

Metaheurísticas aplicadas a Selección de Características

Andrés Herranz González

Departamento de Inteligencia Artificial, Universidad Politécnica de Madrid, Spain

Resumen

En el problema de selección de subconjuntos de características, un algoritmo de aprendizaje se enfrenta con el problema de extraer el subconjunto de características más relevante, ignorando el resto. Para lograr el mejor rendimiento posible del algoritmo, se debe considerar cómo interactúan este y el conjunto de entrenamiento. Los algoritmos que optimizan la relación entre las características y la eficiencia de los algoritmos clasificadores que posteriormente utilizaran estas, se denominan de tipo *wrapper*. En este documento se realizará un análisis de las propuestas más relevantes de los dos últimos años.

Keywords: metaheurísticas; Selección de Características; Swarm Intelligence; funciones de transferencia; *wrapper*; aprendizaje automático

1. Introducción

La selección de características o FS (del inglés *Feature Selection*) es una técnica para reducir la dimensionalidad de los datos cuando se desea utilizar estos para problemas de inducción. El proceso de identificación y eliminación de información irrelevante o redundante se conoce como selección de subconjunto de características. La reducción de la dimensionalidad de los datos permite que los algoritmos de clasificación funcionen mucho más rápido y de manera más precisa.

Tradicionalmente, en la literatura, se han dividido los algoritmos de selección de características en tres grandes tipos (Jain y Singh, 2018): algoritmos de *filtrado*, algoritmos *wrapper* y algoritmos *embedidos*. Además, para combinar las ventajas de los métodos de *filtrado* y *wrapper*, se ha propuesto una cuarta categoría de algoritmos denominados *híbridos*.

Un modelo de *filtrado* evalúa las variables sin emplear ningún algoritmo de ML (del inglés *Machine Learning*) y utilizando las características de los datos (Liu, 2007). Este tipo de algoritmos seleccionan las características después de realizar un preprocesamiento. Algunos de los métodos de filtro más comunes son: Ganancia de información, Información mutua, Chi2, Puntuación de Fisher y ReliefF. La principal desventaja del enfoque de filtrado es que ignora los efectos del subconjunto de características seleccionado en el rendimiento del algoritmo de inducción.

Los métodos de *wrapper* hacen la selección de características teniendo en cuenta el algoritmo de aprendizaje que se utilizará. La principal ventaja sobre los métodos de filtrado es que encuentran las variables más relevantes y realizan una selección óptima de las características para el algoritmo de aprendizaje (Ang y col., 2016). Además, consideran las dependencias entre las características y proporcionan resultados más precisos en comparación con los métodos de filtro (Erguzel y col., 2014). El principal problema de este tipo de algoritmos es que tienen que ejecutar el algoritmo de inducción para cada configuración de las variables seleccionadas por el algoritmo de FS. Por tanto, este método es muy complejo y más propenso a hacer *over-fitting* en pequeños conjuntos de datos de entrenamiento.

En el enfoque de selección de características *embedded*, la búsqueda se guía por el proceso de aprendizaje del algoritmo de inducción. Este método hace un mejor uso de los datos disponibles y proporciona una solución más rápida, ya que no requieren la división de los datos de entrenamiento en el conjunto de entrenamiento y el conjunto de validación. Son menos costosos computacionalmente y menos propensos a hacer *over-fitting* que las técnicas *wrapper*. La principal limitación con estos

métodos es que toma decisiones según el clasificador. Por lo tanto, la selección de características puede verse afectada por la hipótesis que haga el clasificador, que podría no funcionar con otros clasificadores (Z.M. y D.F., 2015).

Finalmente, como se ha comentado, aparecen los algoritmos *híbridos*. Estos, combinan algoritmos de *filtrado* y *wrapper*, de manera que se reduce el coste computacional y los tiempos de ejecución. Este tipo de enfoques han recibido gran atención en los últimos años.

Dada la extensión del campo de estudio, en este documento nos vamos a centrar en los algoritmos de tipo *wrapper*, y más específicamente, en aquellos basados en metaheurísticas para realizar la selección. Se hará un breve repaso de las propuestas de los últimos dos años.

