Network group discovery framework

Lovro Šubelj

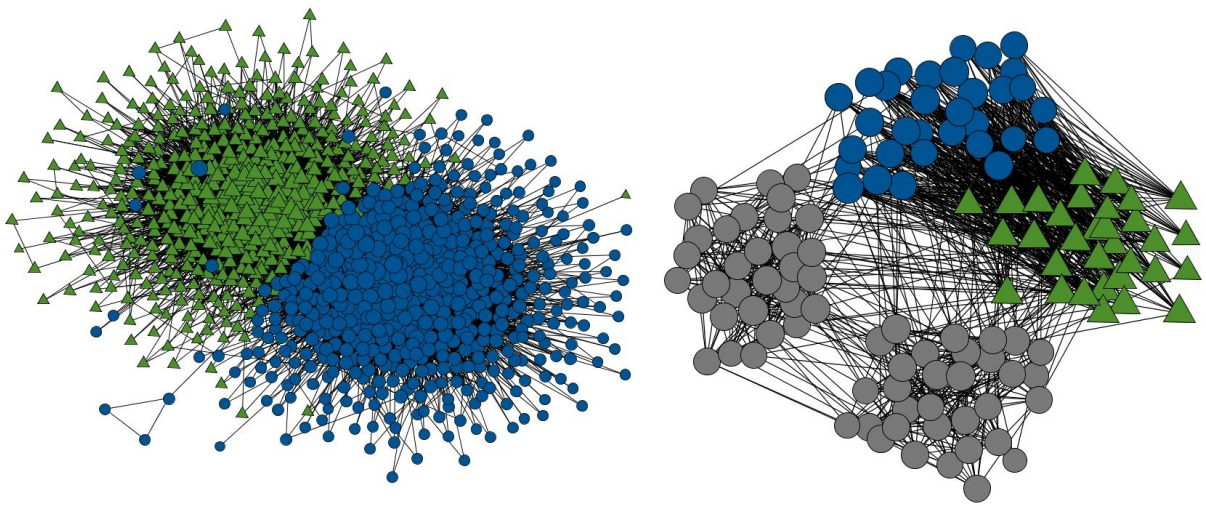
University of Ljubljana,
Faculty of Computer and Information Science

9.7.2014

Groups in real-world networks

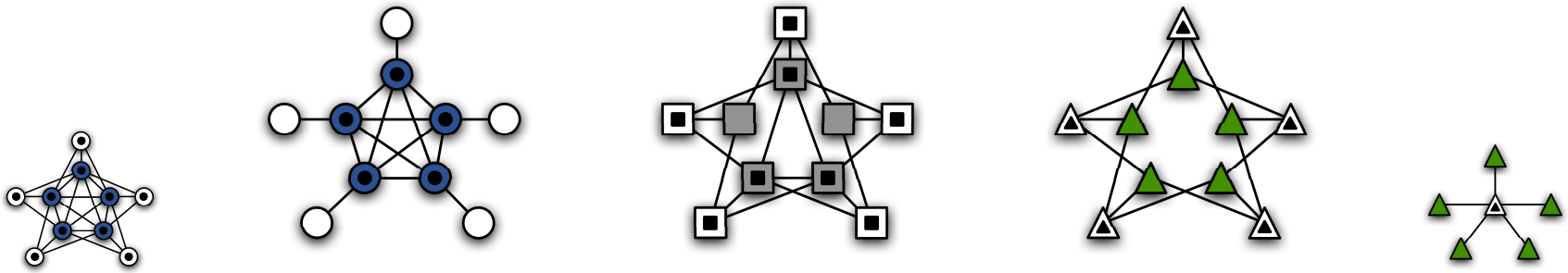
community densely linked nodes that are sparsely linked between
(or dense groups of sparse graphs) Girvan and Newman (2002)

module nodes linked to similar other nodes Newman and Leicht (2007)
(or groups with similar linking pattern)



Group type formalism

Let S be a group (filled) & T its linking pattern (marked). Šubelj et al. (2013a)



Community ($S = T$) *Mixture* ($S \approx T$) *Module* ($S \neq T$)

Let $\tau_{S,T}$ be a parameter of group S & its pattern T.

