



Revista EIA
ISSN: 1794-1237
revista@eia.edu.co
Escuela de Ingeniería de Antioquia
Colombia

CLASIFICADOR BAYESIANO DE DOS CLASES PARA SELECCIONAR LA MEJOR REGLA DE PRIORIDAD EN UN PROBLEMA JOB SHOP: OPEN SHOP

Castrillón Gomez, Omar Danilo; Ariel Sarache, William; Ruiz Herrera, Santiago

CLASIFICADOR BAYESIANO DE DOS CLASES PARA SELECCIONAR LA MEJOR REGLA DE PRIORIDAD EN UN PROBLEMA JOB SHOP: OPEN SHOP

Revista EIA, vol. 16, núm. 31, 2019

Escuela de Ingeniería de Antioquia, Colombia

Disponible en: <https://www.redalyc.org/articulo.oa?id=149258931004>

DOI: <https://doi.org/10.24050/reia.v16i31.867>



Esta obra está bajo una Licencia Creative Commons Atribución-NoComercial-SinDerivar 4.0 Internacional.

CLASIFICADOR BAYESIANO DE DOS CLASES PARA SELECCIONAR LA MEJOR REGLA DE PRIORIDAD EN UN PROBLEMA JOB SHOP: OPEN SHOP

A TWO-CLASS BAYESIAN CLASSIFIER TO SELECT THE BEST PRIORITY RULE IN A JOB SHOP: OPEN SHOP SCHEDULING PROBLEM

CLASSIFICADOR BAYESIANO DE DUAS CLASSES PARA SELECIONAR A MELHOR REGRA DE PRIORIDADE EM UMA LOJA DE TRABALHO
PROBLEMA: ABRIR LOJA

Omar Danilo Castrillón Gomez odcastrillong@unal.edu.co

Universidad Nacional de Colombia – Sede Manizales, Colombia

William Ariel Sarache wasarachec@unal.edu.co

Universidad Nacional de Colombia – Sede Manizales, Colombia

Santiago Ruiz Herrera sruizhe@unal.edu.co

Universidad Nacional de Colombia – Sede Manizales, Colombia

Revista EIA, vol. 16, núm. 31, 2019

Escuela de Ingeniería de Antioquia,
Colombia

Recepción: 22 Enero 2016
Aprobación: 02 Abril 2018

DOI: <https://doi.org/10.24050/reia.v16i31.867>

Redalyc: <https://www.redalyc.org/articulo.oa?id=149258931004>

Resumen: El objetivo de este trabajo es seleccionar, por medio de un clasificador bayesiano de dos clases, la mejor regla de prioridad que puede ser aplicada en un problema Job Shop: Open Shop. En una primera fase se expone el diseño del clasificador, entrenado con 300 problemas generados aleatoriamente. En 150 de ellos, la mejor regla de prioridad para secuenciarlos fue FIFO (First in First Out) y en los restantes fue la regla LPT (Long Process Time). En una segunda fase, un conjunto de 300 problemas diferentes, con las mismas características de la primera fase, fueron generados aleatoriamente. Estos problemas fueron clasificados previamente (sin secuenciarlos) por medio la técnica bayesiana propuesta. Los resultados demuestran que en el 96% de los casos, el clasificador propuesto logra identificar la mejor regla de prioridad para secuenciar pedidos.

Palabras clave: Programación de la producción, Reglas de Prioridad, Clasificador Bayesiano, Job Shop: Open Shop.

Abstract: The aim of the present paper to select, through a two-class Bayesian classifier, the best priority rule to solve a Job Shop: Open Shop scheduling problem. In a first phase, the design of the classifier, trained with 300 randomly problems, is exposed. In 150 of them, the best priority rule for sequencing was FIFO (First In First Out) while in the rest was the LPT (Long Process Time). In a second phase, a set of 300 different problems, with the same characteristics of the first phase, were randomly generated. These problems were classified previously (without sequencing them) through the proposed Bayesian technique. The results show that, in 96% of cases, the proposed classifier identifies the best priority rule to sequence the orders.