2. Estado del arte

Tal y como se ha explicado, los algoritmos de selección de características tipo *wrapper* utilizan diferentes metodologías para optimizar la precisión de un algoritmo de inducción. Para ello, seleccionan subconjuntos de variables y comprueban los resultados obtenidos por el algoritmo de aprendizaje. De esta manera van iterando para obtener la precisión óptima.

Las metaheurísticas generalmente simulan el comportamiento de la naturaleza. Cuando se trata de un conjunto de individuos (como por ejemplo las Colonias de hormigas), hablamos de enjambre (del inglés, *swarm*). Por otro lado, están las Metaheurísticas que hacen referencia a un único individuo (por ejemplo, el Recocido simulado).

En el caso de metaheurísticas adaptadas para el problema de FS, generalmente hablamos de aquellas que usan una población o enjambre, conocidas como *Swarm Intelligence* o SI (Beni y Wang, 1993). Este tipo de algoritmos ha tenido un crecimiento exponencial en las últimas dos décadas, tal y como se puede ver en la Figura 1. Esto se debe a sus buenos resultados y a la gran variedad de nuevas metaheurísticas propuestas (Brezočnik, Fister y Podgorelec, 2018). Sin embargo, tal y como veremos más adelante, también se han aplicado otro tipo de metaheurísticas con una única solución parcial.

Algunas las propuestas más recientes se han utilizado en casos de trabajo reales. Por ejemplo, el algoritmo Grey Wolf Optimization (GWO) se ha utilizado para extraer las características más relevantes en *datasets* de enfermedades arteriales (Al-tashi, Rais y Jadid, 2019). Recientemente, también, se ha utilizado Harmony Search Algorithm (HSA), otro algoritmo evolutivo poblacional, para la identificación de documentos manuscritos.

2.1. Swarm Intelligence: metodología y adaptación a FS

Generalmente al adaptar una metaheurística para un problema que se quiere resolver, se llevan a cabo los siguientes pasos: modelar la representación de los agentes, inicialización de la población, definición de la condición de parada y, por último, la función de aptitud o fitness para poder evaluar las soluciones parciales. Además de todo esto, generalmente se añaden funciones auxiliares en base a cada algoritmo para conseguir nuevos y diferentes comportamientos de estos.

En primer lugar, la representación de los agentes consiste en modelar a los individuos de tal manera que representen las diferentes variables del entorno es decir, las dimensiones. Generalmente, para adaptar estas configuraciones de los individuos, se utiliza una representación binaria (Vieira y col., 2013; Emary, Zawbaa y Hassanien, 2016), donde 1 significa que cogemos la variable para el subconjunto de características, y 0 que no la cogemos. Para ello, y para poder calcular la nueva posición o movimiento de la población, se utilizan funciones de transferencia, generalmente la función Sigmoidal y la tangente hiperbólica. Por ejemplo, en BGOA (Mafarja y col., 2019), se utilizan estas dos funciones para convertir el movimiento en una probabilidad y así poder asignar un 1 o un 0 a la dimensión en particular que estamos tratando.

La función de inicialización, por otro lado, genera de manera aleatoria la población de subconjuntos de características que se irán ejecutando sobre el algoritmo de inducción. La función de aptitud permite valorar la precisión de un posible subconjunto, de modo que esta será la función a optimizar. Generalmente se utiliza la precisión (en inglés, *accuracy*) obtenida con el algoritmo de aprendizaje. Por

último, la condición de parada hace referencia: al umbral bajo el cual damos por válido el subconjunto seleccionado y no se seguirá optimizando (Packianather y Kapoor, 2015); o a un tiempo máximo en caso de tener un tiempo límite para el cálculo; o a un número concreto de iteraciones (Mohammadi y Abadeh, 2014).

2.2. Swarm Intelligence: nuevas propuestas

Dada la gran cantidad de metaheurísticas propuestas en las dos últimas décadas al problema de FS con SI (tal y como se muestra en la Figura 2), y al número de revisiones al respecto (generalmente centradas en aplicaciones de los algoritmos), las propuestas que se van a mencionar son referentes a los últimos dos años.

