Universidad de La Laguna

Algoritmos para la detección de comunidades en redes

Trabajo de fin de grado

Grado en Ingeniería Informática 2018/19

Autor: Miguel Jiménez Gomis

Universidad de La Laguna

Índice

Universidad de La Laguna

1. Resumen
2. Introducción y descripción del problema
3. Antecedentes
4. Algoritmo propuesto
5. Experiencia computacional
6. Summary and conclusions
7. Presupuesto
8. Bibliografía
9. Preguntas

Universidad de La Laguna

1. Resumen

En este trabajo, se propone un nuevo algoritmo para la detección de comunidades en redes que permita realizar asignaciones rápidas a las diferentes comunidades existentes en el grafo, cuando se tratan con datos en tiempo real. El objetivo de este algoritmo es reducir la carga computacional de estos análisis y tener soluciones válidas aproximadas en el menor tiempo, haciendo posible la identificación de los datos existentes en el solapamiento de diferentes comunidades para tenerlos en cuenta y obtener mejores resultados.

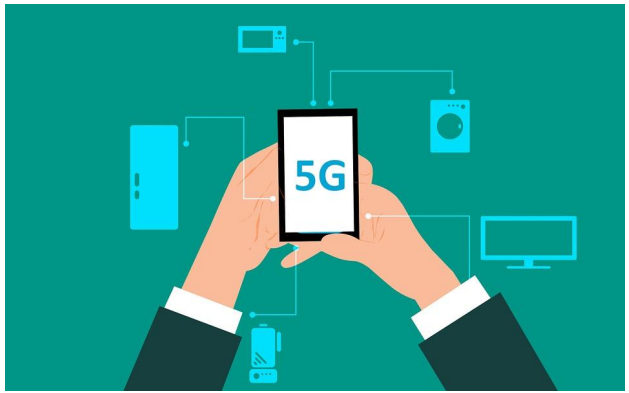
Para comprobar la viabilidad del algoritmo propuesto se realizó una experimentación sobre data set reales, para poder compararlos de manera sencilla con la literatura; y con un conjunto de datos artificial, con el objetivo de que el análisis fuera más amplio.

La experimentación realizada y análisis previos llevados a cabo en este proyecto establecen unas bases sólidas sobre las que seguir trabajando para completar y mejorar el algoritmo propuesto.

Universidad de La Laguna

2. Introducción y descripción del problema

Universidad de La Laguna



[1]

Hitos

1. Revisión bibliográfica de la literatura relativa a las técnicas y campos de aplicación de la detección de comunidades en redes complejas
2. Diseño e implementación de al menos una técnica de detección de comunidades solapadas en redes estáticas
3. Diseño de experimentos y comparativa con la literatura
4. Desarrollo de herramienta para representación gráfica de soluciones
5. Diseño e implementación de al menos una técnica de detección de comunidades solapadas en redes dinámicas

