Data Science
São Paulo School of Advanced Science
on Learning from Data
USP

ROYAL
HOLLOWAY
UNIVERSITY
OF LONDON

Machine Learning algorithms for making inferences on networks and answering questions in Biology and Medicine

Alberto Paccanaro
Department of Computer Science
Royal Holloway, University of London

São Paulo School of Advanced Science on Learning from Data, 2019

Why ML and biology

We need to analyse the cell at systems level

Large scale experiments interrogate the cell at the system level

Detect patterns in large amounts of very noisy data
Integrate diverse sets of data from different sources

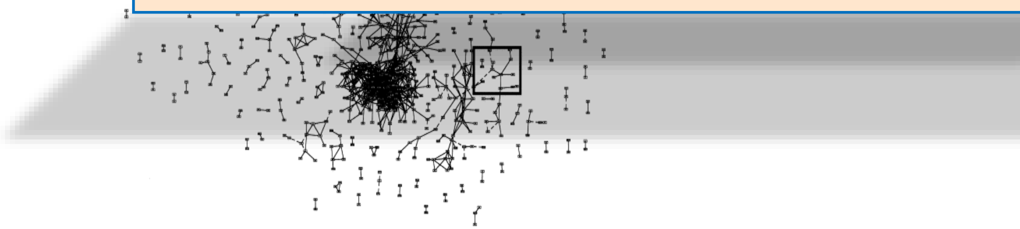
A. Paccanaro, 2019

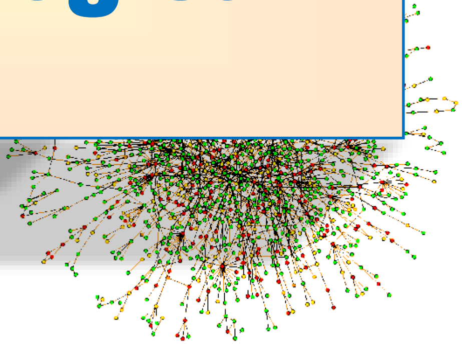
Biological networks

Cell as webs of interactions between biomolecules
Experimental data have a natural representation as networks

Many problems in biology and medicine can be formulated as *problems of inference* on biological networks

Gene





Co-expression networks

Protein-protein interaction networks

[Horak, Genes & Dev.; DeRisi, Science; Qian, J. Mol. Bio; Jeong, Nature,;Tong, Science; Goh, PNAS]

A. Paccanaro, 2019

In my lab, we develop Machine Learning methods for answering questions in Biology and Medicine

focus on biological networks

- At the heart of our research is the biological question, not the methodology – **different areas of ML**
- **Diverse problems**
- Collaborate with **experimentalists**
- We implement **software tools** that allow biologists and clinicians to easily use the methods that we develop

A. Paccanaro, 2019

Acknowledgements

Horacio Caniza

Alfonso E. Romero

Juan Caceres

Haixuan Yang

Mateo Torres

Tamas Nepusz

Diego Galeano

Cheng Ye

Ruben Jimenez

Jessica Gliozzo

Collaborating labs:
Mark Gerstein – Yale University
Andrew Emili – Un of Toronto
Edward Marcotte – Un Texas, Austin
Haiyuan Yu – Cornell University
Paul Matthews – Imperial College
Rajkumar Sasidharan – UCLA
Giorgio Valentini – Un. of Milan
Simon Barak – Ben Gurion Un.
Laszlo Bogre – Royal Holloway

