

Redes bayesianas. Causalidad e inferencia.

*Inteligencia Artificial – Ingeniería del Software Curso
2023/2024*

Propuesta de trabajo

Juan Manuel Álvarez Espada

1. Introducción y objetivos

Una red bayesiana es un modelo gráfico probabilístico que permite descubrir distribuciones conjuntas complejas (modelos, por ejemplo) a partir de distribuciones condicionales locales simples. Es la forma más básica y robusta de obtener conocimientos de entornos inciertos (lo que denominamos incertidumbre).

La representación más habitual de una red bayesiana es la de un grafo dirigido en el que cada nodo está comentado con información probabilística cuantitativa siendo su especificación simple la siguiente:

- Conjunto de variables aleatorias, X_i que forman los nodos de red. Pueden ser discretas o continuas.
- Conjunto de enlaces dirigidos que conectan pares de nodos. Se dice que **X es padre** de Y si hay una flecha que va del nodo X al nodo Y.
- Cada nodo X_i tiene una distribución / densidad de probabilidad condicionada $P(X_i | \text{Padres } X_i)$ que cuantifica el efecto de los padres del nodo.
- El **grafo es acíclico dirigido** o GAD.

Las redes bayesianas tienen muchas aplicaciones en campos tan dispares como: la biología; la medicina; el procesamiento del lenguaje o de imágenes, o los sistemas de toma de decisiones en ambiente de incertidumbre.

Centrándonos en esta última aplicación, la de toma de decisiones en ambiente de incertidumbre, un tipo de red bayesiana que puede utilizarse es la red causal donde el requisito esencial es que la relación establecida mediante enlaces exprese que la relación sea causal y no estrictamente condicional, formalizando la intervención de factores que determinan un resultado.

Riesgos en un proyecto

Un proyecto informático, por ejemplo, es un resultado único obtenido mediante la aplicación de un conjunto de recursos, generalmente determinados de forma progresiva, durante un período de tiempo establecido.

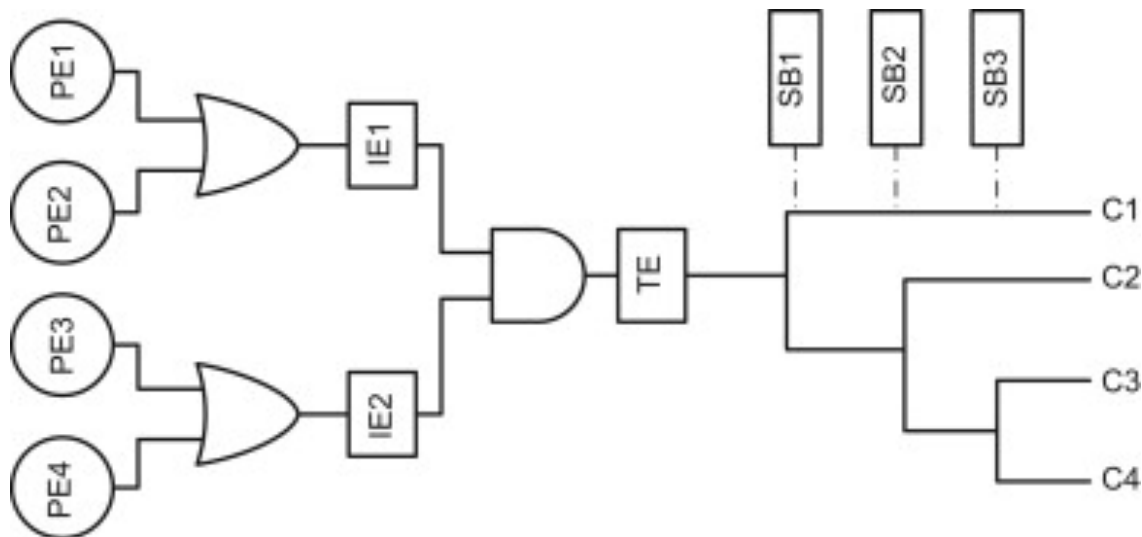
Como cualquier sistema, un proyecto se puede considerar un sistema durante un tiempo limitado, se podrán producir indeterminaciones y cambios en lo programado, debido a la

incertidumbre que se puede tener. Dicha incertidumbre no tiene una única fuente, pudiendo existir incertidumbres externas, internas o híbridas. Estas incertidumbres generan “riesgos”

Un riesgo se puede definir como un evento o suceso, que si ocurre puede tener un impacto negativo o positivo en un sistema, y por ende, en un proyecto. Un riesgo se compone de una probabilidad y de un nivel de impacto en el proyecto (o en una decisión).

