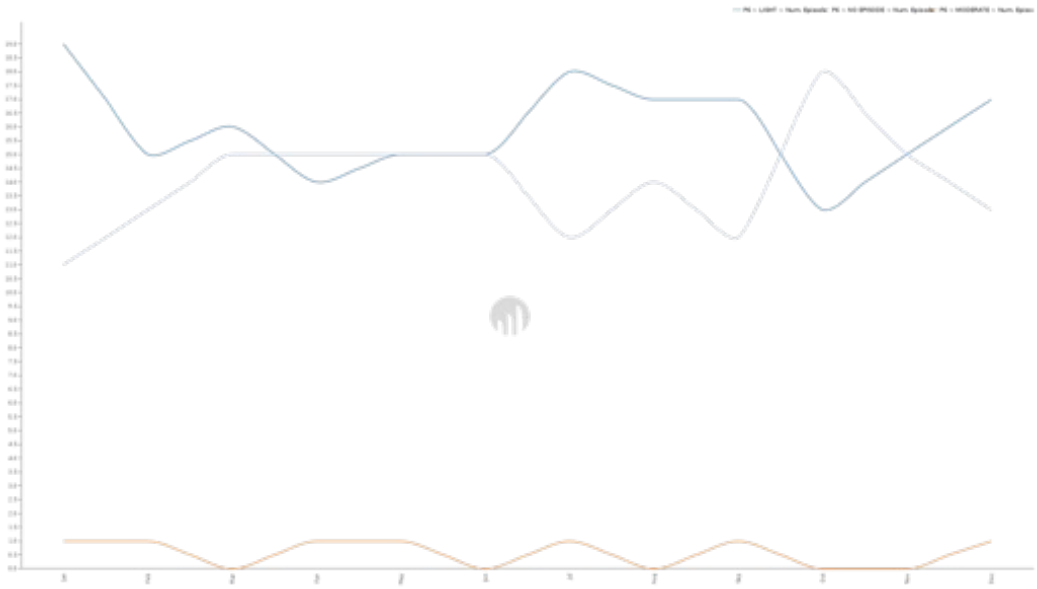
Los episodios que sufre el paciente 4, en todo el estudio, son en su mayoría graves, experimentando una mejora el mes de noviembre donde experimentó el menor número de episodios pero finalizando la etapa de evaluación (mes de diciembre) con la peor cifra de episodios graves.

Paciente 5 (P5)



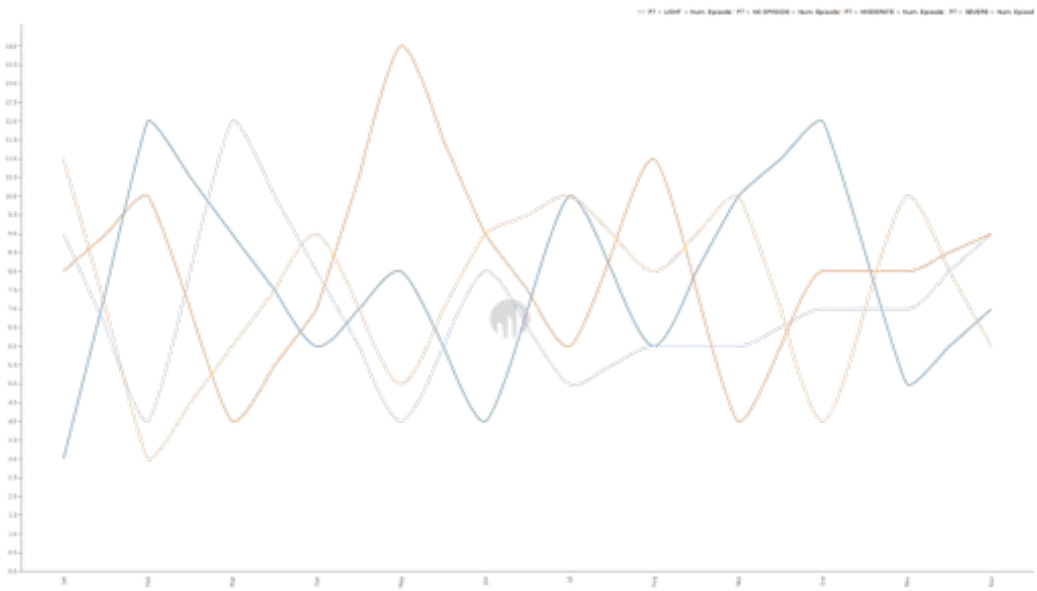
El paciente 5 es de los que mejores resultados podemos apreciar en el estudio. En su mayoría no ha sufrido episodios, siendo únicamente los episodios leves los que despuntan en el mes de febrero.
También podemos ver como sufre un episodio moderado cada dos meses, a excepción de los meses de agosto y septiembre, ya que salta directamente a octubre, donde experimenta el siguiente episodio.

Paciente 6 (P6)



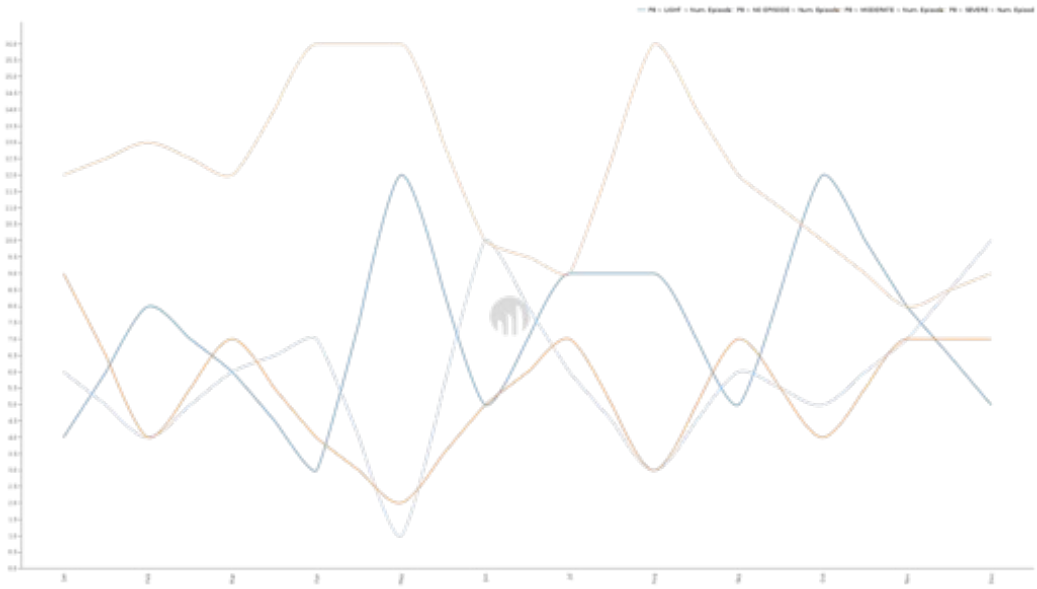
La situación del paciente 6 es algo parecida a la del paciente anterior (P5), aunque en este caso predominan más los episodios leves.
Este paciente sufre como mucho un episodio moderado al mes y aunque en verano y a final de año sufre una caída en el número de días sin sufrir un episodio, se aprecia que este número ha mejorado con respecto al inicio del estudio.

Paciente 7 (P7)



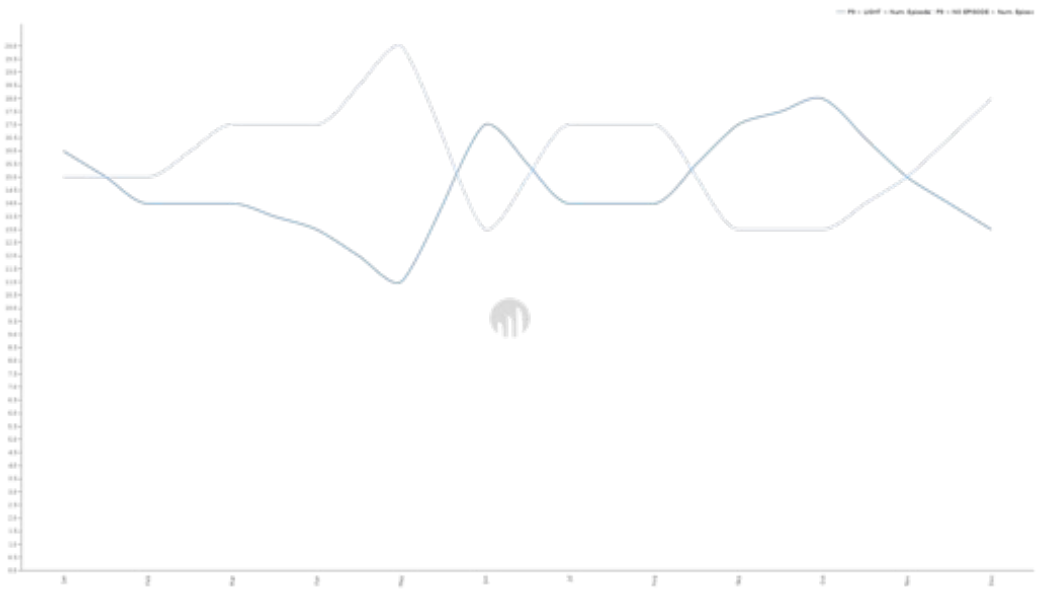
El patrón de este paciente es muy irregular a lo largo del estudio, aunque se puede apreciar al final que existe un acercamiento en el número de episodios de cada estado.

Paciente 8 (P8)



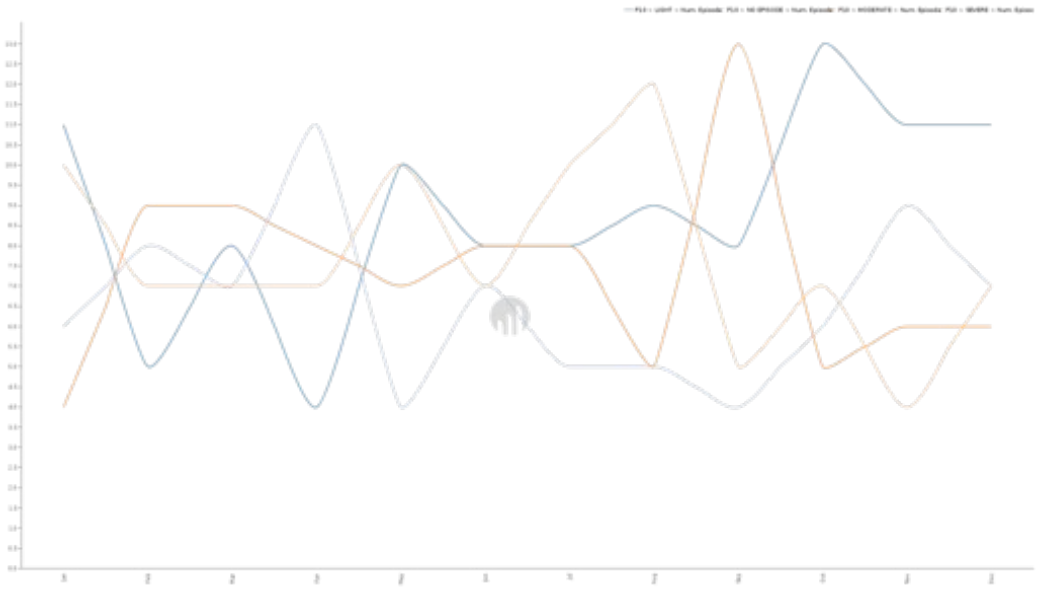
Por lo general en este paciente predominan los episodios graves y observamos una evolución muy irregular en el resto de episodios.
En el periodo vacacional de junio y julio se aprecia una mejoría en cuanto a número de episodios graves sufridos, pero este experimenta un repunte en agosto.