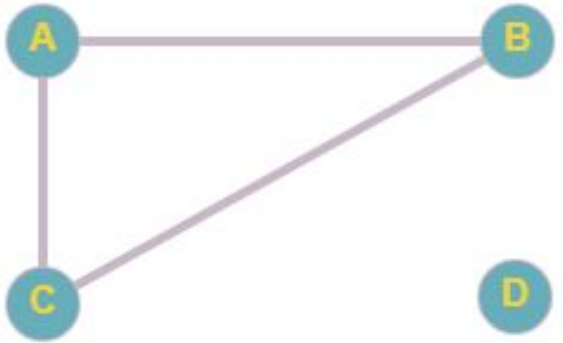
Universidad de La Laguna

3. Antecedentes

Universidad de La Laguna

Grafos

G = {V, E}
V = {A,B,C,D}
E = { {A,B},{B,C},{C,A} }

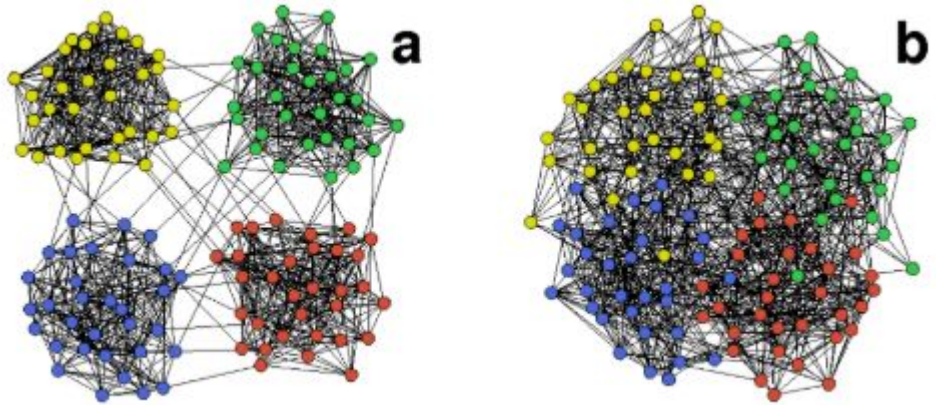


[2]

Universidad de La Laguna

Comunidades en grafos

- Grado interno
- Grado interno promedio
- Densidad de aristas internas
- Grado externo
- Grado externo promedio
- Densidad de aristas externas



[3]

Universidad de La Laguna

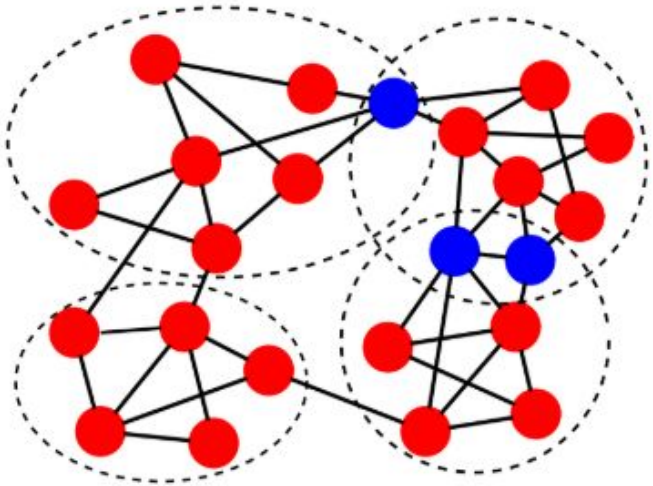
Comunidades solapadas

Modularidad:
M =∑(i) [(ls/L) - (ds/2L)^2]

Grado promedio:
AD(c)=2*(|E(c)|/|c|)

Estructura local:
Γ(v) = {w e V|(v,w) e E} U {v}

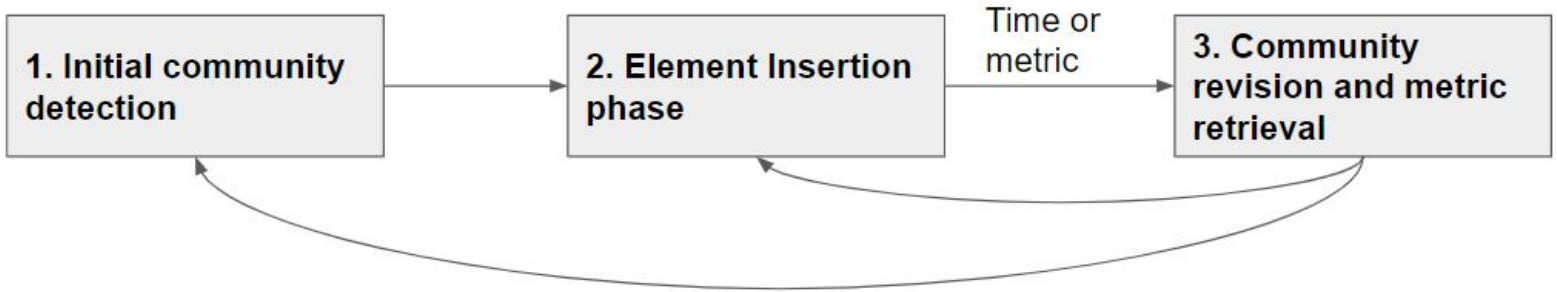
Equivalencia estructural:
σ(u,v)=(|Γ(u) Γ(v)|)/√(|Γ(u)|*|Γ(v)|)



[4][5]

