

# Sistema Experto

Un Sistema Experto (SE) es un programa informático que tiene como solucionar problemas concretos utilizando la Inteligencia Artificial (IA). De esta forma, se busca que el SE se comporte como un profesional en la materia.

En este caso, el SE debe ser capaz de detectar, prevenir y monitorizar conductas autolesivas en adolescentes.

Hay 4 grupos de SEs:

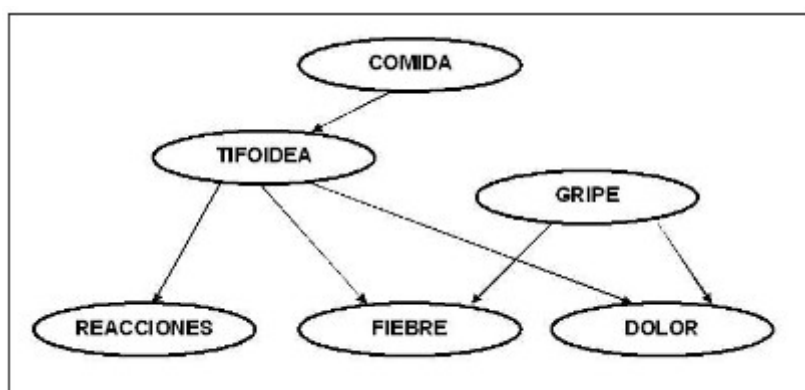
- SE basados en reglas
- SE basados en casos
- **SE basados en redes Bayesianas**
- SE basados en lógica difusa o borrosa

## Red bayesiana

Una red de Bayes o red bayesiana es un modelo gráfico que se muestran variables aleatorias (**nodos**) en un conjunto de datos y las dependencias que hay entre ellas (**arcos**), formando de esta manera, un grafo acíclico dirigido (DAG).

Cabe destacar que no todas las relaciones son directas entre las variables, sino que también pueden haber relaciones indirectas independientes, es decir, variables que están relacionados con otras a través de otros nodos.

Por ejemplo:



Este grafo muestra los síntomas, causas y factores que pueden venir de una enfermedad.

*reacciones* es condicional independiente de C, G, F, D dado *tifoidea* (donde: C es comida, T es tifoidea, G es gripe, R es reacciones, F es fiebre y D es Dolor). Esto es:

$$P(R|C, T, G, F, D) = P(R|T)$$

Esto se representa gráficamente por el nodo T separando al nodo R del resto de las variables.

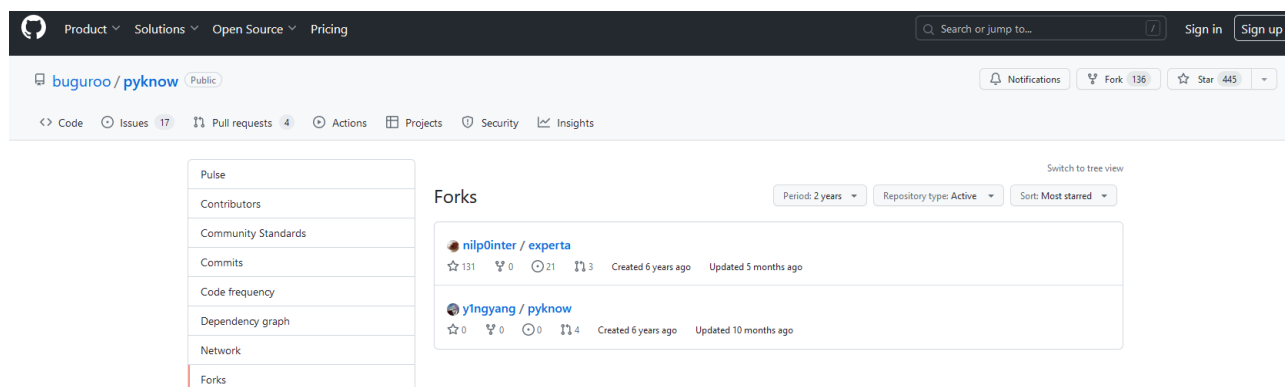
Por ejemplo, una red bayesiana puede representar las relaciones probabilísticas entre enfermedades y síntomas. Dados los síntomas, la red puede ser usada para computar la probabilidad de la presencia de varias enfermedades.