$$\tau_{S,T} = \frac{|S \cap T|}{|S \cup T|}$$

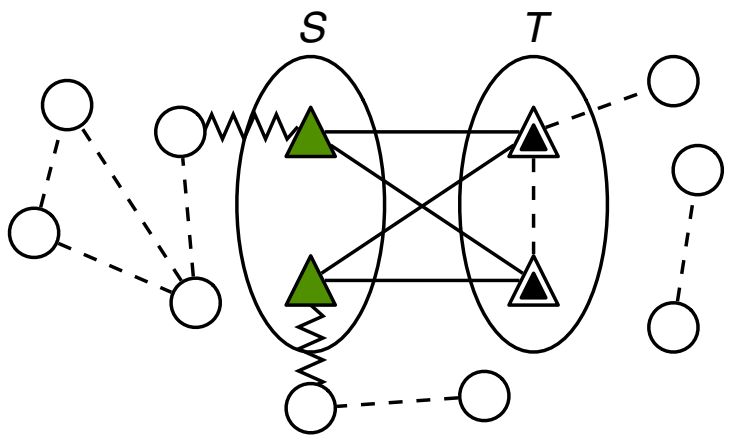
$\tau = 1$ for communities, $\tau \approx \frac{1}{2}$ for mixtures & $\tau = 0$ for modules.

Group quality criterion

Let $L_{S,T}$ be a number of links between S & T.

$$W_{S,T} = \ldots \left(\frac{L_{S,T}}{|S||T|} - \frac{L_{S,T^C}}{|S||T^C|} \right)$$ Šubelj et al. (2013a)

A local asymmetric criterion that *favors links* in (S, T) & *penalizes for links* in (S, T^C). Consistent with wide class of models for $S = T$. Zhao et al. (2011)



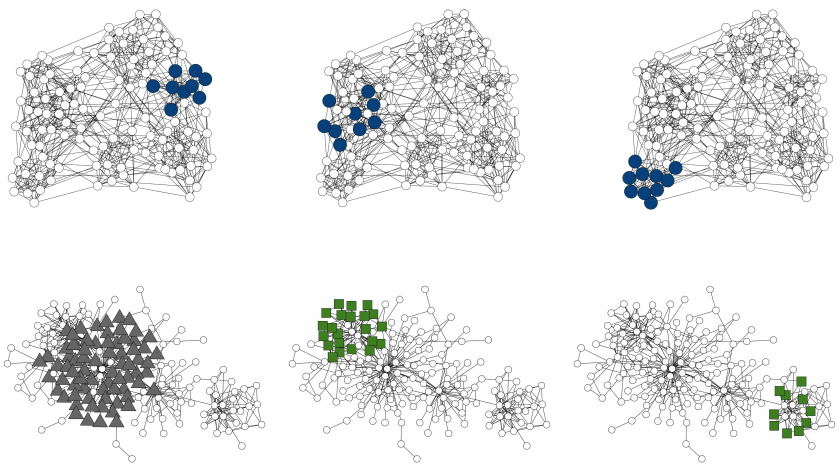
Group discovery by extraction

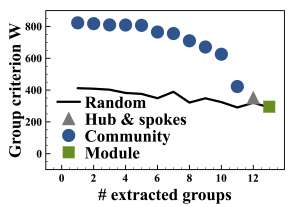
Sequential group extraction: Šubelj et al. (2013a) & Zhao et al. (2011)

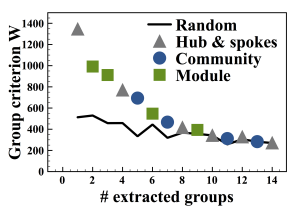
(1) Find S & T that optimize W (tabu search)

(2) Extract *only links between* S & T (& isolated nodes)

(–) Repeat until W larger than at random (by simulation)