Generalmente, todas ellas simulan el comportamiento de un grupo de insectos o animales que se coordinan para explorar el entorno (espacio de soluciones) y encontrar la comida (o en otros casos, cazar a la presa). En función de como se explore el espacio de soluciones, nos encontramos con varios tipos:

En primer lugar, aquellos metahurísticas que simulan el comportamiento de especies que se organizan de una manera determinada para moverse y buscar comida. Estos métodos solo tienen en cuenta la precisión del mejor individuo y la posición de el resto. Una de las propuestas más recientes ha sido Binary Salp Swarm Algorithm (BSSA) (Faris y col., 2018), en el cual los individuos “flotan” en forma de espiral, tomando como centro de la espiral la mejor solución y a partir de ella, el resto forman a su alrededor. Otras propuestas que siguen esta estructura son Binary Dragonfly Optimization Algorithm (BDOA) (Mafarja y col., 2018) y Binary Grasshopper Optimization Algorithm (BGOA) (Mafarja y col., 2019), en las cuales, la población se distribuye de manera uniforme por la superficie teniendo en cuenta la posición de los demás y la posición de la mejor solución.

En segundo lugar, los algoritmos basados en *random walk*, es decir, para cada individuo, seleccionar otro (en base a ciertos criterios) y realizar el movimiento con respecto a este último. Dos propuestas recientes basadas en esta idea son: Binary Crow Search Algorithm (BCSA) (De Souza y col., 2018) y Leavy Antlion Optimization (LALO) (De Souza y col., 2018), en las cuales, cada individuo se mueve respecto de otro. Por ejemplo, en el caso de LALO, las mejores (en base al *fitness* obtenido) hormigas serán hormigas león y estarán estáticas, mientras que las peores, serán hormigas normales, y se moverán en torno a las primeras.

En tercer y último lugar están aquellos algoritmos que simulan especies que desprenden olores o feromonas en base a la calidad de cada individuo. Este es el caso de Binary Butterfly Optimization Algorithm (BBOA) (Arora y Anand, 2019) y Ant Colony Optimization (ACO) modificado (Jameel y Rehman, 2018), en los cuales, en función de la precisión obtenida con cada individuo (cada posible solución), se desprende una mayor cantidad de fragancia o feromonas, respectivamente. De esta manera, el resto de los individuos tienden a ir por donde van las mejores soluciones.

Además de estas, ha habido otras propuestas basadas en SI con comportamientos completamente diferentes. Este es el caso de Brain Storm Optimization (BSO) (Zhang y col., 2018) y Tunable Particle Swarm Optimization (TPSO) (Mallenahalli y Sarma, 2018). En el caso de BSO, por ejemplo, se simula el comportamiento del ser humano, agrupando los individuos en *clusters* y generando nuevas soluciones a partir de ellos.

2.3. Otras propuestas basadas en Metaheurísticas

Además de las propuestas basadas en SI, ha habido gran variedad de novedades en el campo de FS con metaheurísticas. Por un lado, se han propuesto nuevos algoritmos híbridos donde la componente wrapper de los mismos es una metaheurística poblacional. Este es el caso del Binary Cuckoo Search (BCS) modificado (Jiang y col., 2018), que adapta la propuesta original de Cuckoo Search (Yang y Deb, 2009) y de BCS (Gunavathi y Premalatha, 2015) a un modelo híbrido. En este se combinan el algoritmo de filtrado Mutual Information Feature Selection junto con BCS para obtener el mejor subconjunto de características en un menor tiempo y de manera más precisa que un algoritmo únicamente de filtrado.

Por otro lado, se han propuesto metahurísticas no poblacionales para la resolución de problemas de FS. En una de las propuestas más recientes, Costa y col., 2018 propusieron utilizar VNS (del inglés, Variable Neighborhood Search) como algoritmo de optimización para seleccionar el mejor subconjunto de variables. En esta propuesta, el algoritmo VNS va cambiando dinámicamente los vecindarios para evitar quedar en mínimos locales. Los resultados demuestran que, aunque no esta basado en SI, se obtiene una muy buena precisión de los algoritmos de inducción.

3. Conclusiones y trabajo futuro

El estado del arte muestra que el uso de metaheurísticas para el problema de selección de características ha sido ampliamente investigado y se han obtenido muy buenos resultados, mejorando en su mayoría al resto de métodos de FS propuestos en la literatura. Además, se ha remarcado la eficiencia de los algoritmos de Swarm Intelligence, que han aumentado exponencialmente en los últimos años y han demostrado una gran precisión en la resolución de este tipo de problemas.