Keywords: Production scheduling, Priority Rules, Bayesian classifier, Job Shop: Open Shop.

Resumo: O objetivo deste trabalho é seleccionar, por meio de um classificador bayesiano de duas classes, a melhor regra de prioridade que pode-se aplicar em um problema Job Shop: Open Shop. Na primeira fase expõe-se o design do classificador, treinado com 300 problemas gerados aleatoriamente. Em 150, a melhor regra de prioridade para os

seguenciar foi FIFO (First in First out), e nos outros foi LPT (Long Process Time). Na segunda fase, um conjunto de 300 problemas diferentes, com as mesmas características da primeira fase, foram gerados aleatoriamente. Esses problemas foram classificados previamente (sem sequenciar) por meio da técnica bayesiana proposta. Os resultados mostraram que no 96% dos casos, o classificador proposto logrou identificar a melhor regra de prioridade para sequenciar os pedidos.

Palavras-chave: Programação de produção, regras de prioridade, Classificador Bayesiano, Job Shop: Open Shop.

Introducción

En la actualidad, las reglas de prioridad continúan siendo una técnica ampliamente usada en la secuenciación de procesos en diversas industrias. Sin embargo, la dinámica del mercado hace que las empresas manufactureras trabajen con una serie de reglas de programación predefinidas, pues muchas veces no se dispone del tiempo necesario para experimentar con otras alternativas con el fin de determinar la herramienta más eficiente.

Particularmente, en ambientes de manufactura Job Shop: Open Shop, cada trabajo debe ser procesado en una o varias máquinas sin importar el orden. En este caso, el número de posibles soluciones de secuenciación es del orden $n!M$, donde N es el número de pedidos y M es el número de máquinas. Dada la complejidad que implica un problema combinatorio de este orden, resulta bastante difícil encontrar una solución óptima o sub óptima en la tarea de programación de pedidos. No obstante, la aplicación de nuevas metodologías basadas en técnicas de inteligencia artificial permite solucionar este tipo de problemas ahorrando tiempo en cálculos innecesarios. Dentro de ellas, se destacan los denominados clasificadores bayesianos a pesar de que han sido muy poco usados en la solución de problemas de programación de la producción

Los técnicas bayesianas pueden ser consideradas como clasificadores probabilísticos fundamentados en los teoremas de Bayes (Fernandez, 2016). Por medio de estos clasificadores, es posible agrupar datos en diferentes clases definidas por un rango de probabilidades y, posteriormente, reconocer a priori la clase a la que un nuevo dato puede pertenecer. Específicamente en los problemas de secuenciación de pedidos, estos clasificadores permiten agrupar los problemas en diversas clases según la regla de prioridad que mejor permita secuenciar un conjunto de órdenes de fabricación, con el menor tiempo total de proceso u otra métrica de desempeño. Adicionalmente, este clasificador permite reconocer de forma previa a qué clase puede pertenecer cada configuración de programación de N pedidos y M máquinas. Este último aspecto evita realizar un nuevo secuenciamiento cada vez que dicho problema se presente.

Múltiples aplicaciones de las técnicas bayesianas se han identificado en diversos campos de estudio, tales como en la agricultura (Baltazar, aranda, & Aguilar, 2008), la bio-ingiería (Dallaire, Giguère, Émond, & Chaib-draa, 2014), la domótica (Del Sagrado, Sanchez, Rodriguez, & Berenguel, 2016), la inmunología (Hanan, Concha, Toro, & Larragaña, 2013), la

medicina (He, Liu, Hu, Wen, Wan, & Long, 2015), (Karabatak, 2015), (Wiggins, Saad, Litt, & Vachtsevanos, 2008), (Yin, Kissinger, Moreno, Galinski, & Styczynski, 2015), la seguridad (Mukherjee & Sharmaa, 2012), (Xiang, Yong, & Meng, 2008) los reconocimientos de video (Roy, Shivakumara, Roy, Pal, Tan, & Lu, 2015) y la optimización (Salama & Freitas, 2014), entre otras (Mujalli, Lopez, & Garach, 2016), (Zaidan, Ahmad, Karim, Larbani, Zaidan, & Sali, 2014).