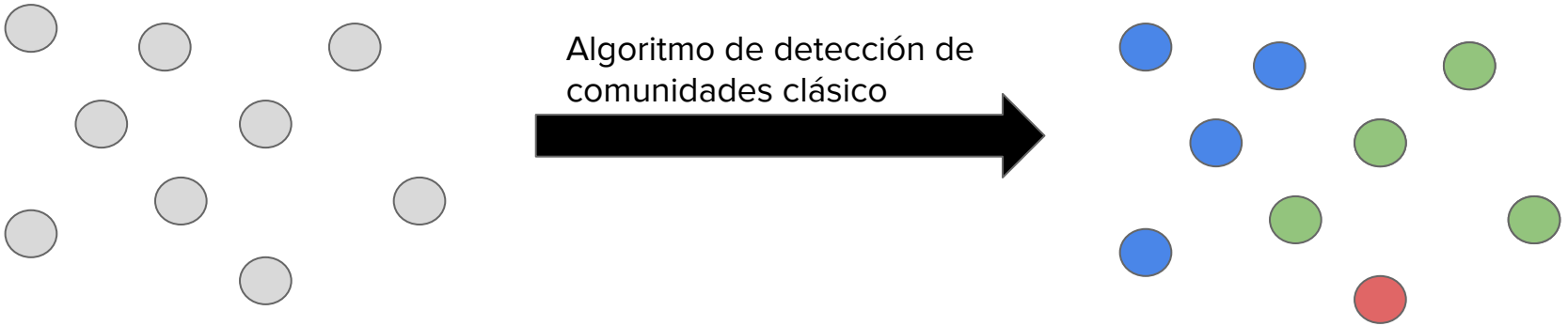
Universidad de La Laguna

4. Algoritmo propuesto


1. Initial community detection
2. Element Insertion phase
Time or metric
3. Community revision and metric retrieval

Universidad de La Laguna

Fase 1. Detección de comunidades iniciales



Algoritmos clásicos:

- CPM
- CONGA
- LED
- Fuzzy clustering
- Girvan–Newman

[6][7][8][9][10]

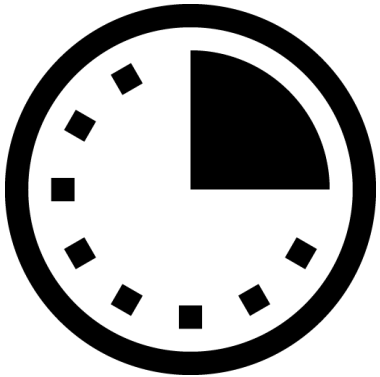
Universidad de La Laguna

Fase 2. Inserción de elementos

Pseudocódigo de la inserción por medio de Modularidad.
Require graph G, , initialCommunities, error (%) 1. maxModularity = 0; 2. insertions = []; 3. For each community in initialCommunities: 4.calculate modularity of data; 5. If datamodularity >= max Modularity * (error/100) : 6. insertions.add community; 7. End Loop; 8. For each community in insertions: 9 add data to community; 10. End Loop;

Universidad de La Laguna

Fase 3. Revisión de comunidades y obtención de métricas



Universidad de La Laguna

5. Experiencia computacional

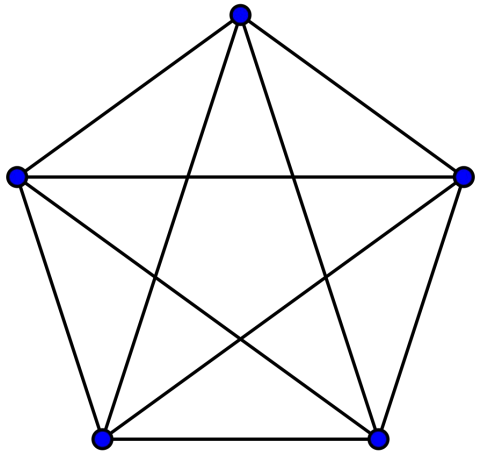
Universidad de La Laguna

Diseño de la experimentación

Test artificial

Test con datos reales

- Karate Club Dataset
- College football Dataset



[11][12][13]

Universidad de La Laguna

Algoritmo LED (Loop Edge Detection)

Pseudocódigo del algoritmo LED. Primera parte.

Algorithm 1. LED algorithm.
Require: Graph G, α
Ensure: Clusters Ci; where i(1; 2; ...; k).
1: curG=G; //pointer to current graph
2: edgeListi; //store the ith deleted edges
3: graphListti;//store the ith graph
//Step1 and Step2: Calculating Structural Similarity and Deleting edges less than α
4: repeat
5: for each edge in curG do
6: Calculate Structural Similarity;
7: end for
8: push curG into graphList;
9: copy curG to G';
10: if w(e) < α then
11: curG = curG - {e};
12: edgeListi = edgeListi + (e);

Pseudocódigo del algoritmo LED. Segunda parte.