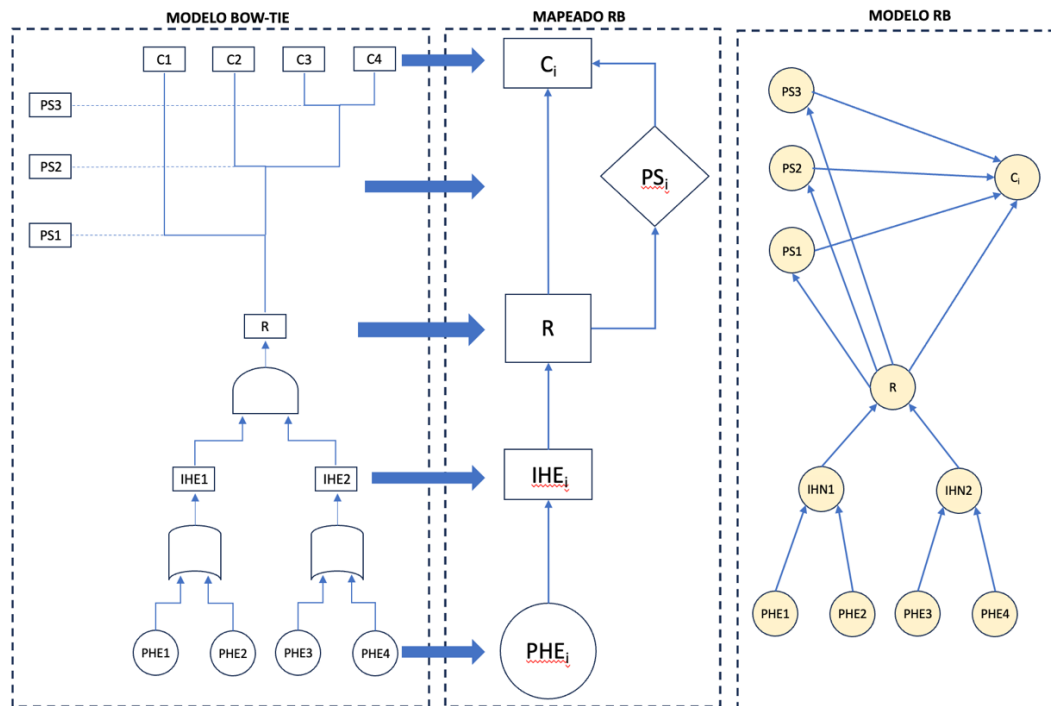
No existe un único riesgo, sino que existen riesgos que pueden estar relacionados con otros riesgos.

Un enfoque gráfico para entender este concepto es utilizar el modelo Bow-Tie para riesgos complejos, definido en (Khakzad et al., 2012):