http://www.paccanarolab.org

The Menu

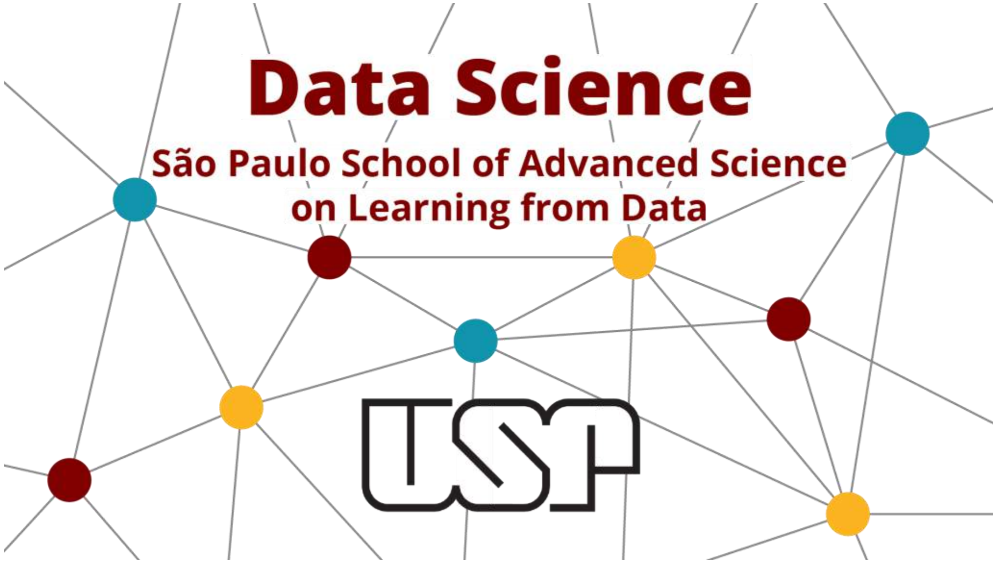
1. **Network Science (brief intro)**

2. **Biological networks**

3. **Network Medicine, Systems Pharmacology**
   - Measure of distance between hereditary disease modules on the interactome (2015)
   - Disease gene prediction for uncharted diseases (2019)

4. **Recommender Systems**
   - Method for predicting the frequency of drug side effects (under review)

5. **Clustering, Spectral Clustering, Information diffusion**
   - ClusterONE (2012)
   - Spectral clustering of protein sequences (2009)
   - An information diffusion approach to de-noise large-scale networks (2012)



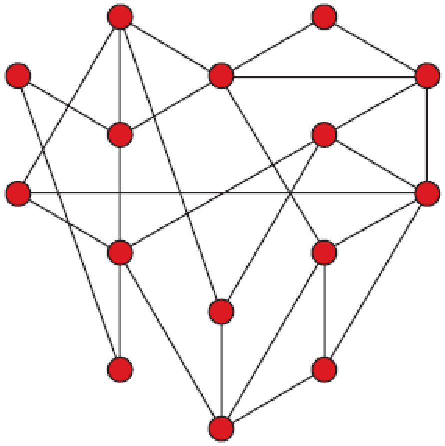
A brief intro to Network Science

Alberto Paccanaro
Department of Computer Science
Royal Holloway, University of London
www.paccanarolab.org

The Erdös-Rényi model (1959)

To build a random graph with n nodes:

```
For each pair of nodes
    connect the pair with probability $p$
endfor;
```