Paciente 9 (P9)



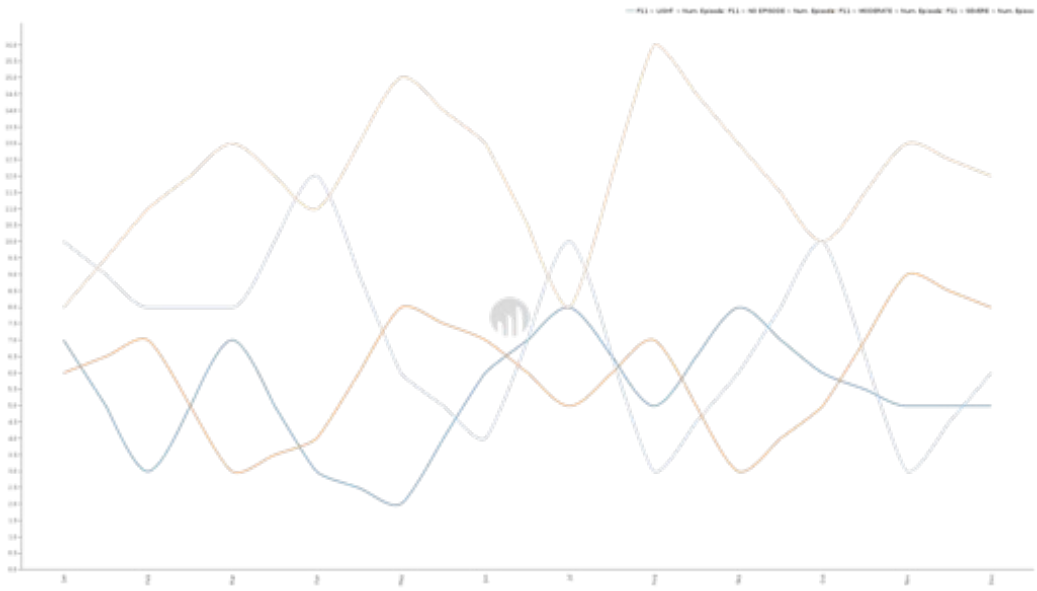
La gráfica del paciente 9 junto con el paciente 5 es una de las que mejores resultados se pueden apreciar. Sólo experimenta episodios leves o ausencia de episodios, siendo su mejor mes el mes de mayo.
Luego claramente se va alternando la predominancia de uno y otro finalizando la prueba el mes de diciembre con una mayor ausencia de episodios.

Paciente 10 (P10)



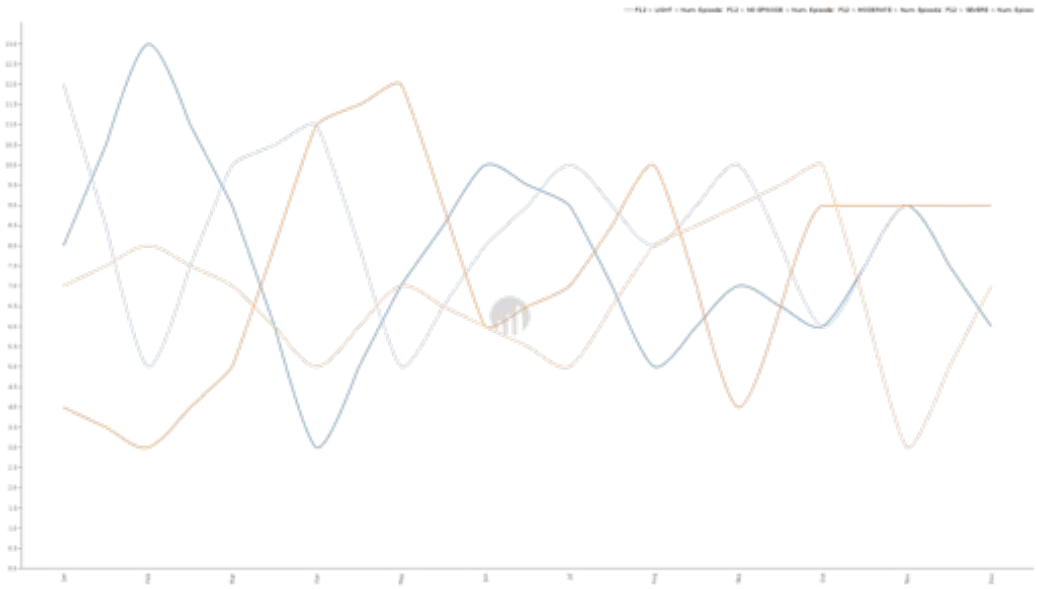
De esta gráfica podemos extraer que el paciente 10 tuvo su peor etapa los meses de agosto y septiembre, donde sufrió el mayor número de episodios graves y moderados respectivamente, pero hacia final de año el número de episodios leves supera al resto con diferencia.

Paciente 11 (P11)



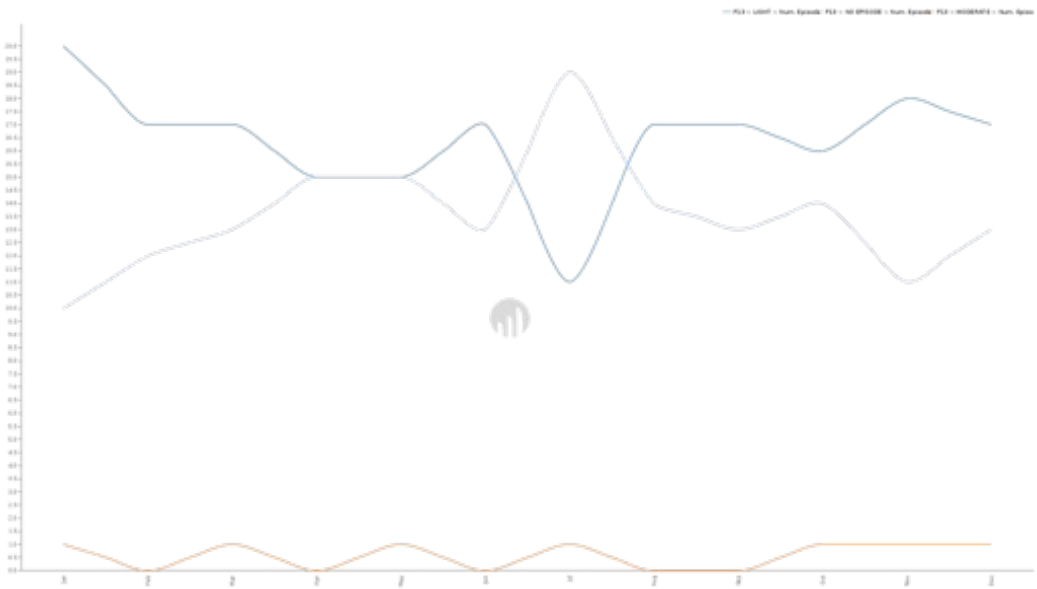
Este paciente sufre en su mayoría episodios graves durante todo el año, a excepción de los meses de enero, abril y julio.

Paciente 12 (P12)



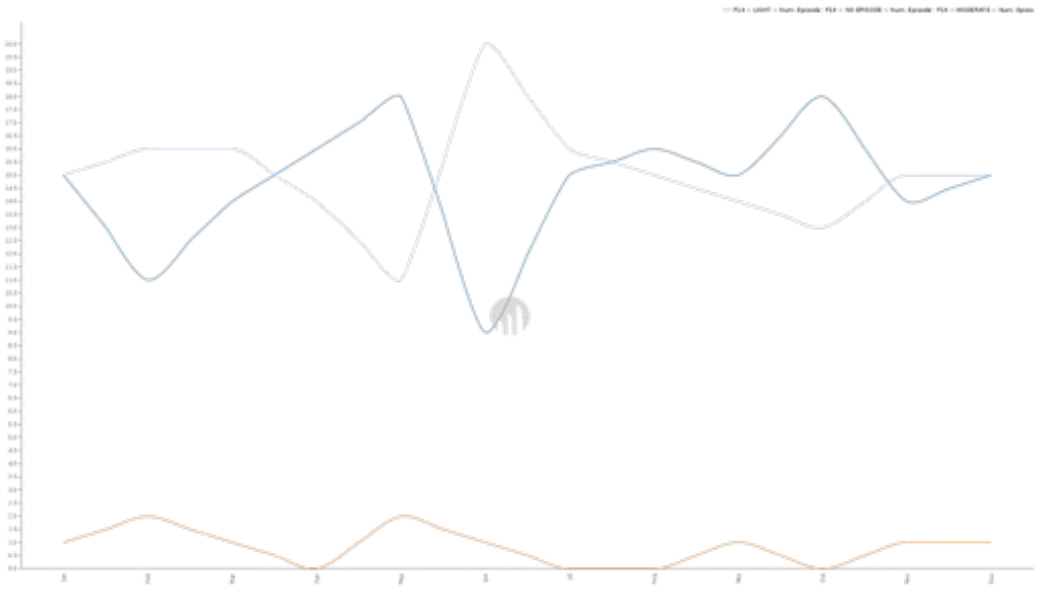
La evolución del paciente 12, tal y como podemos ver en la gráfica, es bastante irregular presentando picos de distintos tipos de episodios a lo largo del año.

Paciente 13 (P13)



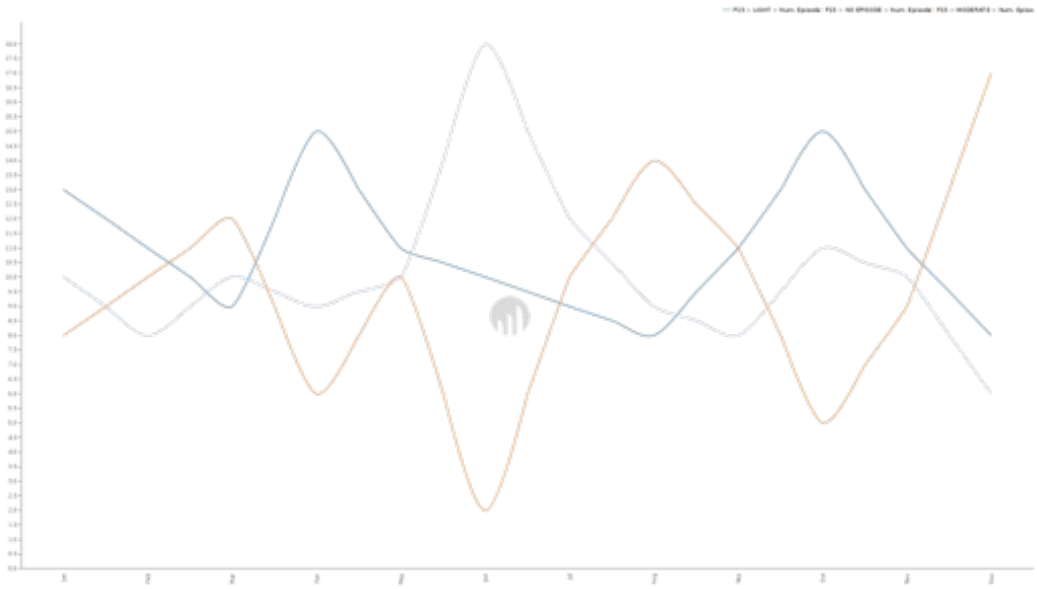
Los episodios que sufre el paciente 13, son eminentemente leves o ausencia de ellos. Destacamos el hecho de que sufre un episodio moderado cada dos meses, salvo en la recta final del año, donde los meses de octubre, noviembre y diciembre ha sufrido un episodio cada mes. El estado del paciente es bastante estable y regular a lo largo de todo el año.