Sin embargo, la revisión bibliográfica permitió identificar pocas contribuciones asociadas a problemas de programación de la producción, lo cual sugiere que la aplicación de clasificadores bayesianos en este campo de estudio es un tópico que amerita más investigación. En particular, solo se encontró un trabajo orientado a la aplicación de técnicas bayesianas en un sistema de producción Flexible Job Shop (Sun, Lin, Wang, Gen, & Kawakami, 2015). Dicho trabajo abordó el problema a partir de un algoritmo evolucionario híbrido basado en un enjambre de partículas y una red bayesiana con el fin de minimizar el tiempo total de proceso.

Tomando en consideración los aspectos antes planteados, el objetivo del presente artículo es presentar el proceso de diseño de un clasificador bayesiano para identificar la regla de prioridad más adecuada que permita secuenciar un conjunto de ordenes de fabricación en un ambiente de manufactura Open Shop: Job shop.

El clasificador fue entrenado previamente con el fin de establecer la regla de prioridad más adecuada para secuenciar un conjunto de pedidos. Posteriormente, fue validado con otro conjunto de datos diferentes. Como resultado se encontró que en el 96% de los casos se logró identificar la regla de prioridad más adecuada para aplicar a cada configuración de problema NxM.

El resto del artículo se ha estructurado de la siguiente manera: en la Sección 2 se expone la metodología usada para la construcción del clasificador. En la Sección 3 se presentan los resultados experimentales observados. Finalmente, en la Sección 4, se exponen un conjunto de conclusiones asociadas a los resultados obtenidos.

2. Materiales y Métodos

La metodología usada consta de los siguientes pasos:

Paso 1: Definición del conjunto de problemas. En el diseño de esta investigación se definieron dos conjuntos de problemas: problemas para el entrenamiento y problemas para la validación. El primer conjunto permite definir las funciones probabilísticas bayesianas para cada clase. El segundo, tiene como fin probar el clasificador bayesiano y establecer su efectividad en el reconocimiento de la clase más adecuada a la cual puede pertenecer un problema de programación. Una descripción del tratamiento dado a cada conjunto es la siguiente:

a) Conjunto 1 (entrenamiento): Este conjunto se conformó con 300 problemas Job Shop: Open Shop. En 150 de ellos la mejor regla de prioridad para secuenciar el problema fue la técnica FIFO. En los otros 150 problemas la mejor regla de prioridad para secuenciar el problema fue la técnica LPT. Cada uno de los dos subconjuntos de 150 problemas fueron agrupados en categorías JSSPNxM, donde N es el número de

pedidos y M es el número de máquinas, así: 50 problemas JSSP5x5; 50 problemas JSSP 5x6 y 50 problemas JSSP 6x6.

b) Conjunto 2 (validación): En este conjunto se definieron 300 problemas Job Shop: Open Shop, organizados de forma similar a los problemas definidos en la fase de entrenamiento. La generación de todos los problemas se hizo de forma aleatoria.

Paso 2: Diseño del Clasificador. Para la definición del clasificador bayesiano se aplicó el siguiente subprocedimiento:

a) Se define x_i , como una hiper matriz de tres dimensiones, como el conjunto de características. Las dimensiones son definidas de la siguiente forma: Las filas representan el número de problemas objeto de análisis, las columnas son el número de características, de ancho $n \times m$ (n = pedidos; m = máquinas) y c es el número de clases o reglas de prioridad a identificar, en este caso 2.

b) Se calcula el vector de medias u_i y la matriz de covarianza de Σ_i y x_i , K termino independiente.