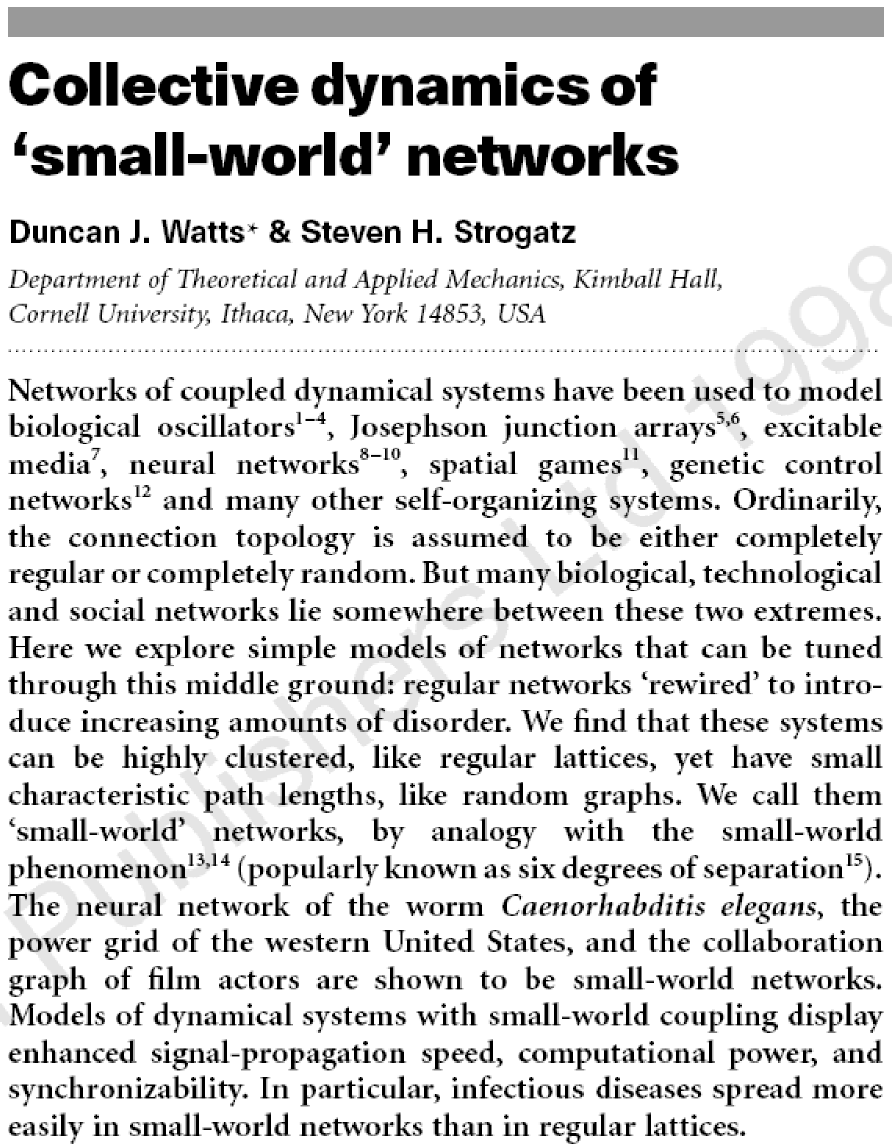
This creates a graph with approximately:

$$p\ \frac{n(n-1)}{2}$$

randomly placed links.

A. Paccanaro, 2019

Nature, Vol. 393, 440, 1998



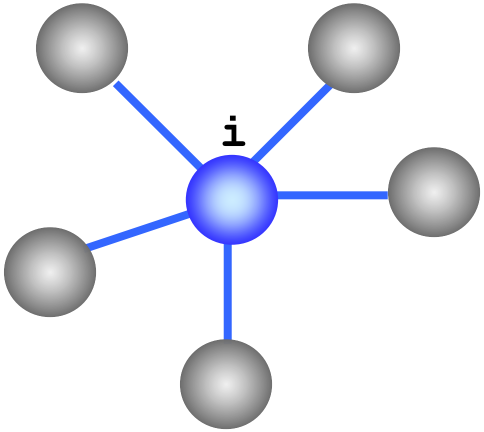
Collective dynamics of ‘small-world’ networks

Duncan J. Watts[*] **& Steven H. Strogatz**

Department of Theoretical and Applied Mechanics, Kimball Hall,
Cornell University, Ithaca, New York 14853, USA

Networks of coupled dynamical systems have been used to model biological oscillators[1–4], Josephson junction arrays[5,6], excitable media[7], neural networks[8–10], spatial games[11], genetic control networks[12] and many other self-organizing systems. Ordinarily, the connection topology is assumed to be either completely regular or completely random. But many biological, technological and social networks lie somewhere between these two extremes. Here we explore simple models of networks that can be tuned through this middle ground: regular networks ‘rewired’ to introduce increasing amounts of disorder. We find that these systems can be highly clustered, like regular lattices, yet have small characteristic path lengths, like random graphs. We call them ‘small-world’ networks, by analogy with the small-world phenomenon[13,14] (popularly known as six degrees of separation[15]). The neural network of the worm *Caenorhabditis elegans*, the power grid of the western United States, and the collaboration graph of film actors are shown to be small-world networks. Models of dynamical systems with small-world coupling display enhanced signal-propagation speed, computational power, and synchronizability. In particular, infectious diseases spread more easily in small-world networks than in regular lattices.