Paciente 14 (P14)



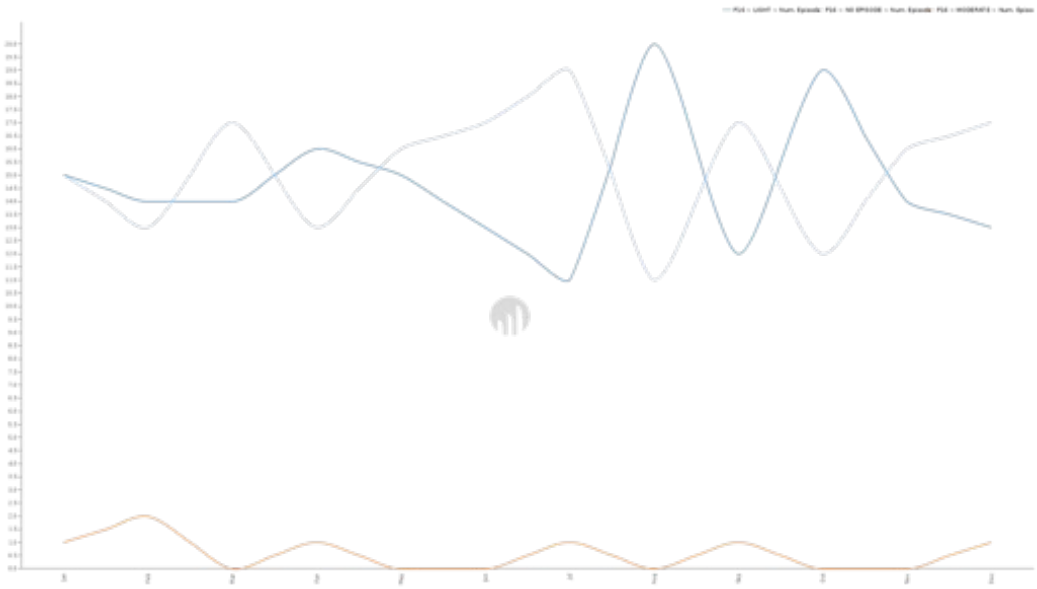
Al igual que en el caso anterior (con el paciente 13), este paciente sufre en su mayoría episodios leves, siendo los episodios moderados meramente anecdóticos, llegando como máximo a sufrir dos episodios por mes. Se aprecia que el mejor mes de este paciente fue junio, donde experimentó el mayor número de días sin episodios.

Paciente 15 (P15)



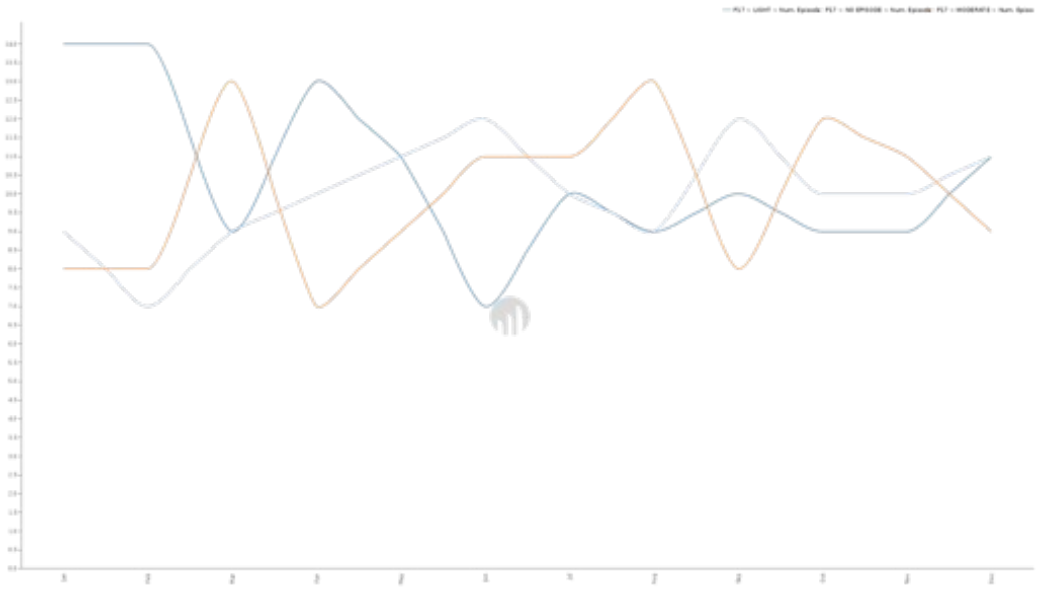
Este paciente, a pesar de no sufrir episodios graves en todo el año y de tener una evolución bastante irregular, podemos ver como la tendencia final muestra un aumento de episodios moderados frente a un descenso de los días sin episodios o episodios leves. Destacamos también que el mejor mes del paciente fue junio, donde la mayoría de días no sufrió ningún episodio.

Paciente 16 (P16)



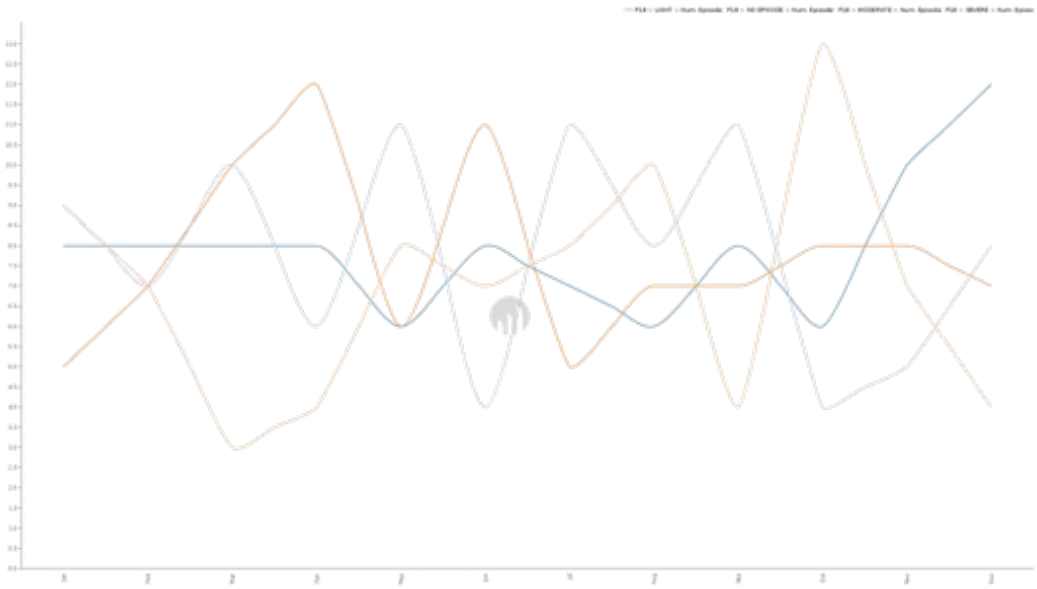
El paciente 16 muestra una gráfica muy parecida a la del paciente 14, donde podemos ver una ausencia de episodios graves y una predominancia de episodios leves o ausencia de ellos. A lo largo del año estos valores se van alternando, despuntando unos meses los episodios leves frente a la ausencia de ellos y viceversa.

Paciente 17 (P17)



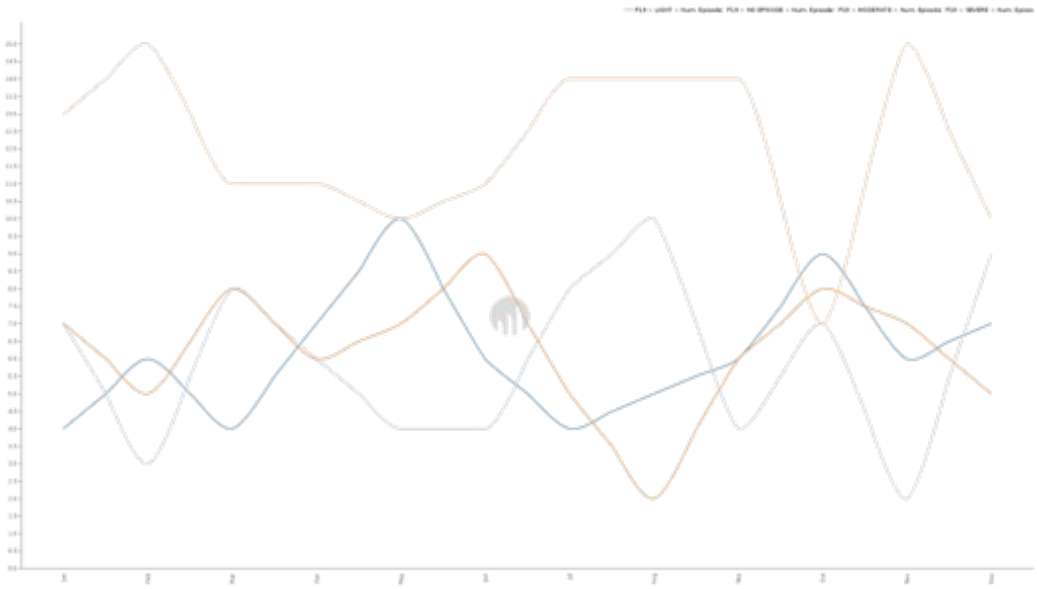
La evolución del paciente 17, a pesar de no sufrir ningún episodio grave a lo largo de todo el estudio, es bastante dispar, alternándose los meses que sufre episodios leves, con los moderados o ausencia de los mismo. Por lo que no se puede establecer un patrón claro de evolución.

Paciente 18 (P18)



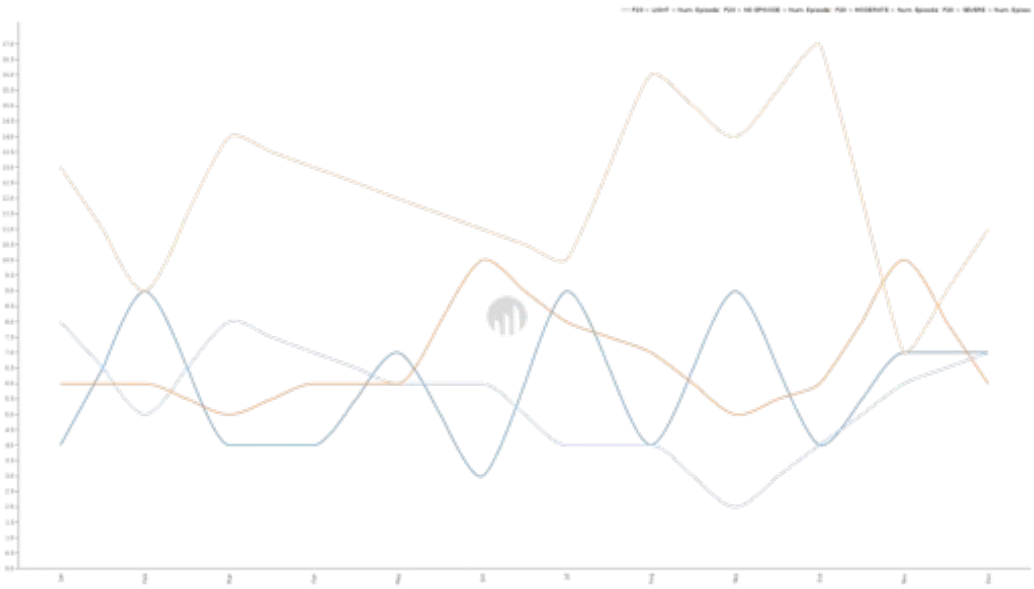
Este paciente presenta grandes cambios cada mes, despuntando en algunos casos los episodios moderados o la ausencia de ellos. Cabe destacar el mes de octubre, donde sufrió el mayor número de episodios graves, así como la tendencia a experimentar más episodios leves frente al resto al finalizar el año.