Las líneas de trabajo abiertas serían, mayormente, el uso de otros tipos de metaheurísticas no basadas en enjambres, que también han demostrado una gran eficiencia en la resolución de problemas de optimización y la adaptación de los métodos ya propuestos a opciones híbridas donde se mejoren los tiempos de ejecución, ya que las metaheurísticas, al ser generalmente métodos *wrapper*, requieren un mayor tiempo de computo.

Appendices

A. Apéndice I: Aumento de publicaciones

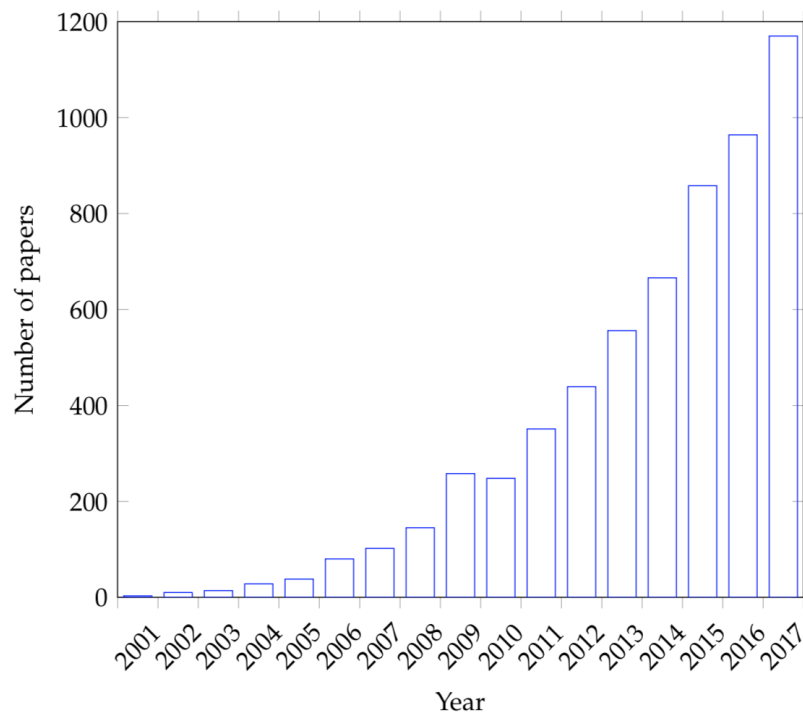


Figura 1: Metaheurísticas aplicadas a FS: Aumento de las publicaciones en los últimos años