13: end if
14: until edgeListi is not empty
15: label each inter-connected-components as an initial community Ci;
//Step3: Identify Overlapping Vertices
16: for each edgeListi in reverse order do
17: if two vertices of the edge in different initial communities then
18: judge the vertices of the edge is overlapping or not using AD(c) (formula(4));
19: end if
20: end for
//extend initial communities
21: attach the isolated vertices to the one initial community with highest Structural Similarity; return Clusters;

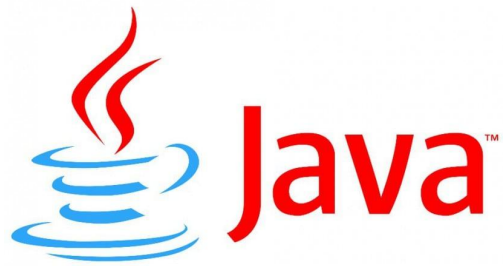
Universidad de La Laguna

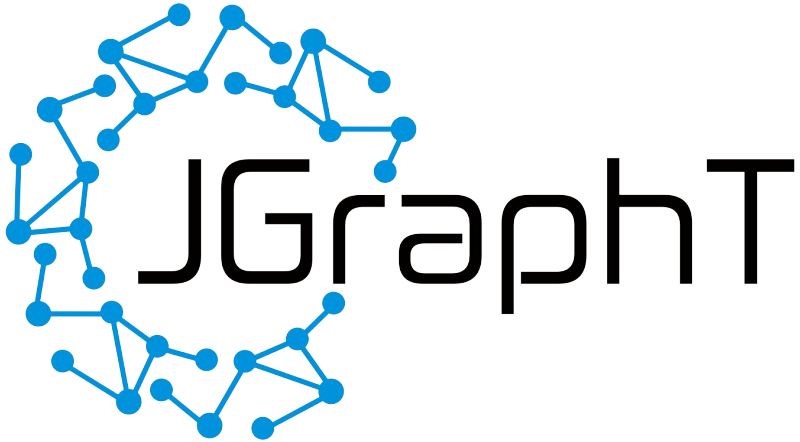
Algoritmo LED (Loop Edge Detection)

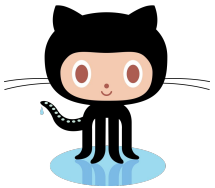
Modificación de LED. El código de esta tabla sustituye las líneas desde la 4 hasta la 14 del pseudocódigo del algoritmo original.
4: def recursive (graph, α): 5: Calculate Structural Similarity; 6. Delete edges; 7. Check if G.inter-connected-components == 1: 8: Mark G as initial community; 9. Else: 10: For each inter-connected-components in G: 11. recursive(inter-connected-components, α); 12. end IF;

Universidad de La Laguna

Implementación de la experiencia computacional





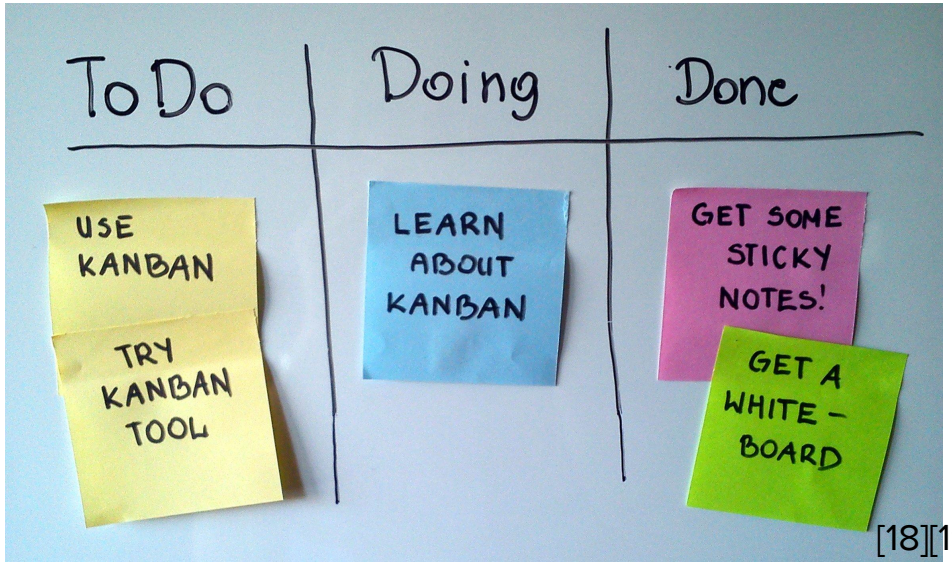


[14][15][16][17]