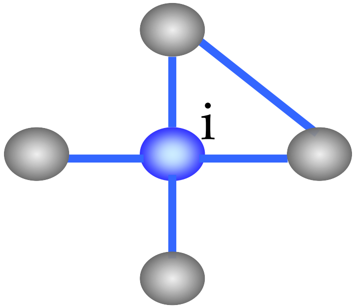
Degree of a node: the number of edges incident on the node



Degree of node i = 5

(Local) Clustering coefficient → LOCAL property

The clustering coefficient of node i is the ratio of the number E_i of edges that exist among its neighbours, over the number of edges that could exist



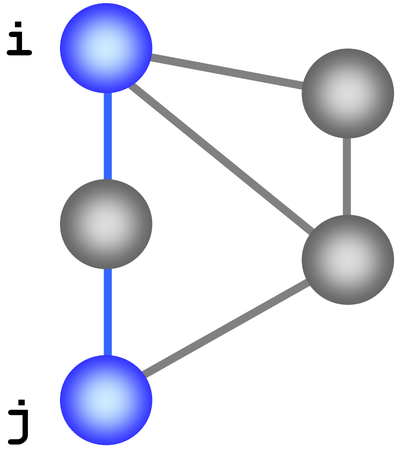
if node i has k neighbours, then at most k(k-1)/2 edges can exist between them

Clustering coefficient of node i = 1/6

The clustering coefficient for the entire network C is the average of all the C_i

Characteristic path length → GLOBAL property

$L_{(i,j)}$ is the number of edges in the shortest path between vertices *i* and *j*

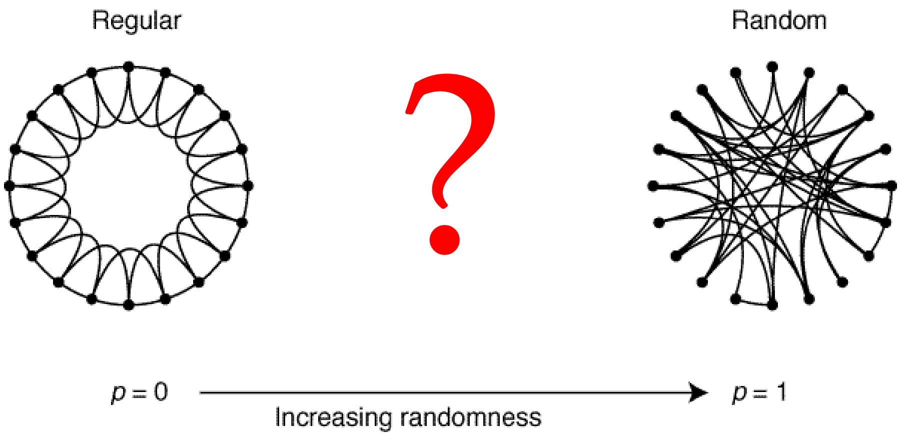


$$L_{(i,j)} = 2$$

The characteristic path length L of a graph is the average of the $L_{(i,j)}$ for every possible pair *(i,j)*

A. Paccanaro, 2019

Watts & Strogatz: the idea/the question



From T.J.Watts, S.H Strogatz, Nature, Vol. 393, 440, 1998

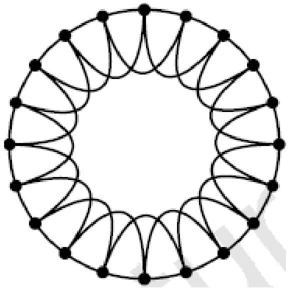
REWIRING PROCEDURE

- Start with a regular network with n vertices
- Rewire each edge with probability p