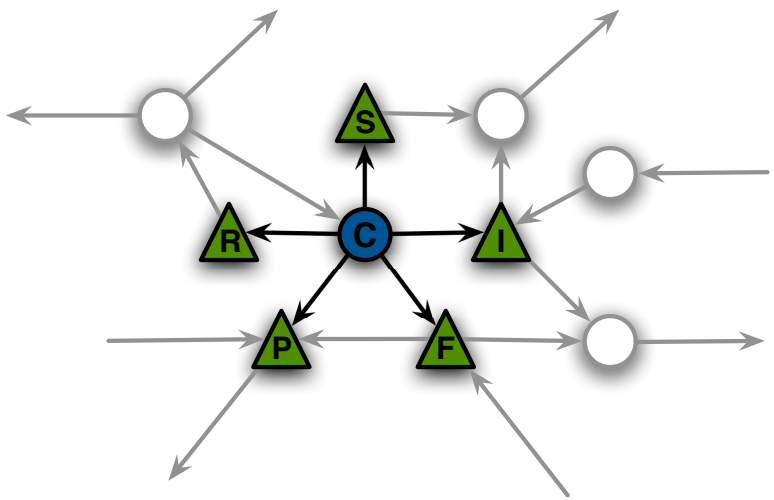
Software networks

Class dependency software networks: Šubelj and Bajec (2011)

nodes → *classes* of an object-oriented software project

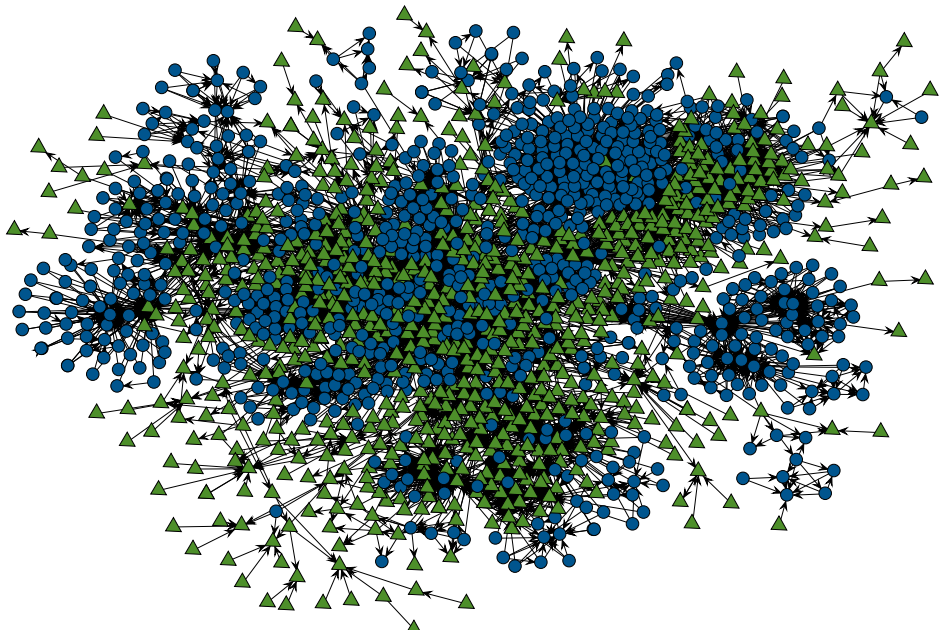
links → *dependencies* between classes (e.g., inheritance)

```
class C extends S implements I {

        F field;

        public C() { ... }

        void foo(P parameter) { ... }

        private R bar() { ... }

}
```



Structure of software networks

Software networks are similar to other real-world networks. Valverde et al. (2002)