Paciente 19 (P19)



Los episodios graves predominan en la evolución del paciente 19, siendo únicamente superados el mes de octubre. El resto de episodios se alternan mes a mes aunque nunca superando a los episodios graves (salvo el mes de octubre).

Paciente 20 (P20)

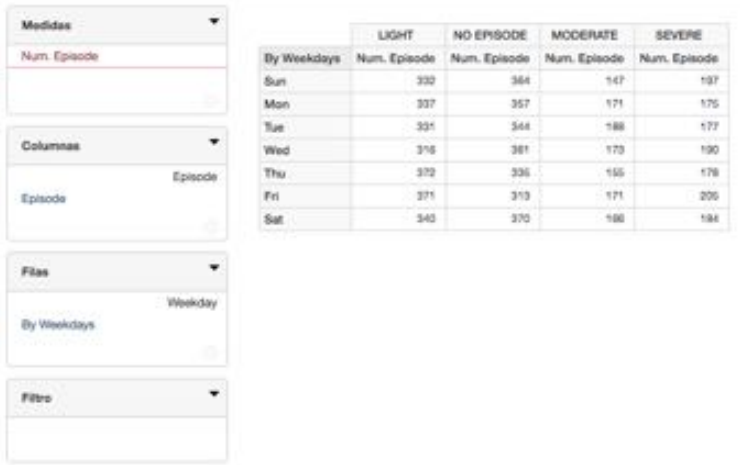


La lectura de esta gráfica es muy parecida a la del paciente anterior, donde podemos ver una mayoría clara de episodios graves sufridos a lo largo del año, teniendo sus picos máximos los meses de agosto y octubre, siendo únicamente superados por el número de episodios moderados el mes de noviembre.

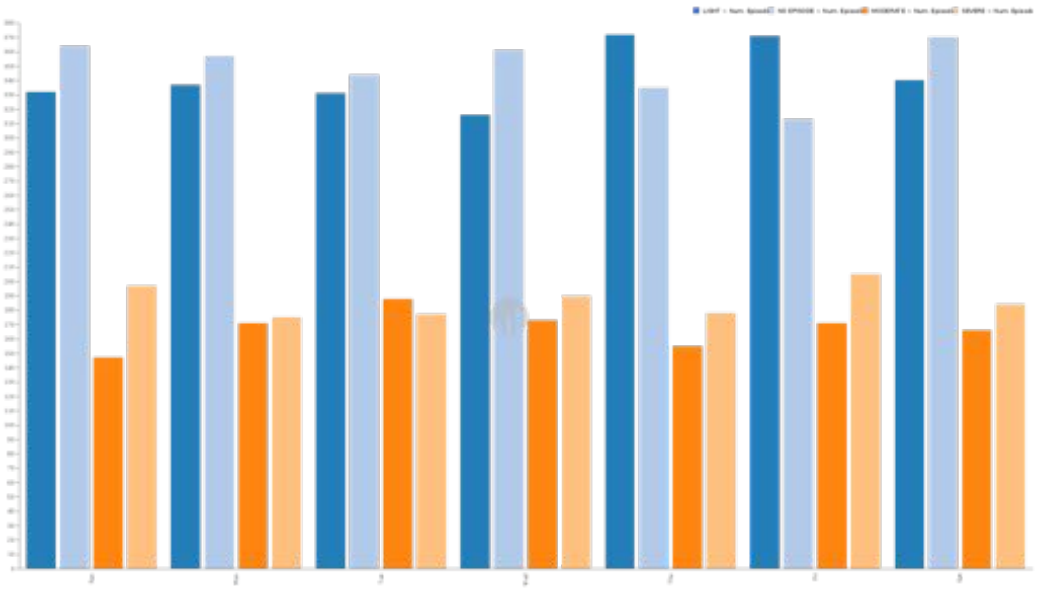
Pregunta 6

¿Se puede establecer alguna relación entre los episodios de crisis y el momento del día o de la semana o del año?

Debido a que la granularidad del dato obtenido desde el fichero fuente, sonde pruebas por día, no podemos sacar conclusiones a nivel de distintos momentos del día.
Sí hemos dispuesto la información de forma que podamos consultar los datos a nivel de días de la semana, así como a nivel de mes. Igualmente se puede consultar por día del año, que es el dato base.

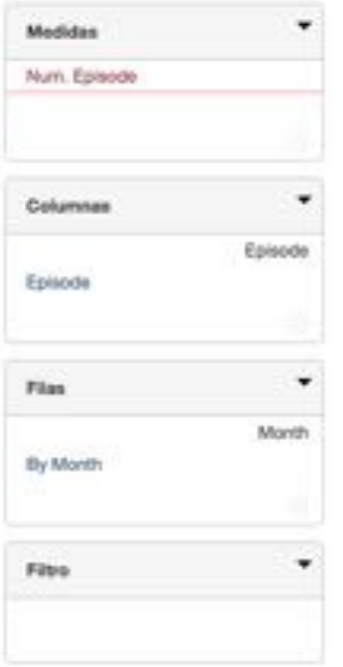


	LIGHT	NO EPISODE	MODERATE	SEVERE
By Weekdays	Num. Episode	Num. Episode	Num. Episode	Num. Episode
Sun	332	364	147	197
Mon	337	357	171	170
Tue	331	344	188	177
Wed	316	361	173	190
Thu	372	336	155	179
Fri	371	313	171	200
Sat	340	370	166	184

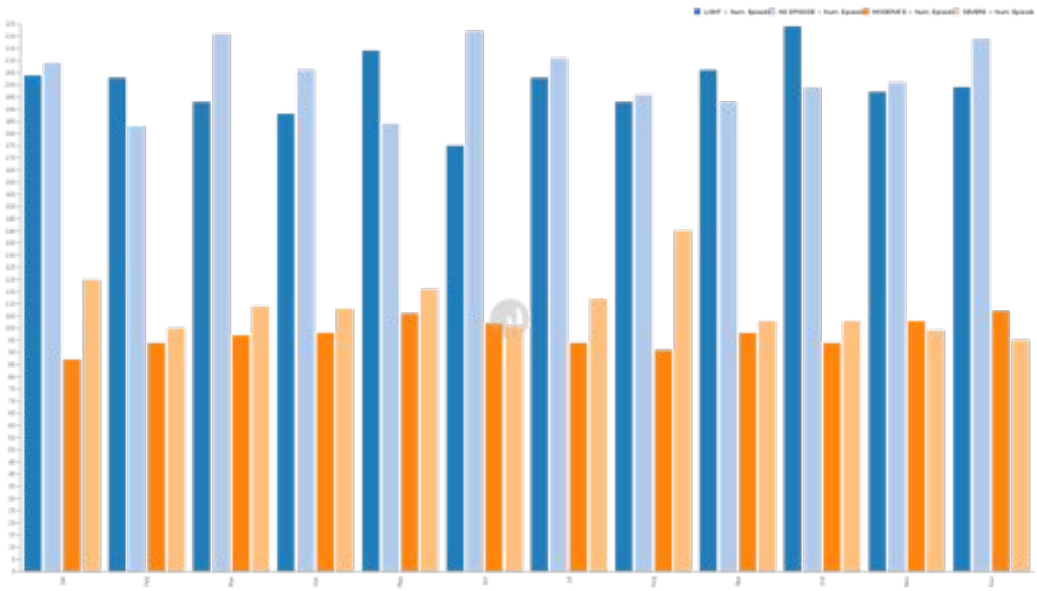


Según la consulta anterior, podemos ver como predominan los episodios leves o ausencia de episodios durante toda la semana. Por lo general, superando la ausencia de episodios a excepción de los jueves y viernes. En menor medida tenemos un mayor número de episodios graves frente a moderados durante toda la semana a excepción de los martes, aunque las diferencias son muy leves.

Si organizamos la información por meses, obtenemos los siguientes resultados.

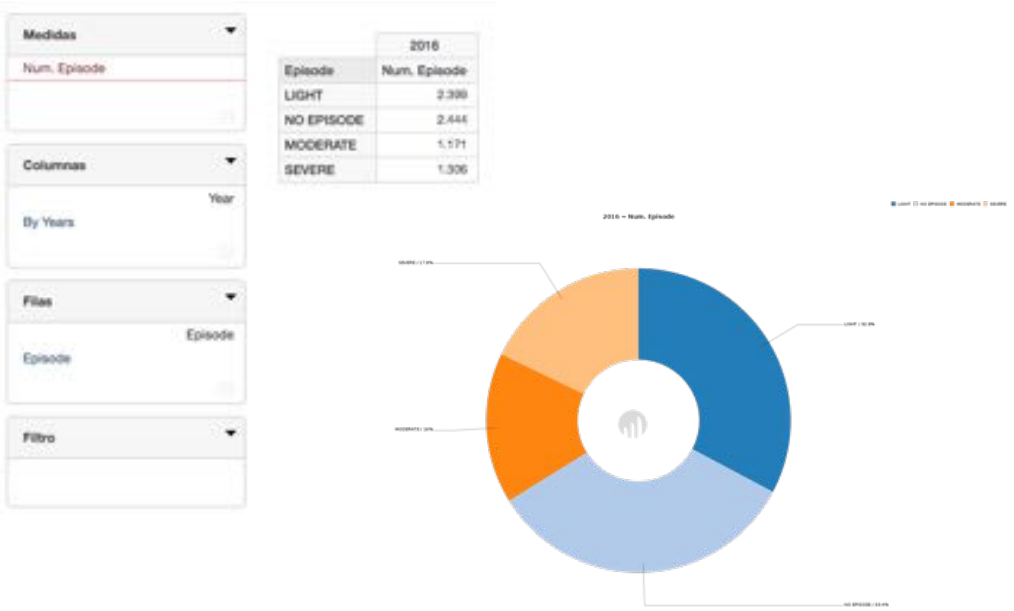


	LIGHT	NO EPISODE	MODERATE	SEVERE
By Month	Num. Episode	Num. Episode	Num. Episode	Num. Episode
Jan	204	209	87	120
Feb	203	183	94	100
Mar	193	221	97	109
Apr	188	206	96	106
May	214	184	106	116
Jun	175	222	102	101
Jul	203	211	94	112
Aug	193	196	91	140
Sep	206	193	98	103
Oct	224	199	94	103
Nov	197	201	103	99
Dec	199	219	107	96



Podemos ver igualmente que el mayor número de días se corresponden a días con episodios leves o sin episodios. Cabe destacar los meses de marzo, junio y diciembre por su mayor número de días sin episodios. Igualmente llama la atención el mes de agosto, donde podemos ver un incremento significativo de episodios graves.

Ahora, si consultamos los datos a nivel anual, obtenemos la siguiente tabla y gráfica:



Podemos apreciar como el mayor número de días se reparten entre episodios leves o ausencia de episodios con un porcentaje muy similar. Por otro lado, el resto también se reparte de forma equitativa entre episodios moderados y graves.

Pregunta 7

La realización de actividades físicas mejora o empeora el estado de ánimo de los pacientes.