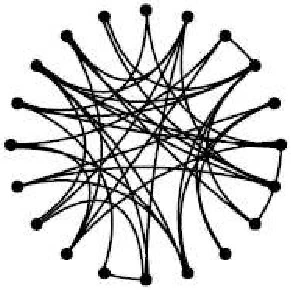
$p = 0$ ➔ regularity
$p = 1$ ➔ disorder (random)
Question: what happens for $0 < p < 1$?

Quantify the structural properties of the graph by its characteristic path length *L(p)* and clustering coefficient *C(p)*

A. Paccanaro, 2019



n vertices
k edges per vertex



From T.J.Watts, S.H Strogatz, Nature, Vol. 393, 440, 1998

For *p* → *0* (Regular Networks):

- high clustering coefficient
- high characteristic path length

- highly clustered
- large world [L grows lin. with n]

For *p* → *1* (Random Networks):

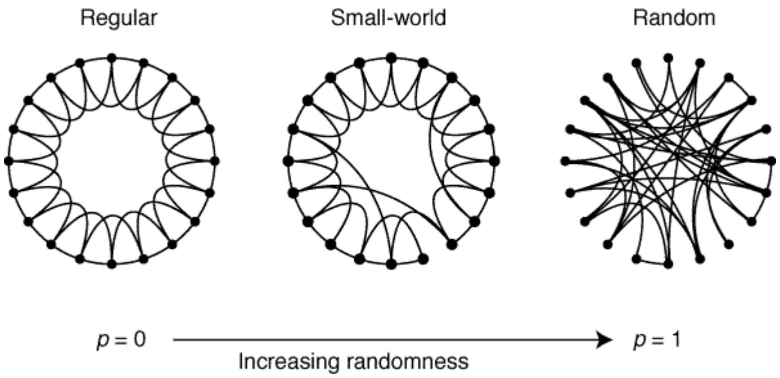
- low clustering coefficient
- low characteristic path length

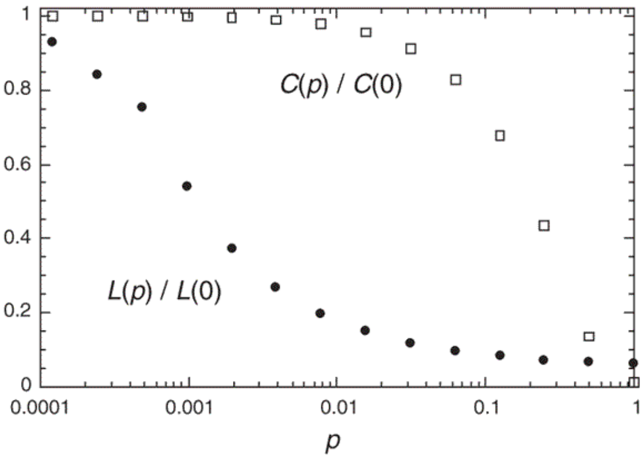
- poorly clustered
- small world [L grows log. with n]

A. Paccanaro, 2019

This might lead to think that large C is always associated with large L, and small C with small L…

1) There is a broad interval of *p* for which *L* is small, but *C* remains large





From T.J.Watts, S.H Strogatz, Nature, Vol. 393, 440, 1998

A. Paccanaro, 2019

2) Hypothesis: small-world property might be common in sparse networks with many vertices as even a tiny fraction of short cuts could be sufficient