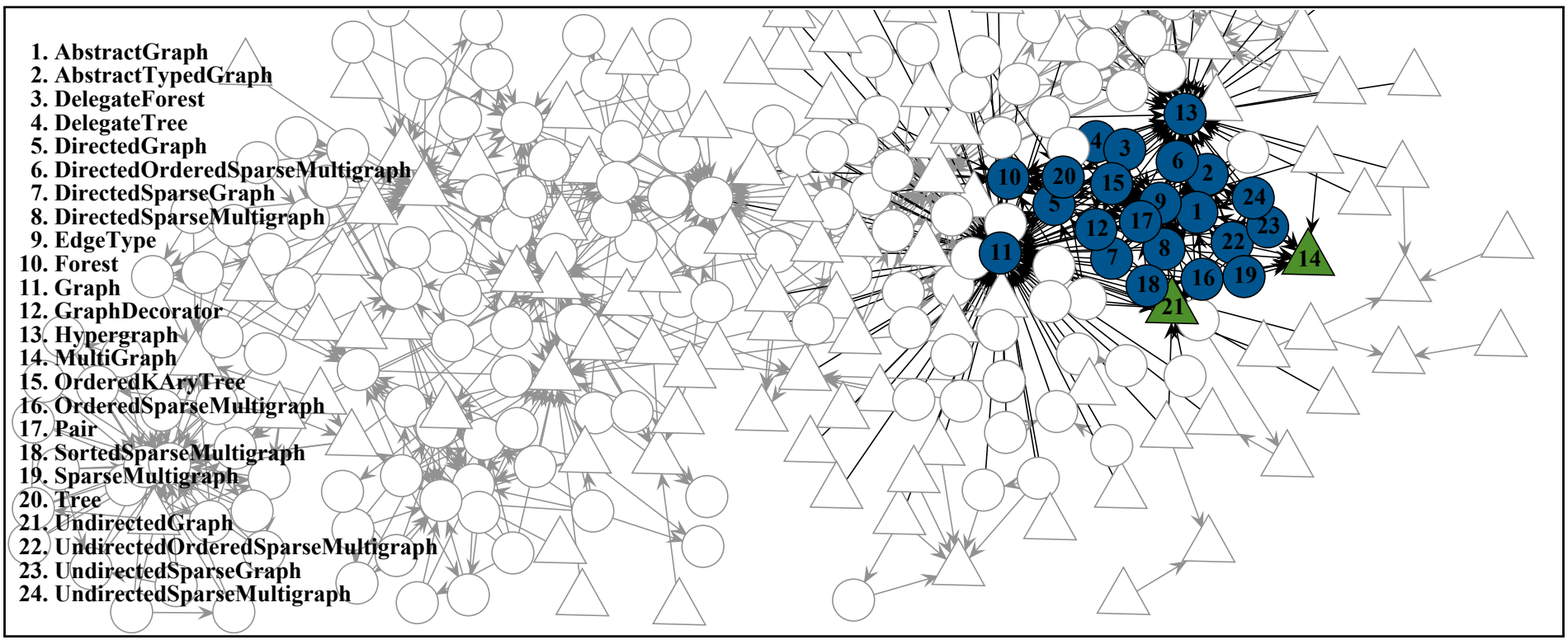


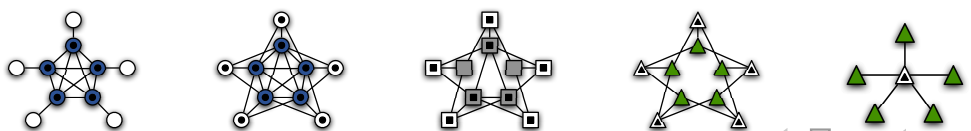
software networks = Šubelj et al. (2013b)

= dense social network structure + sparse Internet topology

Communities in software networks

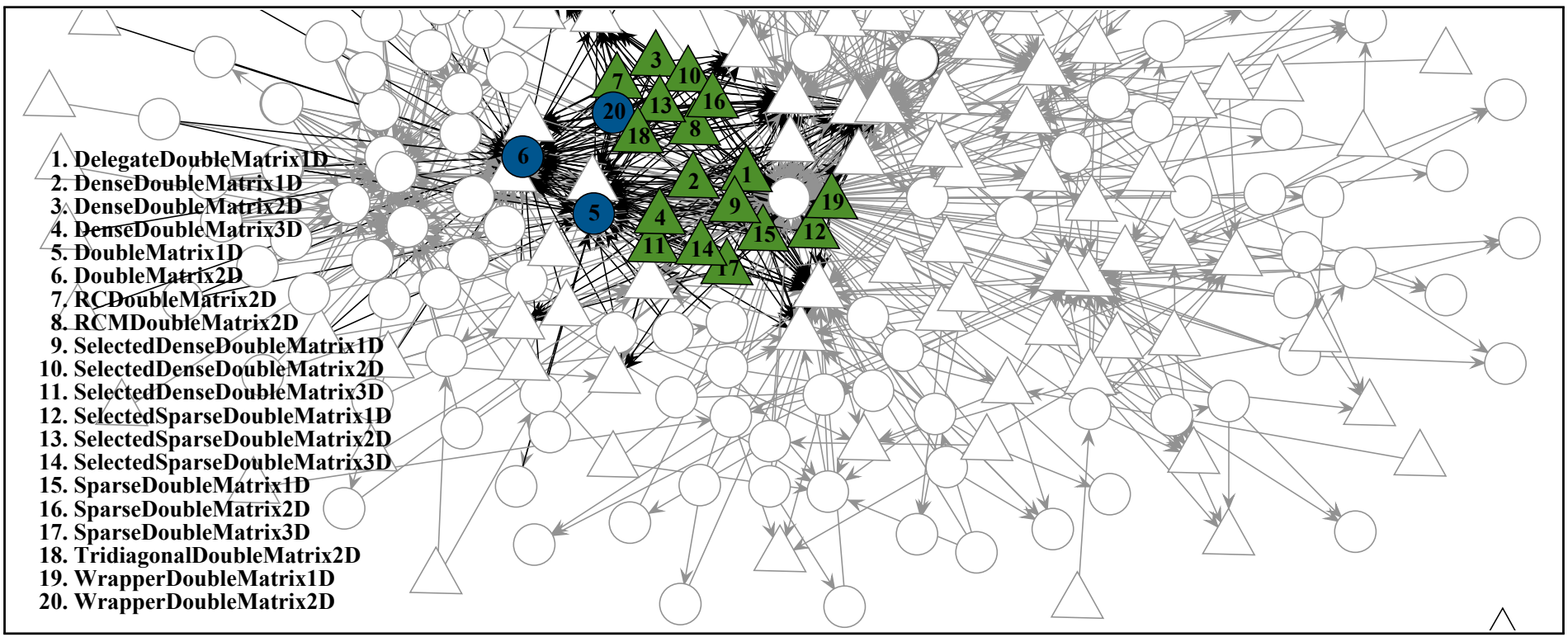
Communities are *core classes* of the software project. Šubelj and Bajec (2011)