Universidad de La Laguna

Herramienta para la representación gráfica de soluciones y Metodología de desarrollo

jgraph
connecting the dots



[18][19]

Universidad de La Laguna

Resultados de la experiencia computacional

Ejemplo de Output

Karate: 50%	[Init:0.08584043421635067	0:0.09768049410826109	1:0.09399468539381972	2:0.10341814395936527	3:0.09965771865291029
	4:0.11043152607484653	5:0.10651111309137723	6:0.0976627260230579	7:0.09292490191673984	8:0.10927017858449642
	9:0.10583776684667907	10:0.12545129759045792	11:0.1257536841177242	12:0.13799999999999998	13:0.1417270012496024
	14:0.1640586899713655	15:0.2286394456622278	16:0.33645316800701813]		
Football: 90%	[Init:0.05455563624354258	0:0.05777997107911524	1:0.059904	2:0.06100027142427686	3:0.0639940164642304
	4:0.0662735500056825	5:0.06967257664383456	6:0.07349992822276138	7:0.0745844747787063	8:0.0807056482930449
	9:0.0458209346501329	10:0.05038561318926301	11:0.06295292548589213]		

Universidad de La Laguna

Resultados de la experiencia computacional

Ejemplo con datos reales

Karate club dataset			
n nodos	34		
% de nodos iniciales	**Mod inicial**	**Numero de inserciones**	**Mod final**
10	0.01776008983786565	30	0.3516153268219384
30	0.062160314432529784	23	0.3462234462175538
50	0.08584043421635067	17	0.33645316800701813
70	0.10360052405421631	10	0.3367305503318155
90	0.1332006737839924	3	0.32143477176559476
College footbal			
n nodos	115		
% de nodos iniciales	**Mod inicial**	**Numero de inserciones**	**Mod final**
10	0.002899091781057818	119	0.8515159050521792
30	0.016472112392373967	93	0.6236254483254857
50	0.030835794398524064	66	0.428625145345822
70	0.044408815009840216	37	0.2251049497997793
90	0.05455563624354258	12	0.06295292548589213

Universidad de La Laguna

Resultados de la experiencia computacional

Comparativa con la literatura

	Girvan–Newman	LED	Algoritmo Propuesto
Karate Club	0.4012	0.4155	0.3384
College Football	0.5995	0.5995	0.4383

Universidad de La Laguna

Resultados de la experiencia computacional

Matriz del incremento de la modularidad promedio

Numero de clusters	% inicial de nodos							
	10	20	30	40	50	60	70	80
2	0,35121973	0,272680517	0,11062632	0,088436272	0,058642505	0,0491006	0,03337	0,02377
3	0,372644041	0,272398667	0,08058366	0,057857441	0,038876257	0,03228378	0,02201	0,01562
4	0,340143973	0,193151157	0,05660807	0,041107448	0,028420146	0,02426021	0,01638	0,01166
5	0,307157836	0,178992724	0,0412227	0,032354638	0,022572885	0,01876108	0,01275	0,00911
6	0,223209366	0,15335754	0,02378096	0,024614851	0,018058249	0,01495072	0,0102	0,00728
7	0,180863377	0,051130252	0,01981867	0,021989164	0,014874251	0,01231747	0,0084	0,00599
8	0,156568514	0,063063257	0,01615324	0,01613298	0,012528755	0,01037929	0,0071	0,00505
9	0,156112422		0,01486007	0,01434122	0,010755671	0,00890281	0,00611	0,00435