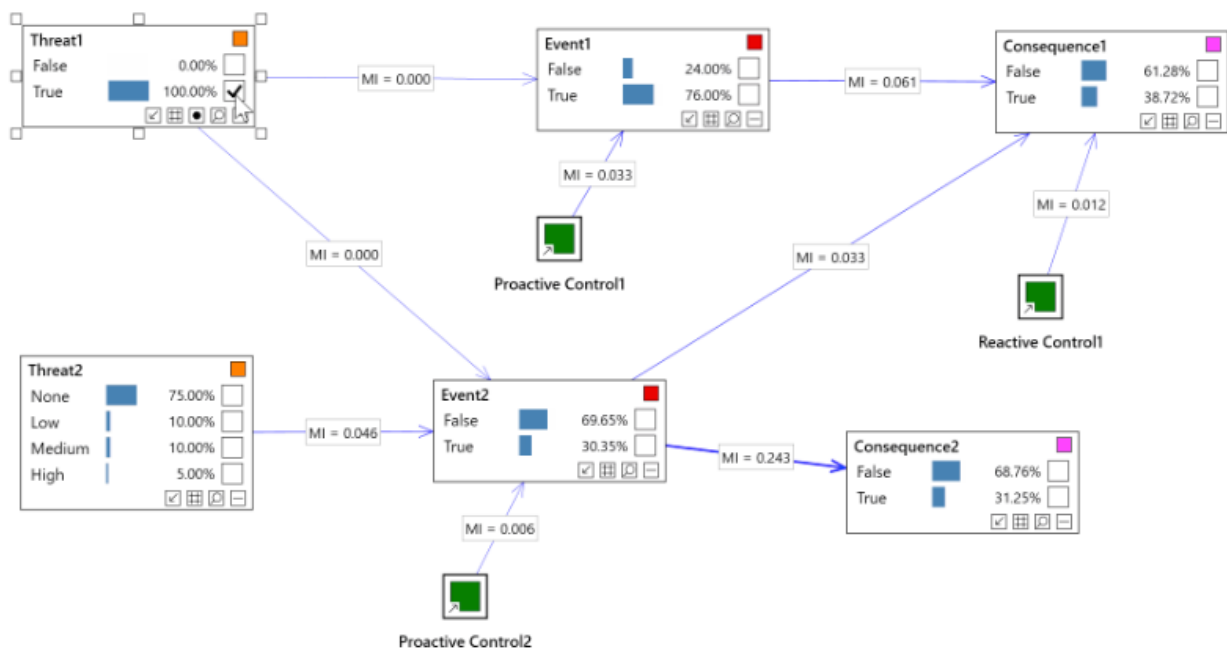


En este modelo podemos observar que un riesgo superior o riesgo sistémico es consecuencia de varios riesgos intermedios IE_i y estos a su vez son consecuencia de riesgos primarios PE_i , **obsérvese las puertas lógicas que unen los distintos riesgos**. A su vez, el riesgo sistémico o principal puede causar varias consecuencias C_i dependiendo de las barreras de seguridad SB_i establecidas.

Podemos convertir este modelo de gestión de riesgos en un modelo de red causal siguiendo el siguiente mapeado que entronca ambos modelos:



Existen algunos softwares comerciales que pueden resolver directamente este mapeado y poder inferir sobre la red causal para determinar qué consecuencias podría tener la probabilidad de producirse un riesgo. Por ejemplo, [Bayes Server](#) realiza un mapa de riesgo mediante el siguiente modelo:



Otras empresas que trabajan con softwares similares son: [Norsys Software Corp.](#), [SpiceLogic Inc.](#), [HuginExpert](#), [Quantics Biostatistics](#) o [BayesFusion LLC](#), etc.

El **objetivo principal** de esta propuesta es construir una red bayesiana causal que resuelva de la mejor manera posible una red de riesgos o de toma de decisiones, a raíz

de los resultados obtenidos al realizar una inferencia exacta o aproximada sobre el modelo. Para ello será necesario alcanzar los siguientes **objetivos específicos**:

1. Entender el funcionamiento de las redes bayesianas y su aplicación a tareas de decisión, y sobre todo de gestión de riesgos.
2. Aprender a usar la biblioteca [pgmpy](#). Esta biblioteca es una implementación en Python para redes bayesianas con enfoque modular y extensible. Implementa varios algoritmos de aprendizaje estructural, inferencia exacta y aproximada e inferencia causal.
3. Construir redes bow-tie que represente un proceso de gestión de riesgos o de toma de decisiones, de **al menos 10 nodos** y con estructura DAG como se ha representado anteriormente. Se utilizarán variables discretas con dominios finitos.
4. Realizar un proceso de aprendizaje automático para dicha red utilizando alguno de los modelos que aporta la librería (hay que estudiar cuál podría ser el mejor modelo para ser utilizado).
5. Documentar el trabajo realizado usando un formato de artículo científico. Incluyendo bibliografía en formato APA (7º Ed)
6. Realizar una presentación (PDF, PowerPoint o similar) de los resultados obtenidos.

2. Descripción del trabajo

A continuación, se describe con más detalle cómo debe llevarse a cabo el trabajo.

Implementación de la red bayesiana

Se deberá presentar una Red Bayesiana realista como ejemplo para realizar las pruebas. Puede ser inventada o basada en algún ejemplo real (en cuyo caso, se explicitará la correspondiente referencia), esta última opción será mejor valorada. Las probabilidades utilizadas deberán ser, también, lo más cercana a la realidad utilizando información que como en otros casos, deberá ser referenciada.