B. Apéndice II: Nuevas metaheurísticas en los últimos 20 años

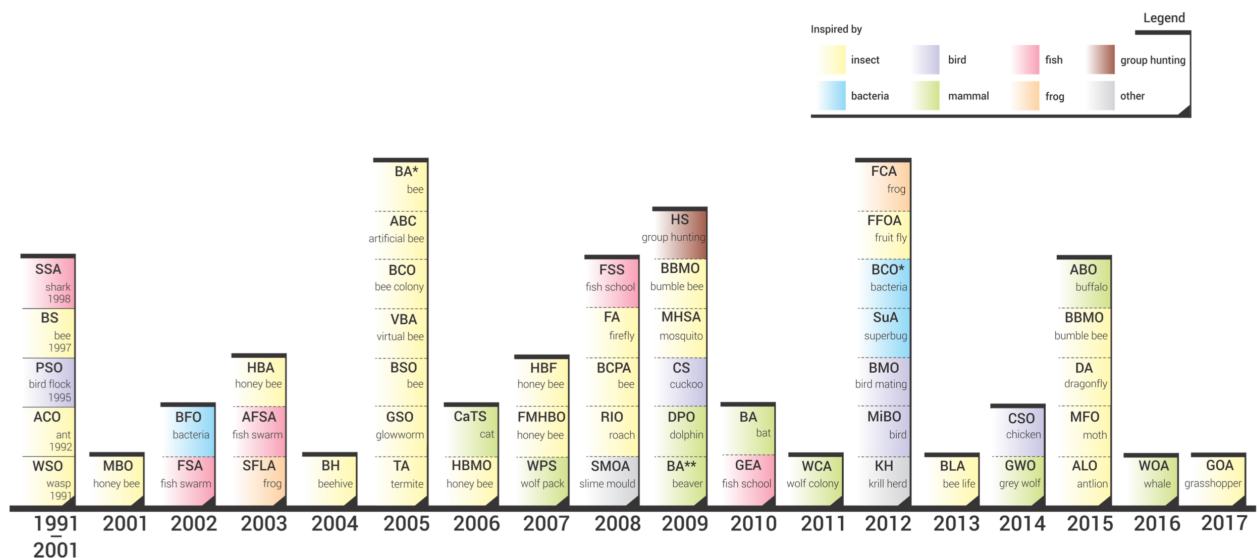


Figura 2: Aparición de las nuevas metaheurísticas en los últimos años

Referencias

- Al-tashi, Qasem, Helmi Rais y Said Jadid (2019). *Feature Selection Method Based on Grey Wolf Optimization for Coronary Artery Disease Classification*. Vol. 843. Springer International Publishing, págs. 257-266. ISBN: 978-3-319-99006-4. DOI: 10.1007/978-3-319-99007-1. URL: <http://dx.doi.org/10.1007/978-3-319-99007-1>. DOI: 10.1007/978-3-319-99007-1.
- Ang, Jun Chin y col. (2016). “Supervised, unsupervised, and semi-supervised feature selection: A review on gene selection”. En: *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 13.5, págs. 971-989. ISSN: 15455963. DOI: 10.1109/TCBB.2015.2478454.
- Arora, Sankalap y Priyanka Anand (2019). “Binary butterfly optimization approaches for feature selection”. En: *Expert Systems with Applications* 116, págs. 147-160. ISSN: 09574174. DOI: 10.1016/j.eswa.2018.08.051. URL: <https://doi.org/10.1016/j.eswa.2018.08.051>.
- Beni, Gerardo y Jing Wang (1993). “Swarm Intelligence in Cellular Robotic Systems”. En: *Robots and Biological Systems: Towards a New Bionics?* 1.2, págs. 703-712. URL: <http://link.springer.com/10.1007/978-3-642-58069-7>.
- Brezočnik, Lucija, Iztok Fister y Vili Podgorelec (2018). “Swarm Intelligence Algorithms for Feature Selection: A Review”. En: *Applied Sciences* 8.9, pág. 1521. ISSN: 2076-3417. DOI: 10.3390/app8091521. URL: <http://www.mdpi.com/2076-3417/8/9/1521>.
- Costa, H y col. (2018). “A VNS algorithm for feature selection in hierarchical classification context”. En: *Electronic Notes in Discrete Mathematics* 66, págs. 79-86. ISSN: 15710653. DOI: 10.1016/j.endm.2018.03.011.
- De Souza, Rodrigo Clemente Thom y col. (2018). “A V-Shaped Binary Crow Search Algorithm for Feature Selection”. En: *2018 IEEE Congress on Evolutionary Computation, CEC 2018 - Proceedings*. IEEE, págs. 1-8. ISBN: 9781509060177. DOI: 10.1109/CEC.2018.8477975.
- Emary, E., Hossam M. Zawbaa y Aboul Ella Hassanien (2016). “Binary grey wolf optimization approaches for feature selection”. En: *Neurocomputing* 172, págs. 371-381. ISSN: 18728286. DOI: 10.1016/j.neucom.2015.06.083. URL: <http://dx.doi.org/10.1016/j.neucom.2015.06.083>.
- Erguzel, Turker Tekin y col. (2014). “Ant colony optimization based feature selection method for QEEG data classification”. En: *Psychiatry Investigation* 11.3, págs. 243-250. ISSN: 19763026. DOI: 10.4306/pi.2014.11.3.243.
- Faris, Hossam y col. (2018). “An efficient binary Salp Swarm Algorithm with crossover scheme for feature selection problems”. En: *Knowledge-Based Systems* 154.March, págs. 43-67. ISSN: 09507051. DOI: 10.1016/j.knosys.2018.05.009. URL: <https://doi.org/10.1016/j.knosys.2018.05.009>.
- Gunavathi, C. y K. Premalatha (2015). “Cuckoo search optimisation for feature selection in cancer classification: a new approach”. En: *International Journal of Data Mining and Bioinformatics* 13.3, pág. 248. ISSN: 1748-5673. DOI: 10.1504/IJDMB.2015.072092. URL: <http://www.inderscience.com/link.php?id=72092>.
- Jain, Divya y Vijendra Singh (2018). “Feature selection and classification systems for chronic disease prediction: A review”. En: *Egyptian Informatics Journal* 19.3, págs. 179-189. ISSN: 11108665. DOI: 10.1016/j.eij.2018.03.002. URL: <https://doi.org/10.1016/j.eij.2018.03.002>.
- Jameel, Saba y Saif Ur Rehman (2018). “An optimal feature selection method using modified wrapper-based ant colony optimisation”. En: *J.Natn.Sci.Foundation Sri Lanka* 46.2, págs. 143-151.
- Jiang, Yun y col. (2018). “Modified Binary Cuckoo Search for Feature Selection: A Hybrid Filter-Wrapper Approach”. En: *Proceedings - 13th International Conference on Computational Intelligence and Security, CIS 2017* 2018-January.2, págs. 488-491. DOI: 10.1109/CIS.2017.00113.