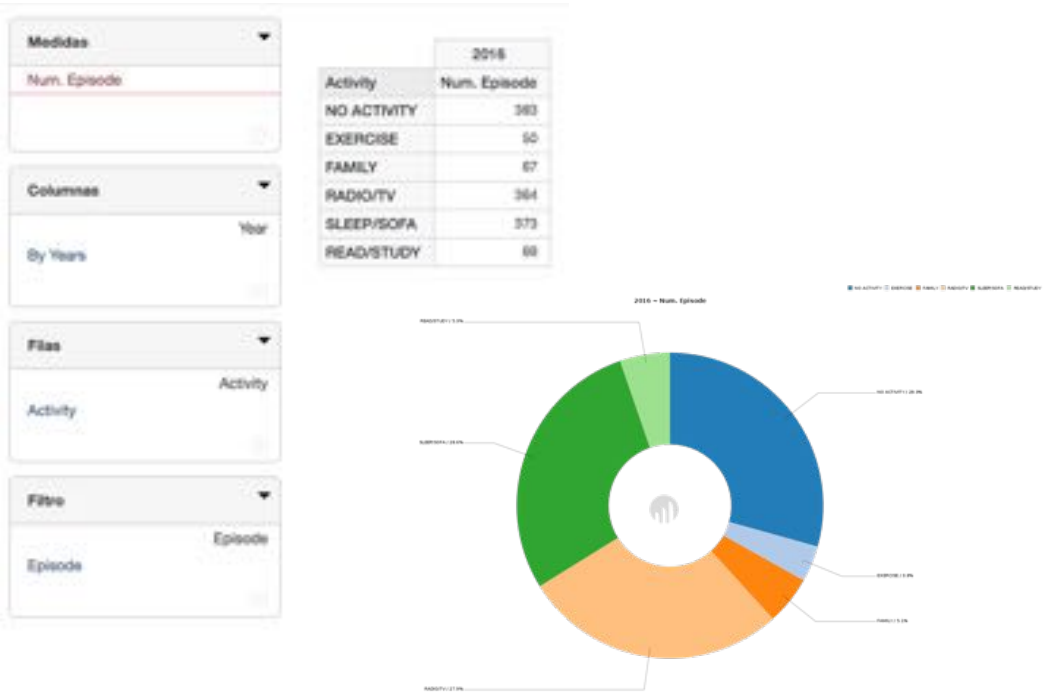
No se puede responder a esta pregunta, pues no se dispone de información relativa a los estados de ánimo de los pacientes.

Pregunta 8

¿Hay algún tipo de actividad que mejore el día a día de los pacientes?

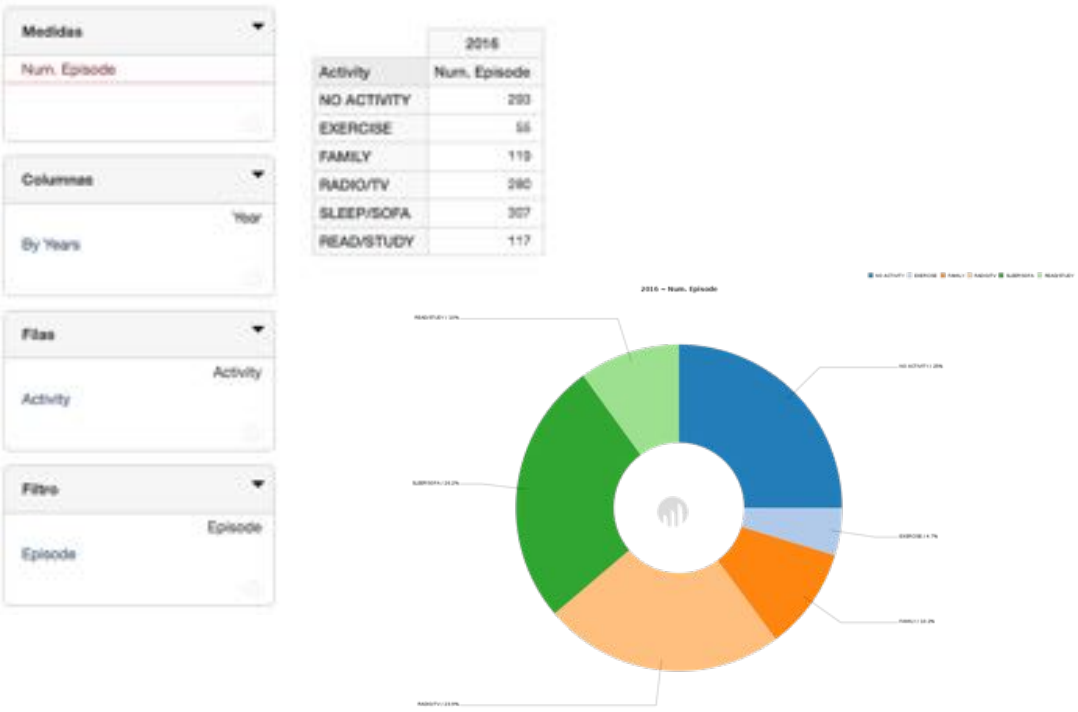
Podemos evaluar de forma anual el número del tipo de episodios que han sufrido los pacientes en relación a la actividad realizada con carácter anual. Para ello vamos a filtrar en cada consulta por un tipo de episodio.

GRAVE



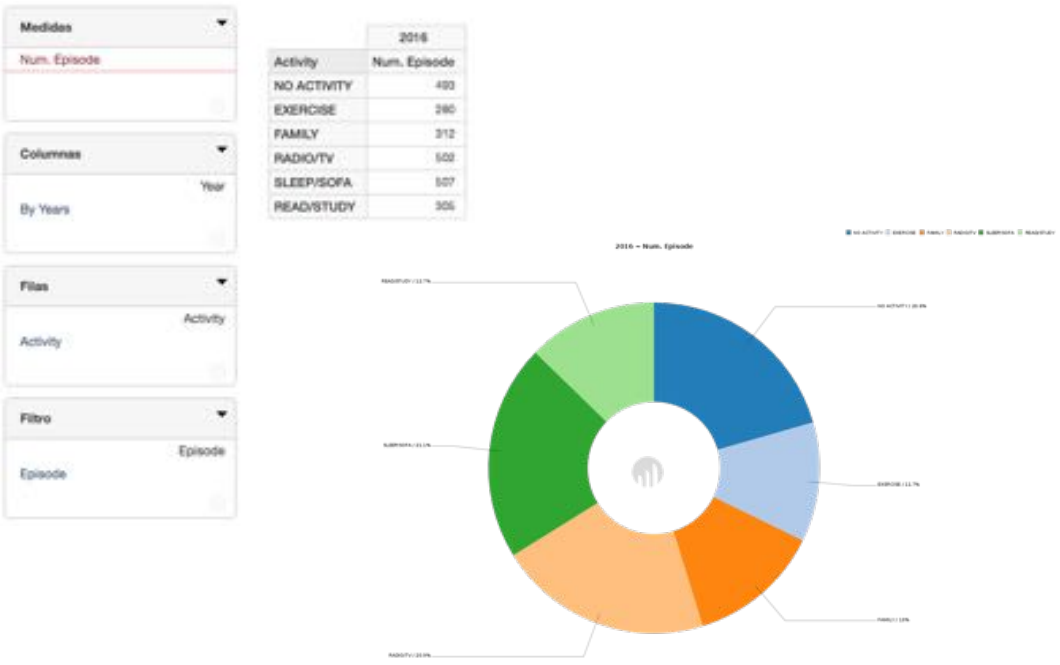
Se puede comprobar como para las actividades que menos esfuerzo se requieren ("SLEEP/SOFA" - "RADIO/TV" - "NO ACTIVITY") ocasionan el mayor porcentaje de episodios graves.

MODERADO



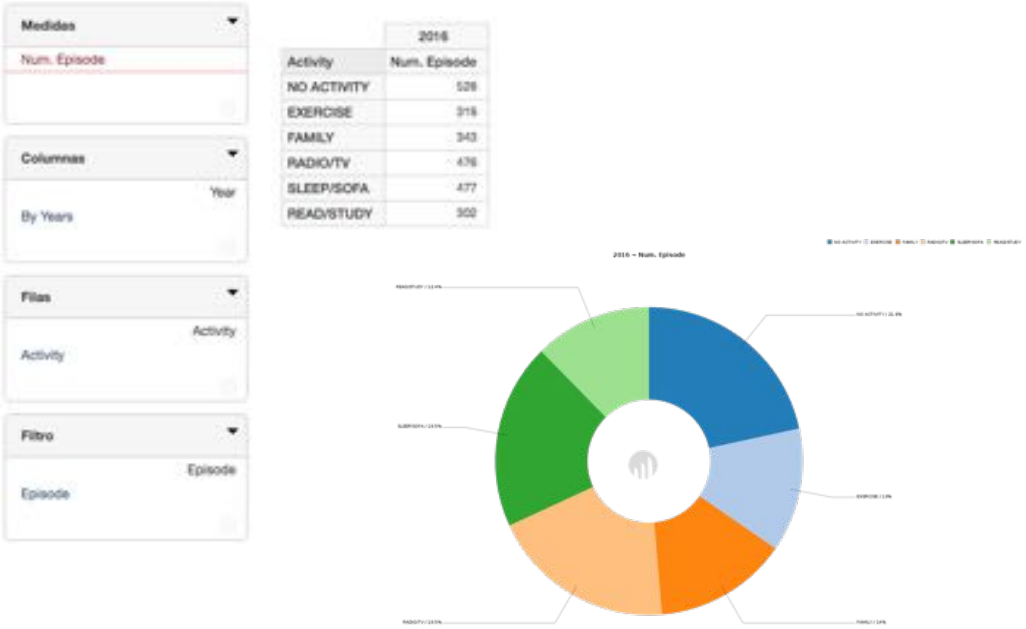
Al igual que en los casos de episodios graves, en los episodios moderados podemos comprobar como las actividades que menos esfuerzo por parte del paciente requieren, ocasionan el mayor número de episodios moderados.

LEVE



Para el caso de los episodios leves, observamos como la diferencia entre la relación del tipo de actividad y sus consecuencias para generar episodios disminuye, no siendo tan clara como en los casos anteriores. Aunque sí podemos decir que sigue existiendo un porcentaje mayor de sufrir episodios leves si se realizan actividades de poco esfuerzo ("SLEEP/SOFA" - "RADIO/TV" - "NO ACTIVITY").

SIN EPISODIO



Al igual que en el caso anterior para los episodios leves, para los casos de no sufrir ningún tipo de episodio no se puede establecer una relación directa con las actividades, pues no existen grandes diferencias. Siguen acaparando los mayores porcentajes las actividades que menos esfuerzo físico requieren, aunque no por mucha distancia.

Podemos concluir que las actividades físicas ("EXERCISE") o mentales ("READ/STUDY" - "FAMILY") pueden ayudar a disminuir significativamente el número de episodios graves y moderados, pero para los casos de episodios leves esta mejora no será tan acuciante.
Por otro lado y contradiciendo el efecto que supone para los casos graves y moderados, el desempeño de este tipo de actividades no supondrá un aumento del número de días en los que un paciente no sufre ningún tipo de episodio.

3. Conclusiones

Una vez finalizado el trabajo podemos sacar una serie de conclusiones tanto a nivel funcional, como a nivel técnico en la implementación del sistema de análisis basado en Pentaho.

3.1. Apartado técnico

La suite "community" de Pentaho ofrece las herramientas básicas necesarias para poder desplegar un sistema completo de BI, esto es, desarrollo de los procesos ETL, diseño de cubos OLAP y sistema de análisis de datos. Hemos podido observar como el correcto diseño de este sistema nos ofrece una potencia de consulta y versatilidad muy elevada, de forma que podemos "jugar" con los elementos para poder sacar respuestas. Suponemos que fuera de la versión gratuita de Pentaho seguramente podremos tener un amplio abanico de posibilidades en este área, pero la limitación económica que planteamos desde un comienzo nos hizo decantarnos por la plataforma más versátil y completa a coste cero.
Una vez comprendido todas las fases de implementación detalladas en este trabajo podemos enfrentarnos desde una perspectiva de principiante a cualquier proyecto de Business Intelligence.