Inferencia exacta y aproximada

En este trabajo se pide utilizar en Python, al menos, dos algoritmos de inferencia exacta y dos algoritmos de inferencia aproximada de las aportadas por [pympg](#). Explicando las diferencias y aportando con cuál de los algoritmos se podría trabajar mejor.

Metodología

Para la consecución de los dos primeros objetivos específicos, se debe estudiar con detalle las redes bayesianas y su traslación a Python. Para el primer objetivo existe suficiente bibliografía al respecto. Al final de este documento vienen varios documentos accesibles para poder obtener criterios al respecto. Es importante conocer la sintaxis, la semántica y los procesos de inferencia exacta y aproximada.

Respecto al segundo punto, la documentación sobre la librería [pgmpy](#) es extensa incluso en la propia página de la librería. En bibliografía se aporta información adicional.

Se indica a continuación los requisitos mínimos que se exigen a los modelos construidos en función de la convocatoria de la que se trate:

- En la **convocatoria de junio**, cada grupo deberá tratar previamente con el profesor de esta propuesta, que modelo va a utilizar, y sobre qué proyecto o sistema (informático, control, procesos industriales, toma de decisiones, gestión de crisis) se va a realizar el trabajo. En cuanto al proceso de aprendizaje, se indica que vienen dentro de la librería modelos de aprendizaje. En caso de querer utilizarse otro se propondrá previamente.
- En la **convocatoria de julio**, se deberá realizar un modelo de red bayesiana causal distinta a la convocatoria anterior y utilizar modelos de inferencia exacta y aproximada que no se hayan utilizado en la convocatoria anterior. Se deberá realizar el proceso de aprendizaje automático.
- En la **convocatoria de octubre** se debe de realizar un modelo de red bayesiana causal dinámica con dos iteraciones y posteriormente utilizar dos modelos de inferencia exacta y dos modelos de inferencia aproximada (elegidos por el alumno).

Se puede utilizar Google Colaboratory o Anaconda Jupyter Notebook. En Anaconda se deberá utilizar el entorno BAP3.YML, donde se definen los paquetes necesarios para poder trabajar como se define en la página de la librería [pgmpy](#).

No se permitirá un mismo modelo de datos de partida para diferentes grupos.

Documentación y entrega

El trabajo deberá documentarse siguiendo un formato de artículo científico, con una **extensión mínima de 6 páginas**. En la página web de la asignatura se pueden encontrar plantillas donde se sugiere una estructura general. Estas plantillas siguen el formato de los *IEEE conference proceedings*, cuyo sitio web guía para autores ofrece información más detallada. El documento entregado deberá estar en formato PDF.

En el caso concreto de este trabajo, la memoria deberá al menos incluir: introducción; descripción del modelo que se va a modelar, descripción de la red bayesiana, nodos, enlaces, probabilidades tomadas, modelo creado, inferencia elegida, aprendizaje automático, explicando las dificultades encontradas y las decisiones de diseño adoptadas para abordarlas; descripción de los resultados alcanzados; conclusiones; bibliografía. **En ningún caso debe incluirse código en la memoria.**

La entrega del trabajo consistirá en la memoria del trabajo y el código implementado (ficheros py o cuadernos de Jupyter). La memoria del trabajo y el código implementado deben subirse a la página de la asignatura en **un único fichero comprimido zip**. El fichero zip deberá nombrarse, al igual que los ficheros que contiene con los primeros apellidos de los miembros del grupo. **En caso de no proporcionar un fichero con ese nombre exacto esa parte de la evaluación no se realizará.**

Presentación y defensa

Como parte de la evaluación del trabajo se deberá realizar una defensa de este, para lo que se citará a los alumnos de manera conveniente.

El día de la defensa se deberá realizar una pequeña presentación (PDF, PowerPoint o similar) de 10 minutos en la que participarán activamente todos los miembros del grupo que han desarrollado el trabajo. Esta presentación deberá seguir a grandes rasgos la misma estructura que la memoria del trabajo, haciendo especial mención a los resultados obtenidos y al análisis crítico de los mismos.

En los siguientes 10 minutos de la defensa, el profesor procederá a realizar preguntas sobre el trabajo, que podrán ser tanto de la memoria como del código fuente.