Matriz de modularidad inicial

Numero de clusters	% inicial de nodos							
	10	20	30	40	50	60	70	80
2	0,107122749	0,11703904	0,12781469	0,130175164	0,134100991	0,13482386	0,13624	0,13656
3	0,059715596	0,065052993	0,0723227	0,076071651	0,081504793	0,08375222	0,087	0,08854
4	0,038974462	0,042513888	0,04767701	0,051195456	0,056068253	0,06004258	0,0637	0,06547
5	0,02839904	0,031177625	0,0353245	0,038360337	0,042742841	0,04492286	0,04822	0,04993
6	0,021523372	0,023695537	0,026969	0,029562587	0,03325717	0,03524671	0,03816	0,03972
7	0,017004724	0,018770975	0,02143279	0,02365335	0,026824995	0,02857633	0,03117	0,03259
8	0,013861208	0,015332056	0,01755827	0,019478385	0,022228271	0,02377479	0,02606	0,02737
9	0,011577105	0,012821321	0,01471618	0,016390528	0,018795391	0,02018303	0,02221	0,0234

Universidad de La Laguna

6. Summary and conclusions

Universidad de La Laguna

Conclusions

- Community detection is a computationally demanding task.
- Promising results for the provisional assignment task.
- The main source of error is the initial community detection.
- A first version of this algorithm could be implemented as an API.
- More thorough experimentation and investigation must be conducted.

Future Work

- Deepen the analysis and investigation of this algorithm to improve it.

- Implement a usable version of the algorithm.

Universidad de La Laguna

7. Presupuesto

Costes del proyecto	
Número de horas trabajadas	350
Coste por hora	10 €
Total	3.500 €

Universidad de La Laguna

Bibliografía

[1] A. Katal, M. Wazid and R. H. Goudar, "Big data: Issues, challenges, tools and Good practices," 2013 Sixth International Conference on Contemporary Computing (IC3), Noida, 2013, pp. 404-409. doi: 10.1109/IC3.2013.6612229

[2] Trudeau, Richard J. Introduction to Graph Theory. New York: Dover, 1993. Print.

[3] Santo Fortunato. Community detection in networks: A user guide

[4] M. Girvan, M.E. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. 99 (12) (2002) 7821–7826.

[5] Y. Cai, C. Shi, Y. Dong, Q. Ke, B. Wu, A novel genetic algorithm for overlapping community detection, in: Advanced Data Mining and Applications, Springer, Beijing, China, 2011, pp. 97–108.

[6] H. Sun, J. Huang, X. Zhang, et al., IncOrder: incremental density-based community detection in dynamic networks, Knowl.-Based Syst. 72 (2014) 1–12.

[7] M.E. Newman, M. Girvan, Finding and evaluating community structure in networks, Phys. Rev. E 69 (2) (2004) 026113.

[8] S. Zhang, R.-S. Wang, X.-S. Zhang, Identification of overlapping community structure in complex networks using fuzzy c-means clustering, Phys. A: Stat. Mech. Appl. 374 (1) (2007) 483–490.

[9] T. Ma, Y. Wang, M. Tang, J. Cao, Y. Tian
Abdullah al-dhelaan, mznah al-rodhaan, LED: a fast overlapping communities detection algorithm based on structural clustering
Neurocomputing, 207 (2016), pp. 488-500

[10] Girvan M. and Newman M. E. J., Community structure in social and biological networks, Proc. Natl. Acad. Sci. USA 99, 7821–7826 (2002)
[11] Alba, Richard D. (1973), "A graph-theoretic definition of a sociometric clique" (PDF), Journal of Mathematical Sociology, 3 (1): 113–126, doi:10.1080/0022250X.1973.9989826.
[12] Zachary's karate club: social network of friendships between 34 members of a karate club at a US university in the 1970s. Please cite W. W. Zachary, An information flow model for conflict and fission in small groups, Journal of Anthropological Research 33, 452-473 (1977).
[13] American College football: network of American football games between Division IA colleges during regular season Fall 2000. Please cite M. Girvan and M. E. J. Newman, Proc. Natl. Acad. Sci. USA 99, 7821-7826 (2002).
[14] Java: https://www.java.com/
[15] Github: https://github.com/
[16] Eclipse: https://www.eclipse.org/
[17] JGraphT: https://jgrapht.org/
[18] JGraph : https://github.com/jgraph/
[19] Graph Strem : http://graphstream-project.org/
[20] Weka. https://www.cs.waikato.ac.nz/ml/weka/ (last visited on July 2019)
[21] https://www.indeed.es/salaries/Investigador/a-Salaries

Universidad de La Laguna

¿Preguntas?

https://github.com/MiguelJG/TFG