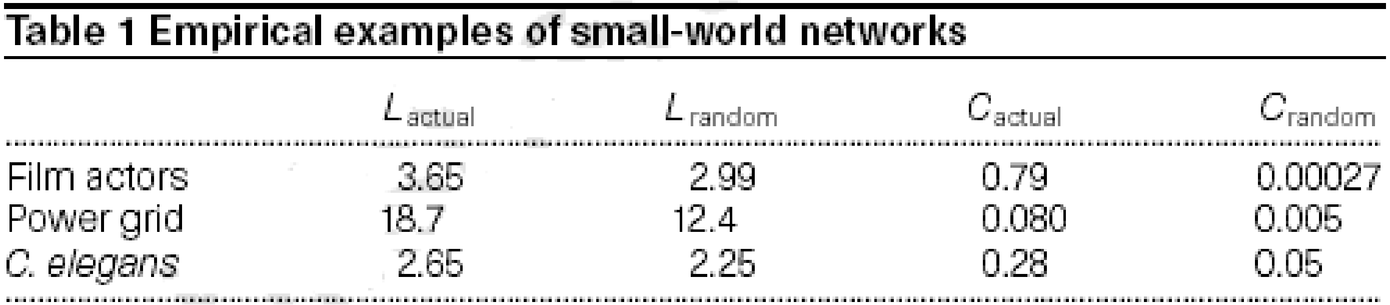


Table 1 Empirical examples of small-world networks

	L_{actual}	L_{random}	C_{actual}	C_{random}
Film actors	3.65	2.99	0.79	0.00027
Power grid	18.7	12.4	0.080	0.005
C. elegans	2.65	2.25	0.28	0.05

From T.J.Watts, S.H Strogatz, Nature, Vol. 393, 440, 1998

Comparison with random graphs with the same number of vertices n and average degree k