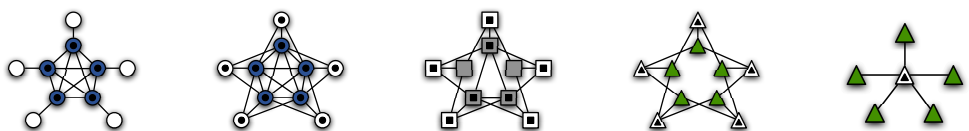




Modules in software networks

Modules are classes with the *same functionality*. Šubelj and Bajec (2012b)





Software engineering

Accuracy of *class package* prediction: Šubelj et al. (2013b)

Software	# Classes	# Categories	Neighbors Γ	**Groups** S	Network N	*Baseline*	*Random*
JBullet	107	11	72.0%	**75.7%**	64.5%	*28.0%*	*8.6%*
colt	154	16	58.4%	**73.4%**	55.2%	*22.7%*	*5.9%*
JUNG	237	31	72.2%	**74.2%**	65.0%	*11.4%*	*3.3%*
Lucene	1335	178	47.1%	**49.2%**	43.7%	*6.4%*	*0.6%*

Accuracy of *high-level class package* prediction:

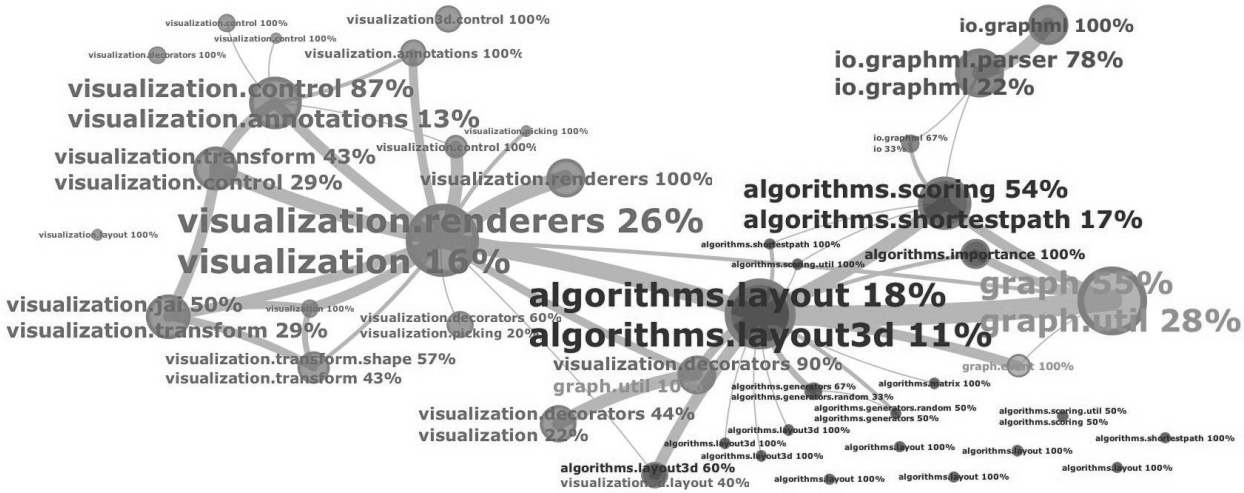
Software	# Classes	# Categories	Neighbors Γ	**Groups** S	Network N	*Baseline*	*Random*
JBullet	107	5	**84.6%**	**85.0%**	78.5%	*64.5%*	*20.4%*
colt	154	10	**86.4%**	83.8%	69.5%	*39.0%*	*9.7%*
JUNG	237	5	89.1%	**90.5%**	**91.1%**	*44.3%*	*20.3%*
Lucene	1335	15	85.5%	**90.8%**	85.0%	*28.2%*	*6.6%*