3. Evaluación del trabajo

Para que el trabajo pueda ser evaluado es imprescindible asegurarse que todos los modelos proporcionados puedan cargarse sin errores.

Para la evaluación del trabajo se tendrán en cuenta los siguientes criterios, considerando una nota total máxima de **4 puntos**:

- **Memoria del trabajo (hasta 1 punto):** se valorará la claridad de las explicaciones, el razonamiento de las decisiones, el análisis y presentación de resultados y el correcto uso del lenguaje. La elaboración de la memoria debe ser original, por lo que no se evaluará el trabajo si se detecta cualquier copia del contenido.
- **Código fuente (hasta 1 punto):** se valorará la claridad y buen estilo de programación, corrección y eficiencia de la implementación y calidad de los comentarios. El código debe ser original, por lo que no se evaluará el trabajo si se detecta código copiado o descargado de internet.
- **Modelo seleccionado en inferencia (hasta 1 punto):** se valorará, tanto de manera absoluta como comparativamente con el resto de los trabajos, el comportamiento de la red seleccionada.
- **Presentación y defensa (hasta 1 punto):** se valorará la claridad de la presentación y la buena explicación de los contenidos del trabajo, así como, especialmente, las respuestas a las preguntas realizadas por el profesor.

IMPORTANTE: cualquier plagio, compartición de código o uso de material que no sea original y del que no se cite convenientemente la fuente, **significará automáticamente la calificación de cero en la asignatura para todos los alumnos involucrados**. Por tanto, a estos alumnos no se les conserva, ni para la actual ni para futuras convocatorias, ninguna nota que hubiesen obtenido hasta el momento. Todo ello sin perjuicio de las correspondientes medidas disciplinarias que se pudieran tomar.

4. Bibliografía inicial

La bibliografía inicial podría ser (puede ampliarse):

- Fridgeirsson, T. V., Ingason, H. T., Jonasson, H. I., & Kristjansdottir, B. H. (2021). The VUCAly of Projects: A New Approach to Assess a Project Risk in a Complex World. *Sustainability* 2021, Vol. 13, Page 3808, 13(7), 3808. <https://doi.org/10.3390/SU13073808>
- Huang, Y., Zhang, Z., Tao, Y., & Hu, H. (2022). Quantitative risk assessment of railway intrusions with text mining and fuzzy Rule-Based Bow-Tie model. *Advanced Engineering Informatics*, 54, 101726. <https://doi.org/10.1016/J.AEI.2022.101726>
- Jamshidi, A., Ait-kadi, D., Ruiz, A., & Rebaiaia, M. L. (2018). Dynamic risk assessment of complex systems using FCM. *International Journal of Production Research*, 56(3), 1070–1088. <https://doi.org/10.1080/00207543.2017.1370148>
- Khakzad, N., Khan, F., & Amyotte, P. (2012). Dynamic risk analysis using bow-tie approach. *Reliability Engineering & System Safety*, 104, 36–44.

- <https://doi.org/https://doi.org/10.1016/j.ress.2012.04.003>
- Martin, O. A., Kumar, R., & Lao, J. (2021). *Bayesian modeling and computation in Python*. Chapman and Hall/CRC.
- Murphy, K. P. (2002a). A tutorial on dynamic bayesian networks. *Internet*, November.
- Murphy, K. P. (2002b). Dynamic bayesian networks. *Probabilistic Graphical Models*, M. Jordan, 7, 431.
- Murphy, K. P. (2002c). *Dynamic bayesian networks: representation, inference and learning*. University of California, Berkeley.
- Russell, S. J., & Norvig, P. (2004). *Inteligencia Artificial: un enfoque moderno*.
- Wu, X., Huang, H., Xie, J., Lu, M., Wang, S., Li, W., Huang, Y., Yu, W., & Sun, X. (2023). A novel dynamic risk assessment method for the petrochemical industry using bow-tie analysis and Bayesian network analysis method based on the methodological framework of ARAMIS project. *Reliability Engineering & System Safety*, 237, 109397.
- Zurheide, F. T., Hermann, E., & Lampesberger, H. (2021). pyBNBowTie: Python library for Bow-Tie analysis based on Bayesian Networks. *Procedia Computer Science*, 180, 344–351.