3.2. Apartado funcional

Si planteamos las conclusiones en relación a los resultados obtenidos para dar respuesta a las preguntas planteadas, podemos decir de forma resumida que aquellas actividades que menor esfuerzo físico o mental requieren son las que más episodios graves o moderados producen. En otra línea también destacamos el beneficio que produce vivir en entornos rurales, pues son los que menos episodios graves registran.

3.3. Gestión del proyecto

El logro de los objetivos ha sido alcanzado con éxito, si bien es cierto que conforme hemos ido avanzando en el diseño del sistema hemos adquirido conocimientos que nos han hecho plantearnos posibles alternativas más óptimas. Por ejemplo, conforme finalizábamos la etapa de ETL, los recursos y conocimientos que disponíamos al final nos hubiesen permitido diseñar procesos mas "elegantes" y limpios.

Haciendo referencia al seguimiento de la planificación no podemos mencionar nada destacable, la consecución de las etapas y distintos hitos se ha podido llevar con éxito y en algunos casos poder adelantar partes del trabajo. Quizá uno de los elementos más críticos y de difícil ponderación fue el tiempo reservado para la adquisición de conocimientos, tiempo que dedicamos a la lectura del libro recomendado en el inicio del proyecto "*The Data Warehouse Lifecycle Toolkit - Ralph Kimball*". Finalmente y con el objetivo de poder concluir los trabajos en fecha, decidimos ser muy estrictos a la hora de seleccionar aquellos

capítulos más interesantes de cara a la elaboración de las distintas etapas.

3.4. Líneas de trabajo futuro

Como posibles líneas de trabajo futuro, podríamos plantear la elaboración de procesos ETL resistentes a actualizaciones (dimensiones que se actualizan, o nuevos datos de entrada para la tabla de hechos), así como el desarrollo de un Dashboard *ad-hoc* que permita a usuarios no familiarizados con plataformas de BI poder consultar datos relativos a su negocio.

Hay que tener en cuenta también que en el caso que nos ocupa (medicina), tenemos datos de pacientes que deben ser tratados con la mayor confidencialidad y por tanto deben ser anónimos. Debido a que el fichero fuente ya estaba de base sin nombres, no ha sido necesario plantear un proceso de anonimización, pero es una fase que deberíamos contemplar en este área.

4. Glosario

BI:	Business Intelligence
DWH:	Data Warehouse
BBDD:	Bases de datos
ETL:	Extract Transform Load (Extracción transformación y carga)
3FN:	Tercera Forma Normal
PK:	Primary Key (Clave primaria)
FK:	Foreign Key (Clave foránea)
XML:	eXtensible Markup Language
OLAP:	On-Line Analytical Processing

5. Bibliografía

[1] Ralph Kimball, Margy Ross, Warren Thornthwaite, Joy Mundy and Bob Becker, "The Data Warehouse Lifecycle Toolkit" Second Edition, Wiley, Indianapolis, 2008

[2] http://global.qlik.com/au/blog/posts/james-fisher/enter-the-era-of-modern-bi-platforms, 03/2017

[3] http://www.informationweek.com/big-data/big-data-analytics/gartner-bi-magic-quadrant-2015-spots-market-turmoil/d/d-id/1319214, 03/2017

[4] http://community.pentaho.com/ , 03/2017

[5] https://www.virtualbox.org/, 04/2017

[6] https://www.ubuntu.com/server, 04/2017

[7] https://mariadb.org/, 04/2017

[8] https://www.mysql.com/products/workbench/, 04/2017

[9] http://meteorite.bi/products/saiku, 05/2017

6. Anexos

6.1 KETTLE: Información general

Tal y como hemos adelantado, vamos a hacer uso de la herramienta Kettle que nos proporciona la suite Community de Pentaho, también conocida como PDI (Pentaho Data Integration).
Esta herramienta está desarrollada en Java, por lo que deberemos tener instalado el JDK en nuestra máquina de trabajo.

En mi caso estoy trabajando con la versión 8:

```
$ java -version
java version "1.8.0_121"
Java(TM) SE Runtime Environment (build 1.8.0_121-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.121-b13, mixed mode)
```

Su ejecución depende del entorno, y en mi caso he de destacar que debo lanzarlo desde el terminal:

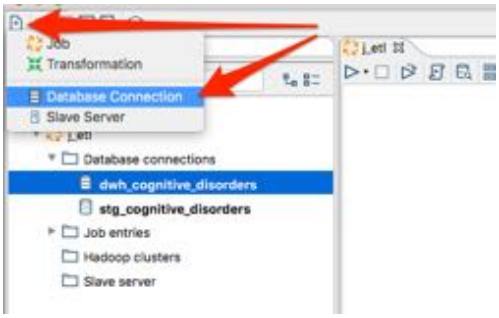
```
$ /Applications/Pentaho/data-integration/spoon.sh
```

Para la correcta ejecución de las siguientes tareas dentro de esta herramienta, es fundamental conocer los siguientes dos puntos:

- Como establecer una conexión con una BBDD
- Como cargar la información desde un Excel

Para el **primer punto**, aunque a priori pueda parecer trivial, creemos interesante mencionar el proceso de creación de esta conexión, ya que si lo hacemos de una forma correcta nos ahorraremos tiempo en las siguientes operaciones.

En el icono que podemos encontrarnos en la parte superior izquierda, tenemos la posibilidad de crear una nueva conexión a base de datos.



Esta opción nos abrirá una pantalla de edición de conexión como la que mostramos a continuación:

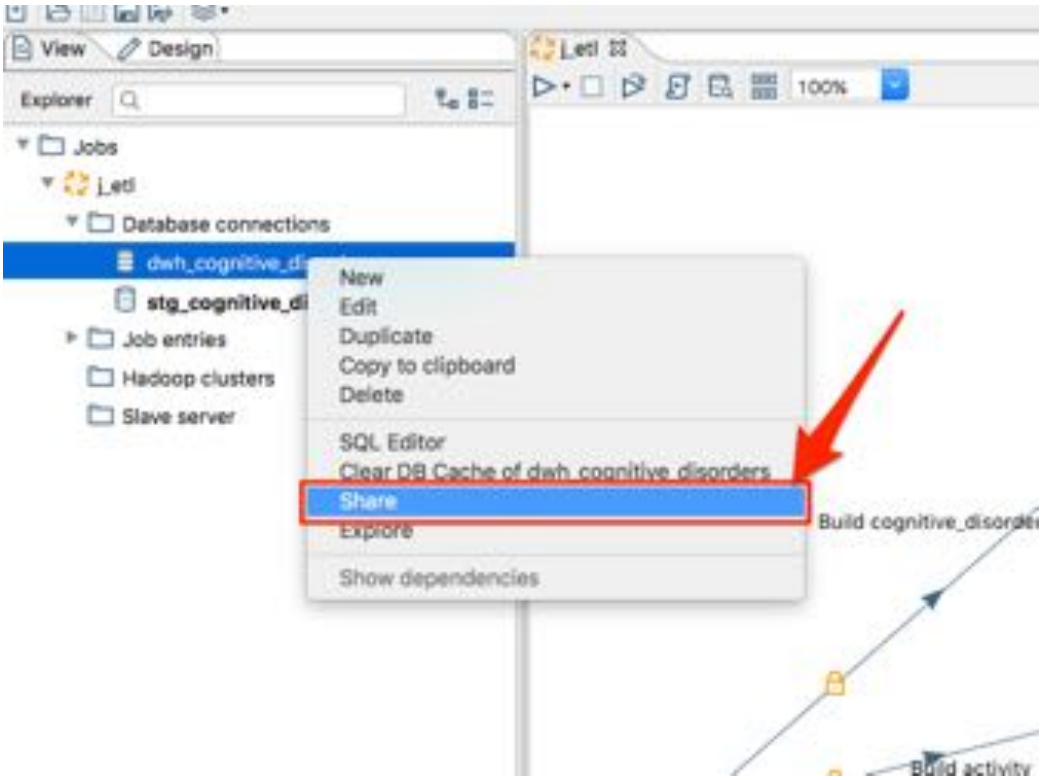
En ella podremos introducir los datos, que en nuestro caso y tal y como hemos preparado nuestro servidor y base de datos son:

Conection Name: dwh_cognitive_disorders
Conection Type: MySQL
Access: Native (JDBC)

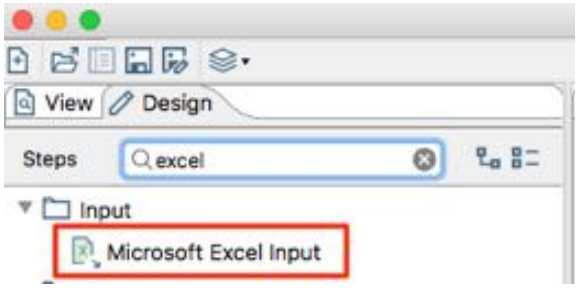
Host name: 192.168.56.10
Database Name: dwh_cognitive_disorders
Port Number: 3306

User Name: bimanager
Password: ********** (no la mostramos por seguridad)

Una vez creada la conexión, si queremos evitar tener que definirla una y otra vez por cada trabajo que creemos en Kettle, debemos activar la opción de compartir. Esta opción se puede encontrar haciendo "click" derecho sobre la nueva conexión creada y pulsando en la opción de "share".

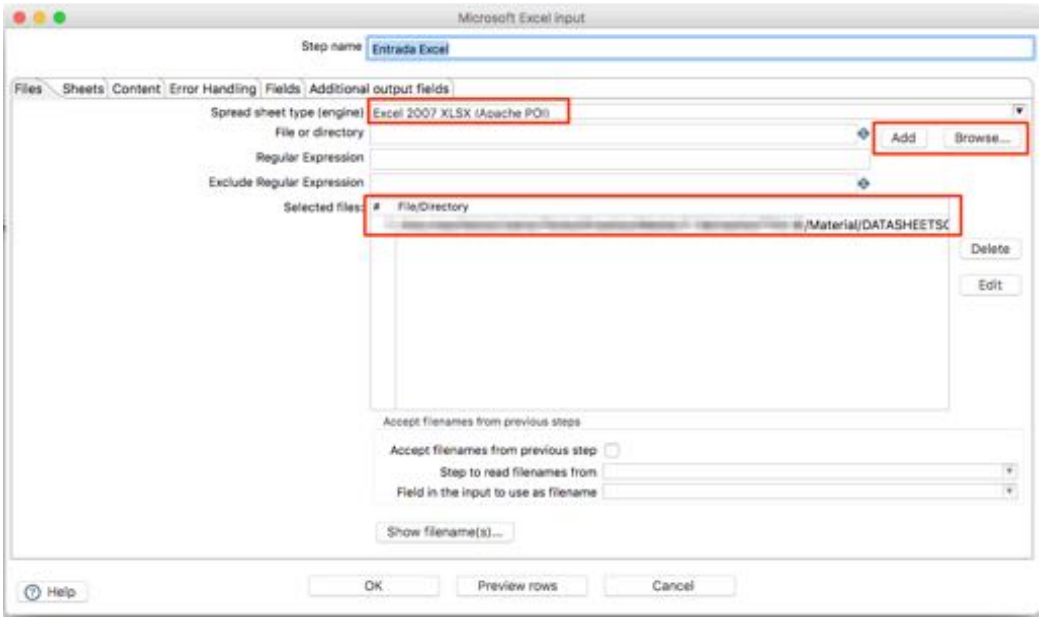


Para la obtención de la información de nuestra fuente Excel, haremos uso de la opción "Microsoft Excel Input" que Pentaho nos proporciona para facilitarnos el trabajo.