Actors:	n=225226	k=61
Power grid:	n=4941	k=2.67
C.Elegans:	n=282	k=14

A. Paccanaro, 2019

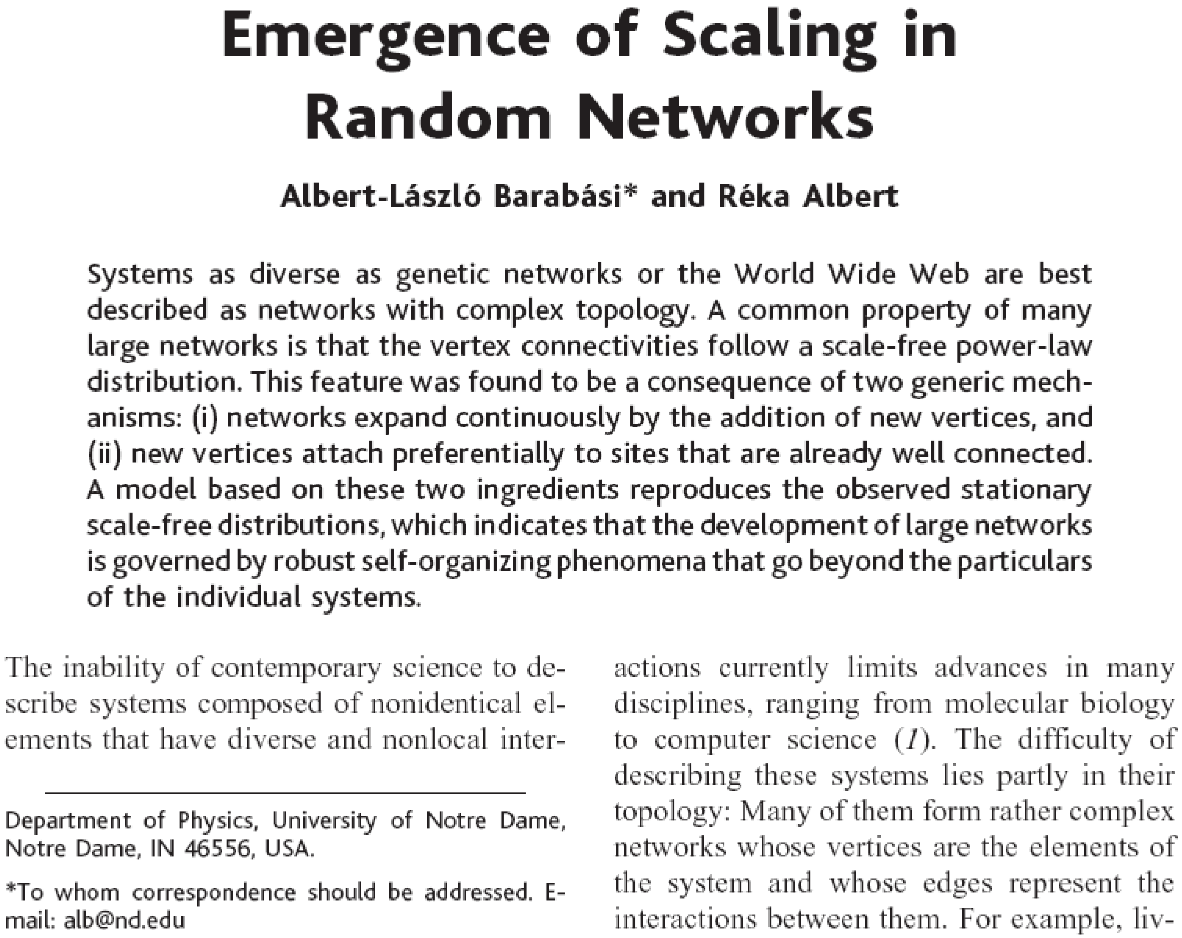
Conclusions

- The *small-world phenomenon* is not merely a curiosity of social networks nor an artefact of an idealized model --- ***it is probably generic for many large, sparse networks found in nature***

- The distinctive combination of high clustering with short characteristic path length in small-world networks **cannot be captured by traditional approximations** such as those based on regular lattices or random graphs.

A. Paccanaro, 2019

Science, Vol. 286, 1999



Emergence of Scaling in Random Networks

Albert-László Barabási* and Réka Albert

Systems as diverse as genetic networks or the World Wide Web are best described as networks with complex topology. A common property of many large networks is that the vertex connectivities follow a scale-free power-law distribution. This feature was found to be a consequence of two generic mechanisms: (i) networks expand continuously by the addition of new vertices, and (ii) new vertices attach preferentially to sites that are already well connected. A model based on these two ingredients reproduces the observed stationary scale-free distributions, which indicates that the development of large networks is governed by robust self-organizing phenomena that go beyond the particulars of the individual systems.

The inability of contemporary science to describe systems composed of nonidentical elements that have diverse and nonlocal interactions currently limits advances in many disciplines, ranging from molecular biology to computer science (*1*). The difficulty of describing these systems lies partly in their topology: Many of them form rather complex networks whose vertices are the elements of the system and whose edges represent the interactions between them. For example, liv-

Department of Physics, University of Notre Dame, Notre Dame, IN 46556, USA.