c) Para cada clase (LIFO o LPT) se define una función de probabilidad según el siguiente sistema de ecuaciones (Duda, Hart, & Stork, 2001, Pag 41):

$$P_i(x) = X^t W_i X + w_i^t X + w_{i0} \quad (1)$$

Donde,

$$W_i = -\frac{1}{2} \Sigma_i^{-1} \quad (2)$$

$$w_i = \Sigma_i^{-1} u_i \quad (3)$$

$$w_{i0} = -\frac{1}{2} u_i^t \Sigma_i^{-1} u_i - \frac{1}{2} \ln(|\Sigma_i|) + \ln(k) \quad (4)$$

Paso 4. Validación del sistema. Cada uno de los conjuntos de validación del paso 1, fueron evaluados en las ecuaciones generadas en el paso 3. La ecuación que genere el mayor resultado, supondrá la clase a la cual debería pertenecer el dato analizado. Los resultados generados por el clasificador fueron comparados con la mejor regla de prioridad que debe secuenciar el problema, obteniéndose el número de aciertos o errores del clasificador.

3. RESULTADOS y DISCUSIÓN

Al aplicar la metodología descrita, se obtuvieron los siguientes resultados:

Paso 1. La Tabla 1, resume la forma como fueron clasificados los problemas descritos en la metodología.

Tabla 1
Tabla 1. Definición de problemas

Problemas	Entrenamiento	Entrenamiento	Validación	Validación
	FIFO	LPT	FIFO	LPT
JSSP 5x5	50	50	50	50
JSSP 5x6	50	50	50	50
JSSP 6x6	50	50	50	50

Tabla 1. Definición de problemas
Tabla 1. Tabla 1. Definición de problemas

Por su parte, las Figuras 1 a la 3, muestran el tiempo total de proceso de estos problemas, respecto a la reglas de prioridad objeto de comparación.

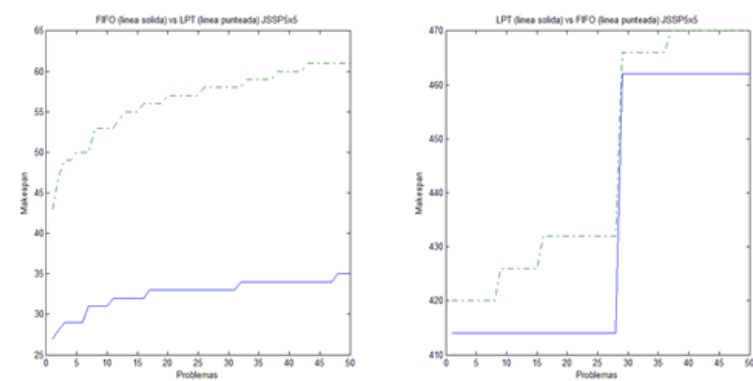


Figura 1
Figura 1. Comparación problemas JSSP 5X5. 50 problemas de entrenamiento.

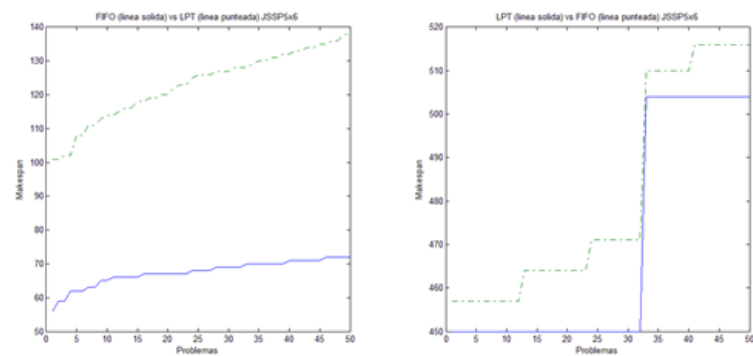


Figura 2
Figura 2. Comparación problemas JSSP 5X6. 50 problemas de entrenamiento.