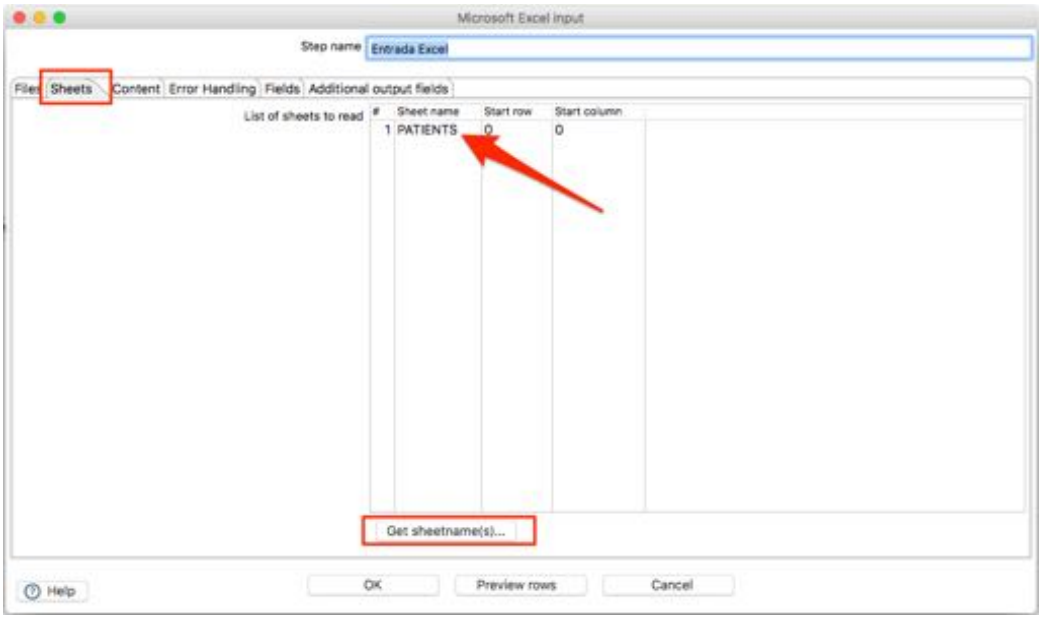


En todas aquellas fases que definamos y que tengamos que recurrir a nuestra fuente Excel, el proceso será muy similar al que definimos a continuación.

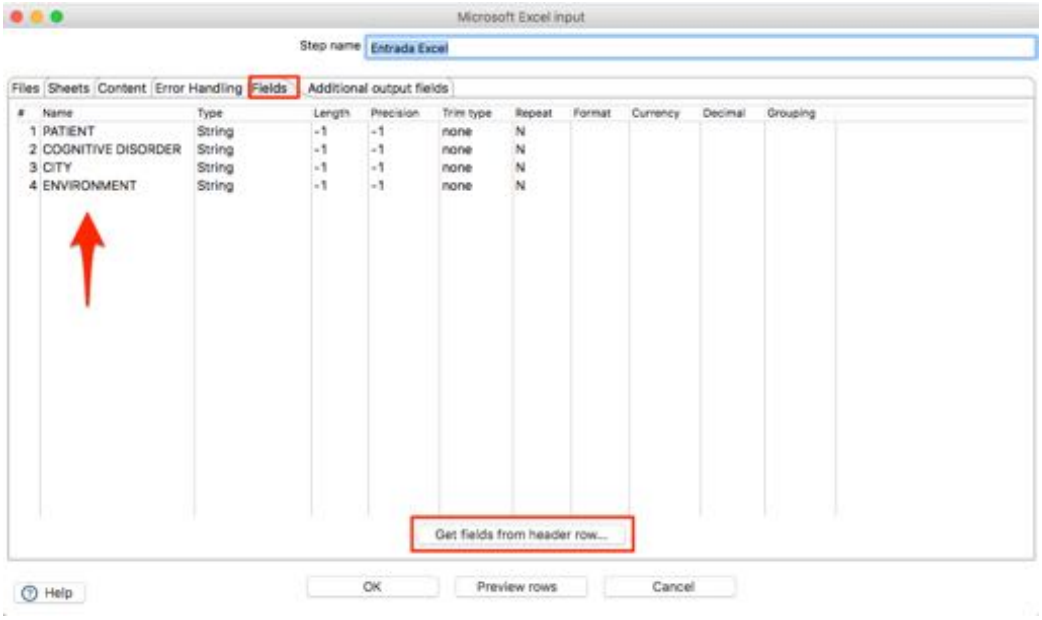
Primero de todo debemos indicar a la herramienta qué tipo de hoja de cálculo es: "Excel 2007 XLSX (Apache POI)"
Luego indicaremos la ruta donde se encuentra nuestro archivo:



Los pasos siguientes dependerán de la información que queramos obtener del fichero. Para el ejemplo que estamos mostrando, donde obtenemos la información de la primera pestaña (PATIENTS), bastará con seleccionar la pestaña correspondiente en la sección "Sheets".



Y por último indicar cuales son los campos que queremos obtener en la sección "Fields".



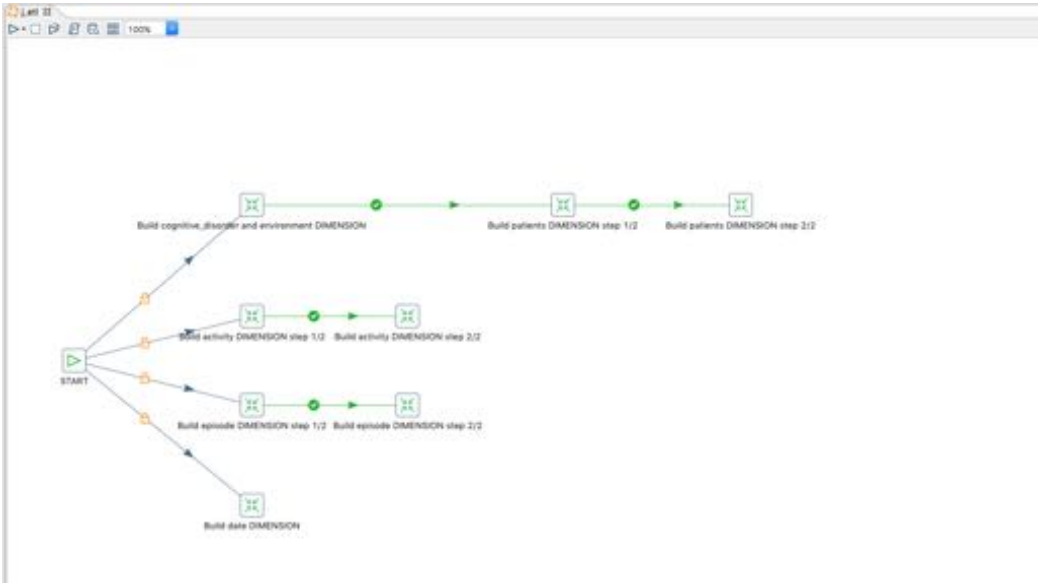
6.2 KETTLE: Construcción de las dimensiones

Para la carga de nuestro DWH, vamos a separar los procesos en dos bloques. Un primer bloque donde se va a construir la información relevante a todas las dimensiones y un segundo bloque donde se realizará a carga de los hechos (medidas).

En este primer bloque de carga de dimensiones, crearemos un trabajo (job) que llamará a cada una de las transformaciones.

Aunque el proceso de creación se ha realizado definiendo primero las transformaciones para luego irlas concatenando dentro del trabajo principal, con el objetivo de ofrecer una visión general de esta fase, vamos a definir primero la capa más externa e iremos adentrándonos en aquellos puntos donde tenga especial interés su mención.

En la siguiente captura podemos ver el esquema de nuestro primer trabajo (job) donde ejecutaremos todas las transformaciones necesarias para completar las dimensiones en nuestro DWH.



Para ello hemos creado un proceso de inicio que ejecutará cuatro ramas (de arriba a abajo):

1. Construirá las dimensiones "cognitive_disorder", "environment" y "patient".
2. Construirá la dimensión "activity".
3. Construirá la dimensión "episode".
4. Construirá la dimensión "date".

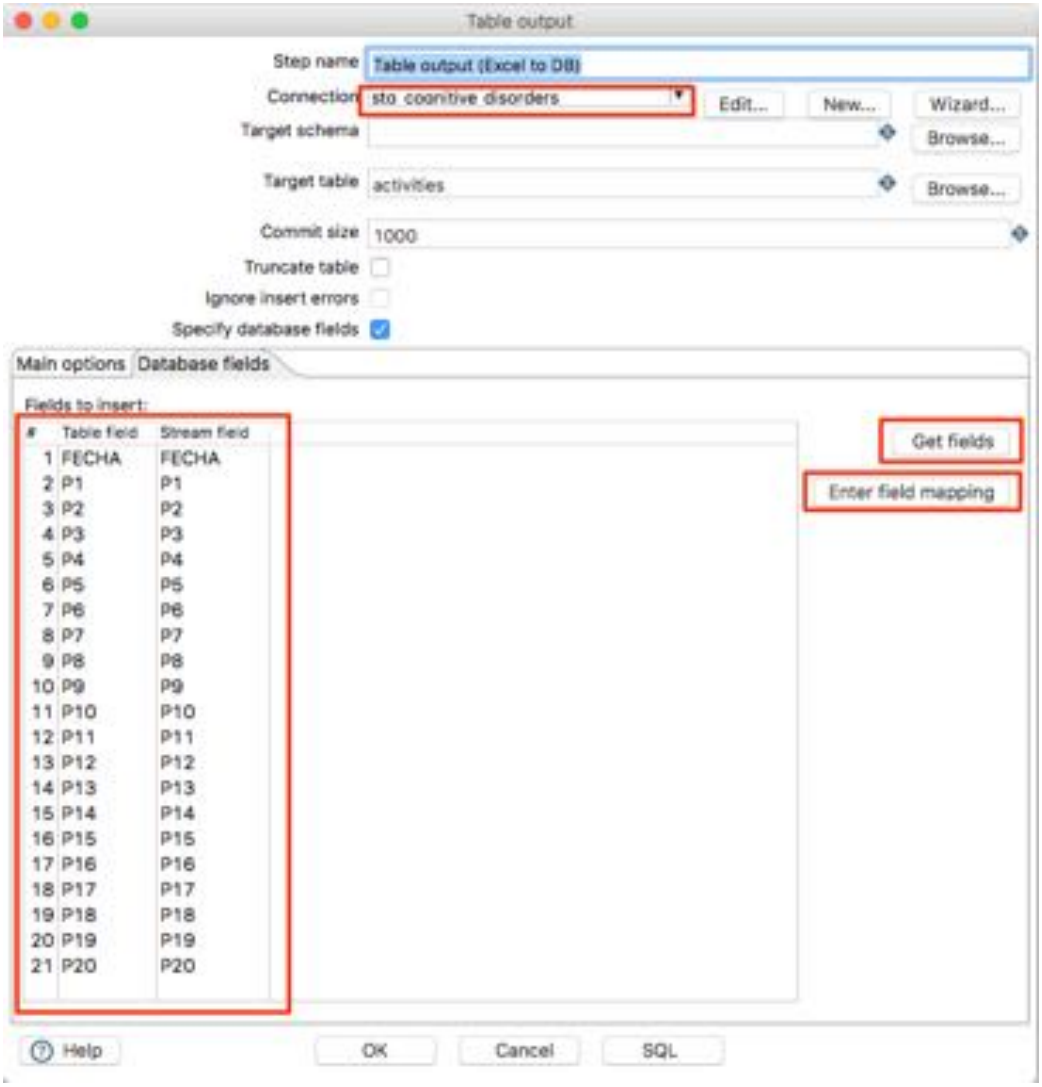
En le primer caso, hemos juntado esas tres dimensiones en un sólo proceso, ya que la elaboración de la dimensión "patient" depende de haber creado previamente las dimensiones "environment" y "cognitive_disorder", recordemos observando el modelo dimensional la relación entre estas tablas.