- Liu H.; Motoda, H (2007). “Computational Methods of Feature Selection”. En: *CRC Press*.
- Mafarja, Majdi y col. (2019). “Binary grasshopper optimisation algorithm approaches for feature selection problems”. En: *Expert Systems with Applications* 117, págs. 267-286. ISSN: 09574174. DOI: 10.1016/j.eswa.2018.09.015. URL: <https://doi.org/10.1016/j.eswa.2018.09.015>.
- Mafarja, Majdi M. y col. (2018). “Binary Dragonfly Algorithm for Feature Selection”. En: *Proceedings - 2017 International Conference on New Trends in Computing Sciences, ICTCS 2017*. Vol. 2018-Janua, págs. 12-17. ISBN: 9781538605271. DOI: 10.1109/ICTCS.2017.43.
- Mallenahalli, Naresh y T Hitendra Sarma (2018). “A Tunable Particle Swarm Size Optimization Algorithm for Feature Selection”. En: *2018 IEEE Congress on Evolutionary Computation (CEC)*, págs. 1-7.
- Mohammadi, F. Ghareh y M. Saniee Abadeh (2014). “Image steganalysis using a bee colony based feature selection algorithm”. En: *Engineering Applications of Artificial Intelligence* 31, págs. 35-43. ISSN: 09521976. DOI: 10.1016/j.engappai.2013.09.016. URL: <http://dx.doi.org/10.1016/j.engappai.2013.09.016>.
- Packianather, Michael S. y Bharat Kapoor (2015). “A wrapper-based feature selection approach using Bees Algorithm for a wood defect classification system”. En: *2015 10th System of Systems Engineering Conference, SoSE 2015*, págs. 498-503. DOI: 10.1109/SYSOSE.2015.7151902.
- Vieira, Susana M. y col. (2013). “Modified binary PSO for feature selection using SVM applied to mortality prediction of septic patients”. En: *Applied Soft Computing Journal* 13.8, págs. 3494-3504. ISSN: 15684946. DOI: 10.1016/j.asoc.2013.03.021.
- Yang, Xin-She y Suash Deb (2009). “Cuckoo Search via Levey Flights”. En: *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC 2009)*, págs. 210-214. ISSN: 2164-7364. DOI: 10.1109/NABIC.2009.5393690. arXiv: 1003.1594.
- Zhang, Xu-tao y col. (2018). *A Wrapper Feature Selection Algorithm Based on Brain Storm Optimization*. Vol. 952. Springer Singapore, págs. 308-315. ISBN: 978-981-13-2828-2. DOI: 10.1007/978-981-13-2829-9. URL: <http://dx.doi.org/10.1007/978-981-13-2829-9> <http://link.springer.com/10.1007/978-981-13-2829-9>.
- Z.M., Hira y Gillies D.F. (2015). “A review of feature selection and feature extraction methods applied on microarray data”. En: *Advances in Bioinformatics* 2015.1. DOI: 10.1155/2015/198363.