Accuracy of *class type*, *version*, *author* prediction:

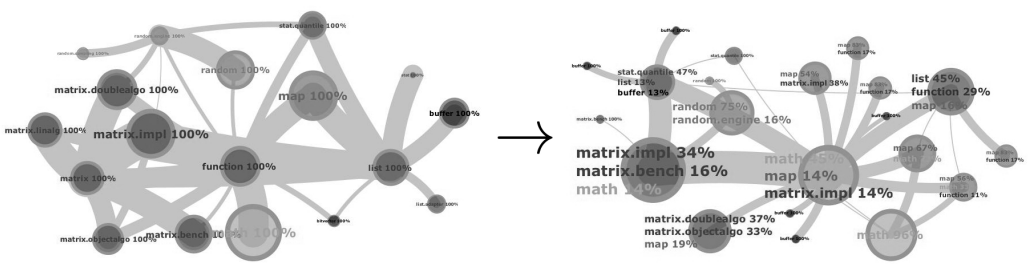
Setting	# Categories	Neighbors Γ	**Groups** S	Network N	*Baseline*	*Random*
Class type	2	65.0%	**85.2%**	**84.8%**	**84.4%**	*49.9%*
Class version	9	67.7%	**72.8%**	66.2%	*44.3%*	*11.2%*
Class author	11	**71.6%**	**71.0%**	**70.9%**	*44.3%*	*9.2%*

Software engineering (II)

High-level abstraction of a software system: Šubelj and Bajec (2012a)



Reorganization of software packages (modular or functional):



$Q = -0.0262$ $Q = 0.5020$

M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *P. Natl. Acad. Sci. USA*, 99(12):7821–7826, 2002.

M. E. J. Newman and E. A. Leicht. Mixture models and exploratory analysis in networks. *P. Natl. Acad. Sci. USA*, 104(23):9564, 2007.

L. Šubelj and M. Bajec. Community structure of complex software systems: Analysis and applications. *Physica A*, 390(16):2968–2975, 2011.

L. Šubelj and M. Bajec. Software systems through complex networks science: Review, analysis and applications. In *Proceedings of the KDD Workshop on Software Mining*, pages 9–16, Beijing, China, 2012a.

L. Šubelj and M. Bajec. Ubiquitousness of link-density and link-pattern communities in real-world networks. *Eur. Phys. J. B*, 85(1):32, 2012b.

L. Šubelj, N. Blagus, and M. Bajec. Group extraction for real-world networks: The case of communities, modules, and hubs and spokes. In *Proceedings of the International Conference on Network Science*, pages 152–153, Copenhagen, Denmark, 2013a.

L. Šubelj, S. Žitnik, N. Blagus, and M. Bajec. Node mixing and group structure of complex software networks. *sub. to Adv. Complex Syst.*, page 23, 2013b.

S. Valverde, R. F. Cancho, and R. V. Solé. Scale-free networks from optimal design. *Europhys. Lett.*, 60(4):512–517, 2002.

Y. Zhao, E. Levina, and J. Zhu. Community extraction for social networks. *P. Natl. Acad. Sci. USA*, 108(18):7321–7326, 2011.