Otro detalle que podemos destacar, es la necesidad de crear varias transformaciones para la carga de una dimensión. Este es el caso de las dimensiones "activity", "episode" y "patient". Nos surge entonces la necesidad de utilizar una segunda BBDD que denominaremos de "staging" donde almacenaremos información proveniente del Excel y que podremos alterar hasta obtener los resultados deseados para cargar en nuestro DWH.

Como ejemplo de este proceso podemos mostrar el caso de la creación de la dimensión "activity".

En una primera instancia definiremos una transformación (t_d_activities.ktr) donde deberemos obtener los datos del excel correspondientes a la pestaña "ACTIVITY VALUES" de la misma forma

que comentamos anteriormente, para finalmente almacenarlos en una tabla de nuestra BBDD de "staging" tal y como mostramos a continuación.



En la siguiente transformación, haremos uso de esta nueva tabla de la base de datos de "staging" para obtener el conjunto completo de datos que necesitamos y finalmente lo almacenaremos en su tabla dimensional.

En el ejemplo mostrado se puede apreciar como mediante una consulta SQL obtenemos y transformamos la información que queremos de la tabla origen para almacenarla finalmente en nuestro DWH.

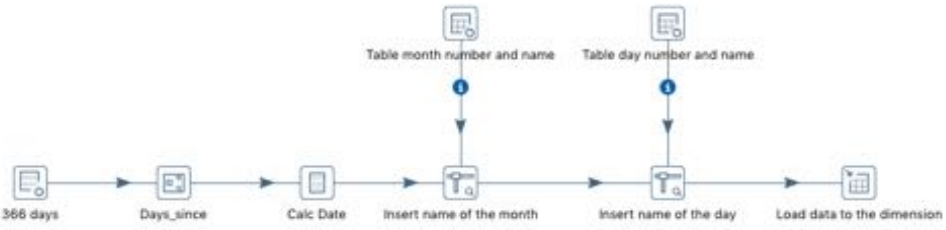
Este procedimiento se puede ver repetido de forma exacta en la carga de la dimensión "episode" y de forma ampliada en la dimensión "patient" ya que en esta última entran en juego más elementos, pues hay que relacionar 3 tablas.

Quizá la dimensión con mayor complejidad de procesado es la dimensión "date", pero debido a que es muy común en todos los proyectos de BI, nos es posible encontrar un ejemplo en las fuentes de recursos de la aplicación Kettle.

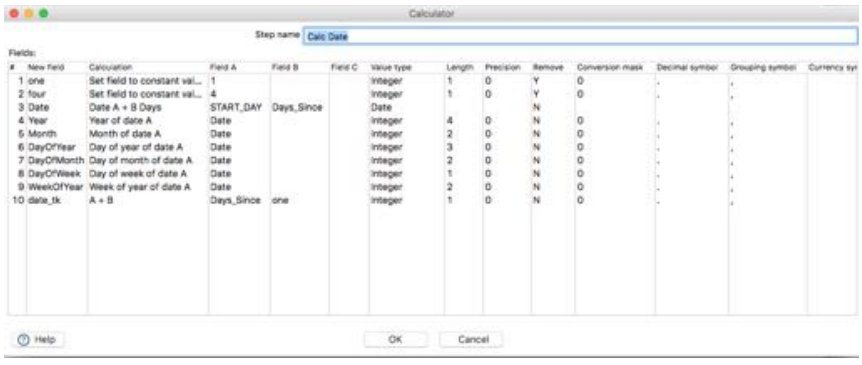
```
/Applications/Pentaho/data-integration/samples/transformations
```

Concretamente hemos hecho uso del archivo: "**General - Populate date dimension.ktr**", adaptándolo a las necesidades de nuestro sistema.

El resultado final es el siguiente:



Este proceso genera una muestra de 366 días, donde se calcula mediante la herramienta "calculator" cada uno de los valores de cada entrada.



Por último y mediante la opción "Stream Value Lookup", añadimos a nuestra tabla tanto los nombres de los meses como los nombres de los días.

6.3 KETTLE: Construcción de los hechos

Pasamos a definir las características más importantes del proceso de carga de hechos (facts) en nuestro DWH.

Al igual que realizamos con las dimensiones, vamos a crear un trabajo (job) que ejecutará las siguientes transformaciones:

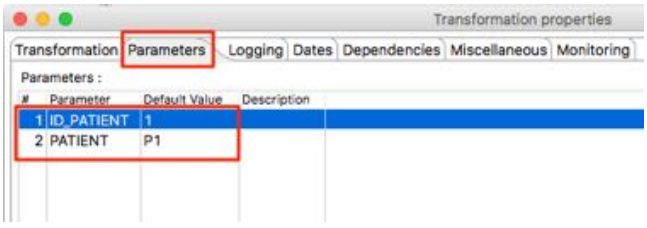
1. **Transformación 1**: Carga de los datos relativos a las horas de sueño en la BBDD de "staging", que luego nos servirá para almacenarlos en la última transformación.
2. **Transformación 2**: Obtención del par clave primaria del paciente y su nombre (esta información la obtendremos directamente de la tabla de dimensiones que hemos creado en el trabajo anterior)
3. **Transformación 3**: Carga de hechos por cada pacientes

Como podemos observar, esta vez hemos añadido un proceso en el segundo paso que nos permite parametrizar las ejecuciones de una transformación.

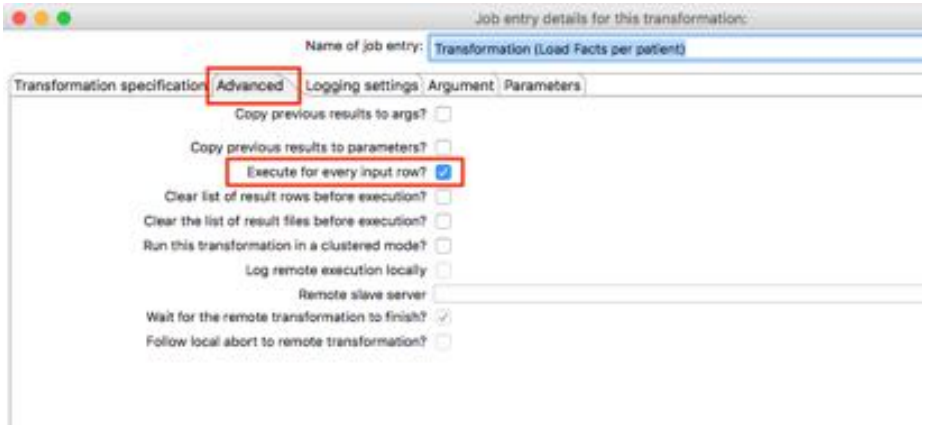
Mediante la opción de "Copy rows to result" en la segunda transformación puedo pasar el resultado de la misma al siguiente paso. De esta forma, consultando nuestra tabla dimensión de pacientes podemos devolver a la siguiente fase todas las filas de esta ejecución.

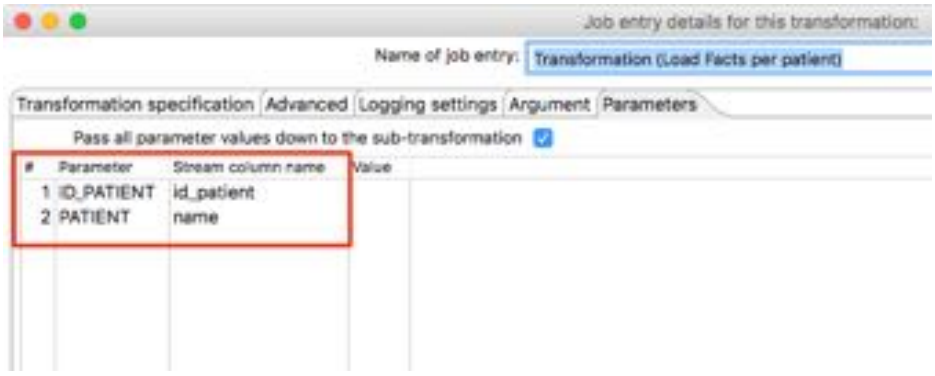
El siguiente paso consiste en adecuar la siguiente transformación para que esta pueda ejecutarse por cada fila que hemos obtenido anteriormente y pasar estos valores como parámetros. Para ello hay que realizar los siguientes pasos.

Dentro de la propia transformación "t_f_records", dentro de sus propiedades en la pestaña "Parameters" debemos definir los parámetros que necesitamos. En nuestro caso ID_PATIENT y PATIENT



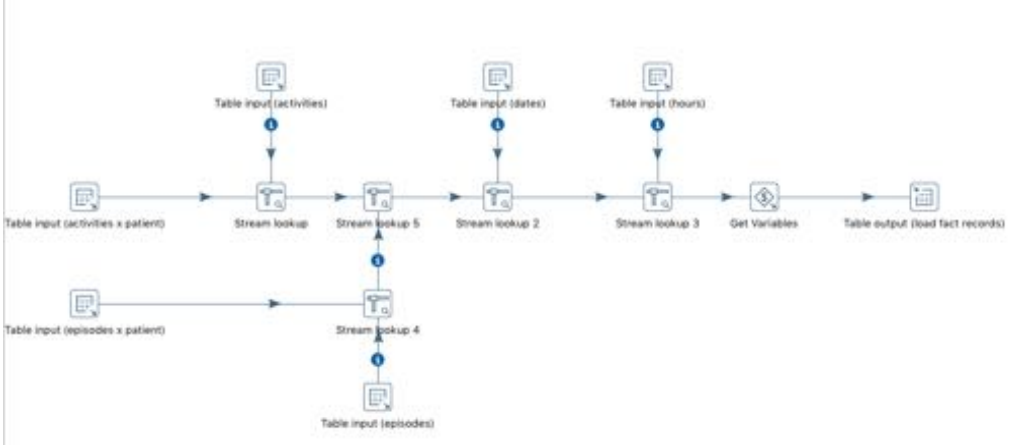
En el área de diseño de nuestro trabajo (job) "j_etl_facts", debemos configurar la instancia donde llamamos a la transformación "t_f_records" para que obtenga los valores provenientes del paso anterior y los pase mediante los parámetros que hemos definido para la transformación (pestaña "Parameters"). También es importante indicar que esta transformación debe ejecutarse por cada fila que le llegue (pestaña "Advanced").





De esta forma podremos luego en nuestra última transformación ejecutar la carga de los hechos para cada paciente, ya que dispondremos de tanto de su identificador en la BBDD como de su nombre.

La última transformación ("t_f_records") es donde vamos a llevar a cabo la carga completa de todos los hechos. Aunque el flujo que vamos a mostrar pueda parecer complejo, realmente el único punto de interés que tiene es el de hacer uso de los parámetros definidos anteriormente.



El resto de operaciones son cruces sencillos entre todas las tablas que contienen las medidas (actividad, episodios u horas de sueño) y cruzarlas con sus respectivas dimensiones para así obtener el identificador, de forma que al final en la tabla de hechos almacenemos las claves primarias de cada uno de los elementos.
Sólo existen dos excepciones:

1. Los datos referentes a las horas, ya que como hemos comentado durante el diseño dimensional, es una dimensión degenerada y por tanto se almacenará su valor directamente en la tabla de hechos.
2. Los datos relativos a los pacientes. En este caso no se realiza ningún cruce con la dimensión paciente, pues recordemos que hemos pasado la información por parámetros y por tanto todos los hechos que se carguen en un ciclo de esta transformación serán relativos a un único paciente. Haremos uso de la opción "Get Variable" para añadir esta información a la tabla que cargaremos finalmente.