Una red bayesiana es un modelo gráfico que muestra variables (que se suelen denominar **nodos**) en un conjunto de datos y las independencias probabilísticas o condicionales entre ellas. Las relaciones causales entre los nodos se pueden representar por una red bayesiana; sin embargo, los enlaces en la red (también denominados **arcos**) no representan necesariamente una relación directa de causa y efecto. Por ejemplo, una red bayesiana se puede utilizar para calcular la probabilidad de un paciente con una enfermedad concreta, con la presencia o no de algunos síntomas y otros datos relevantes, si las independencias probabilísticas entre síntomas y enfermedad son verdaderas, tal y como se muestra en el gráfico. Las redes son muy robustas en los puntos en los que falta información y realizan las mejores predicciones posibles utilizando la información disponible.

Lauritzen y Spiegelhalter crearon un ejemplo común y básico de una red bayesiana en 1988. También se conoce como modelo "Asia" y es una versión simplificada de una red que se puede utilizar para diagnosticar a los nuevos pacientes de un médico; la dirección de los enlaces corresponde por lo general a la causalidad. Cada nodo representa una faceta que se puede relacionar con el estado de un paciente; por ejemplo, "fumador" indica que se trata de un fumador habitual y "VisitaAsia" muestra que recientemente ha viajado a Asia. Los enlaces entre los nodos muestran las relaciones probabilísticas; por ejemplo, fumar aumenta las posibilidades de que el paciente padezca bronquitis y cáncer de pulmón, mientras que la edad parece estar relacionada únicamente con la posibilidad de desarrollar cáncer de pulmón. De la misma forma, las anomalías detectadas en una radiografía de los pulmones pueden estar causadas por tuberculosis o cáncer de pulmón, mientras que las posibilidades de que un paciente tenga dificultades respiratorias (disnea) aumentan si también padece bronquitis o cáncer de pulmón.

Para programarlo, vamos a usar una librería de Python llamado Experta, que es una librería basada en la librería PyKnow (de hecho, *experta* es una fork del repositorio de PyKnow).

Repositorio de PyKnow: <https://github.com/buguroo/pyknow>



Forks del repositorio de PyKnow

A su vez, PyKnow está inspirada en CLIPS, un lenguaje de programación basada en C para sistemas expertos (Repositorio de Github de Experta: <https://github.com/nilp0inter/experta/tree/develop>).

Básicamente, *experta* es una librería que ofrece una alternativa de CLIPS de C en Python. *Experta* es compatible con Python3.

## Documentación de Experta:

- <https://github.com/nilp0inter/experta/blob/develop/docs/talks/Sistemas%20Expertos%20en%20Python%20con%20PyKnow%20-%20PyConES%202017/slides.pdf>
- <https://experta.readthedocs.io/en/latest/>
- <https://readthedocs.org/projects/experta/downloads/pdf/stable/>

## Bibliografía:

- <https://www.ibm.com/docs/es/spss-modeler/saas?topic=models-bayesian-network-node>
- [https://es.wikipedia.org/wiki/Red\\_bayesiana](https://es.wikipedia.org/wiki/Red_bayesiana)
- <https://ccc.inaoep.mx/~esucar/Clases-mgp/caprb.pdf>
- <https://www.unir.net/ingenieria/revista/sistema-experto/>

## Librerías que descargar para ejecutar el SE:

Todos estos comandos deberían estar en un fichero que se pueda ejecutar previamente llamado *requirements*

- *Experta*: \$ pip install experta