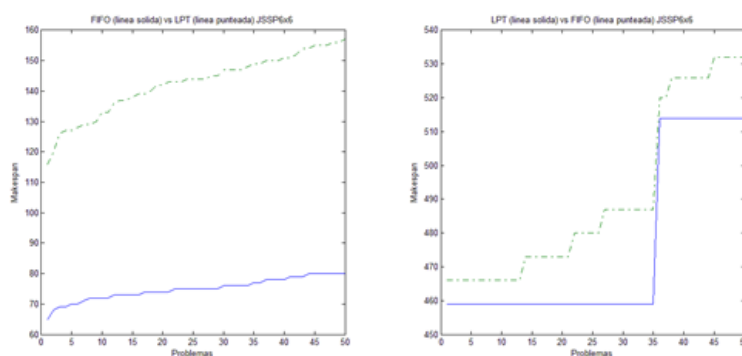
**Figura 3**

Figura 3. Comparación problemas JSSP 6X6. 50 problemas de entrenamiento

En cada una de las anteriores Figuras 1, 2 y 3 se observa que en la definición de los problemas existe una diferenciación marcada, dado que sus tiempos totales de proceso (makespan) se encuentran separados, pudiéndose decir que pertenecen a clases diferentes. En los pasos siguientes 2 y 3, se determinaron la funciones bayesianas para cada una de estas clases, aspecto que facilita la interpretación de los resultados en una posterior fase de validación.

Pasos 2 y 3. Se obtuvo la valoración promedio en la función generada por el clasificador bayesiano, con la cual serán identificadas cada una de las clases mostradas en la Tabla 2:

Tabla 2

Tabla 2. Probabilidad media por clases de problemas

Problemas	Resultados promedio función bayesiana FIFO	Resultados promedio función bayesiana LPT
JSSP 5x5	-62.774,9	-1,341597175570793*1014
JSSP 5x6	-0,16604	-1,205390604469413*1015
JSSP 6x6	-27,290	-1.717062192212598*1016

Tabla 2 Tabla 2. Probabilidad media por clases de problemas

La Tabla 2 ilustra una clara separación entre las clases definidas por las funciones bayesianas, lo cual hace factible su identificación.

Paso 4. En la Tabla 3 se expone el porcentaje de aciertos según la clase de cada problema analizado.

Tabla 3

Tabla 3. Resultados de validación

Problemas	Cantidad	Aciertos FIFO	Aciertos LPT	Promedio
JSSP 5x5	100	76%	100%	88%
JSSP 5x6	100	100%	100%	100%
JSSP 6x6	100	100%	100%	100%
Promedio General				96%

Tabla 3 Tabla 3. Resultados de validación

En la literatura revisada no se encontró una aplicación de los clasificadores bayesianos como la ilustrada en este artículo. De hecho, la aplicación de los clasificadores en esta área es escasa o nula. Por lo tanto, se presenta una nueva técnica de selección de la regla de prioridad más adecuada sin necesidad de realizar un nuevo proceso de secuenciación para cada configuración en particular.

Como se demostró en la Tabla 3, el porcentaje de aciertos (veces en que el clasificador determinó de forma previa la regla de prioridad más adecuada) fue del 96%. No obstante, este porcentaje puede variar según la naturaleza de los mismos problemas.

4. Conclusiones

En este documento se presenta un clasificador bayesiano para identificar dos clases de problemas de secuenciación con una efectividad del 96%. La metodología propuesta está basada en una generación de funciones de probabilidad para cada una de las clases creadas. Sobre las anteriores funciones se prueban los datos de entrenamiento y se supone que el dato pertenece aquella clase cuya función bayesiana sea máxima. Como futuras líneas de investigación se plantea modificar el clasificador bayesiano con el fin de que se puedan reconocer más de dos clases, aspecto que le dará más versatilidad para abordar una mayor diversidad de problemas.

Agradecimientos

5. Agradecimientos

Se agradece la colaboración a la Universidad Nacional de Colombia, específicamente al departamento de Ingeniería Industrial de la Facultad de Ingeniería y arquitectura de la Sede Manizales.

6. Referencias

- Baltazar, A., Aranda, J. I. & Aguilar, G. G. (2008). Bayesian classification of ripening stages of tomato fruit using acoustic impact and colorimeter sensor data. *Computers and Electronics in Agriculture*, No. 60, pp. 113-121.
- Dallaire, P., Giguère, P., Émond, D. & Chaib-draa, B. (2014). Autonomous tactile perception: A combined improved sensing and Bayesian nonparametric approach. *Robotics and Autonomous Systems*, No. 62, pp. 422-435.
- Del Sagrado, J., Sanchez, J. A., Rodriguez, F. & Berenguel, M. (2016). Bayesian networks for greenhouse temperature control. *Journal of Applied Logic*, <http://dx.doi.org/10.1016/j.jal.2015.09.006>, Article in press.
- Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern Classification*. New York, Estados Unidos: John Wiley & Sons, Pagina 41.
- Fernandez, E. (2016). *Análisis de clasificadores Bayesianos*. Argentina: Laboratorio de sistemas Inteligentes, Consultado 18 de febrero de 2006, disponible en <http://materias.fi.uba.ar/7550/clasificadores-bayesianos.pdf>

- Hanen , B., Concha , B., Toro, C. & Larragaña, P. (2013). Predicting human immunodeficiency virus inhibitors using multi-dimensional Bayesian network classifiers. *Artificial Intelligence in Medicine*, No. 57, pp. 219-229.
- He, L., Liu, B., Hu, D., Wen, Y., Wan, M. & Long, J. (2015). Motor Imagery EEG Signals Analysis Based on Bayesian Network with Gaussian Distribution. *Neurocomputing*, <http://dx.doi.org/10.1016/j.neucom.2015.05.133> (Article in press).
- Karabatak, M. (2015). A new classifier for breast cancer detection based on Naïve Bayesian. *Measurement*, No. 72, pp. 32-36.
- Mujalli, R. O., Lopez, G. & Garach L. (2016). Bayes classifiers for imbalanced traffic accidents data sets. *Accident Analysis and Prevention*, No. 88, pp. 37-51.
- Mukherjee, S. & Sharmaa, N. (2012). Intrusion Detection using Naive Bayes Classifier with Feature Reduction. *Procedia Technology*, No. 4, pp. 119-128.
- Roy, S., Shivakumara, P., Roy, P. P., Pal, U., Tan, C. L. & Lu, T. (2015). Bayesian classifier for multi-oriented video text recognition system. *Expert Systems with Applications*, No. 42, pp. 5554-5556.
- Salamanca, K. M. & Freitas, A. A. (2014). Classification with cluster-based Bayesian multi-nets using Ant Colony Optimization. *Swarm and Evolutionary Computation*, No. 18, pp. 54-70.
- Sun, L., Lin, L., Wang, Y., Gen, M. & Kawakami, H. (2015). A Bayesian Optimization-based Evolutionary Algorithm for Flexible Job Shop Scheduling. *Procedia Computer Science*, No. 61, pp. 521-526.
- Wiggins, M., Saad, A. Litt, B. Vachtsevanos, G.(2008). Vachtsevanos, G. Evolving a Bayesian classifier for ECG-based age classification in medical applications. *Applied Soft Computing*, No. 8, pp. 599-608.
- Xiang, C., Yong, P. C. & Meng, L. S. (2008). Design of multiple-level hybrid classifier for intrusion detection system using Bayesian clustering and decision trees. *Pattern Recognition Letters*, No. 29, pp. 918-924.
- Yin, W., Kissinger, J. C., Moreno, A., Galinski, M. R. & Styczynski, M. P. (2015). From genome-scale data to models of infectious disease: A Bayesian network-based strategy to drive model development. *Mathematical Biosciences*, No. 260, pp. 156-168.
- Zaidan, A., Ahmad, N., Karim, H. A., Larbani, M., Zaidan & B. Sali. A. (2014). On themulti-agent learning neural and Bayesian methods in skin detector and pornography classifier: An automated anti-pornography system. *Neurocomputing*, No. 131, pp. 397-418.