*To whom correspondence should be addressed. E-mail: alb@nd.edu

1. Actors
2. Power grid
3. WWW

A. Paccanaro, 2019

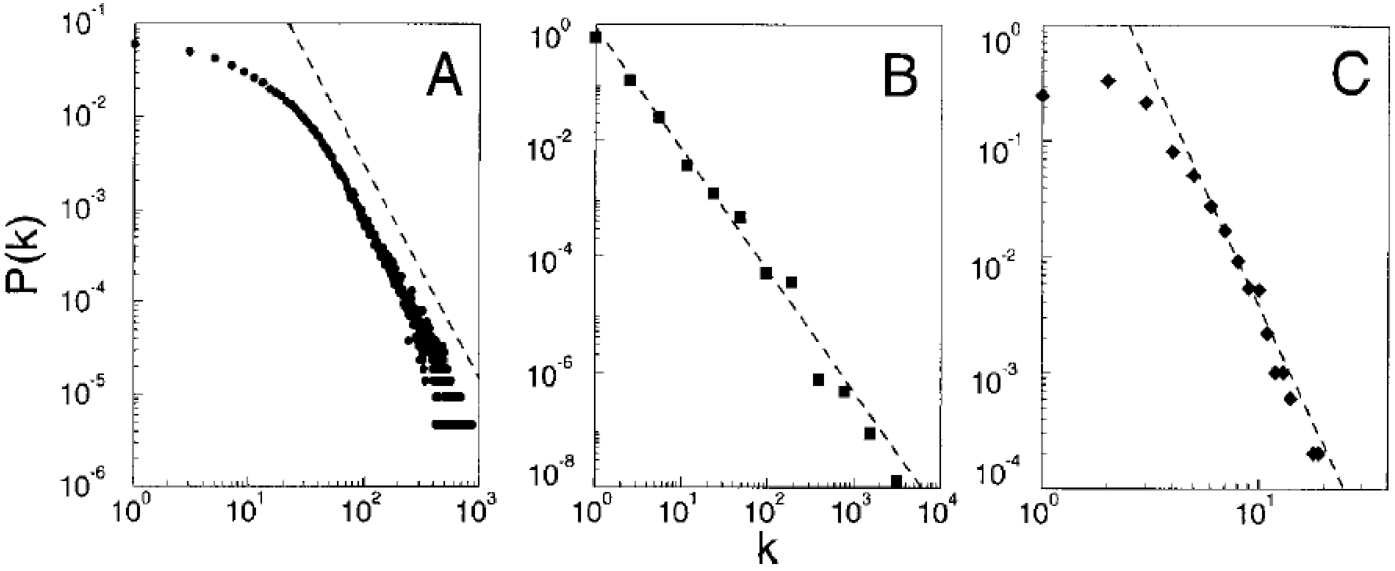


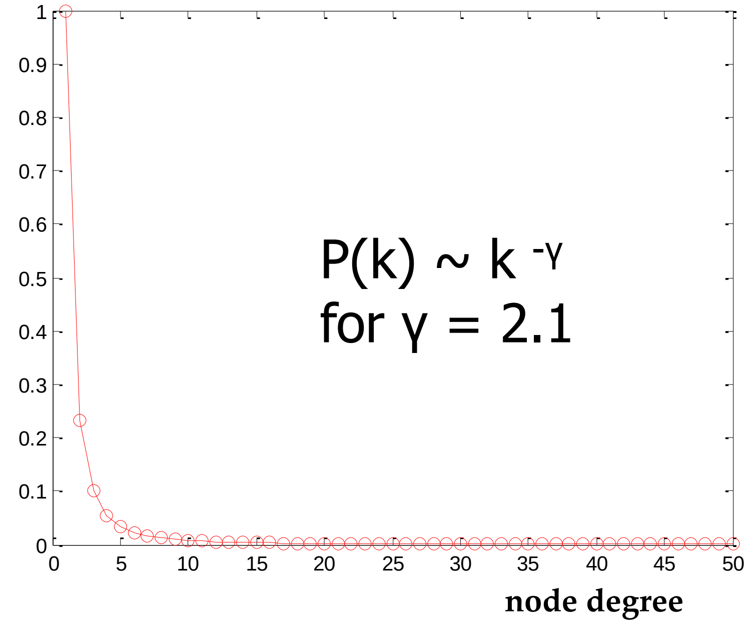
Fig. 1. The distribution function of connectivities for various large networks. **(A)** Actor collaboration graph with $N = 212{,}250$ vertices and average connectivity $\langle k \rangle = 28.78$. **(B)** WWW, $N = 325{,}729$, $\langle k \rangle = 5.46$ (6). **(C)** Power grid data, $N = 4941$, $\langle k \rangle = 2.67$. The dashed lines have slopes (A) $\gamma_{actor} = 2.3$, (B) $\gamma_{www} = 2.1$ and (C) $\gamma_{power} = 4$.

From A.L. Barabasi, R. Albert, Science, Vol. 286, 1999

A. Paccanaro, 2019

Independent of the system and the identity of its constituents, the probability $P(k)$ that a vertex in the network interacts with k other vertices decays as a power law:

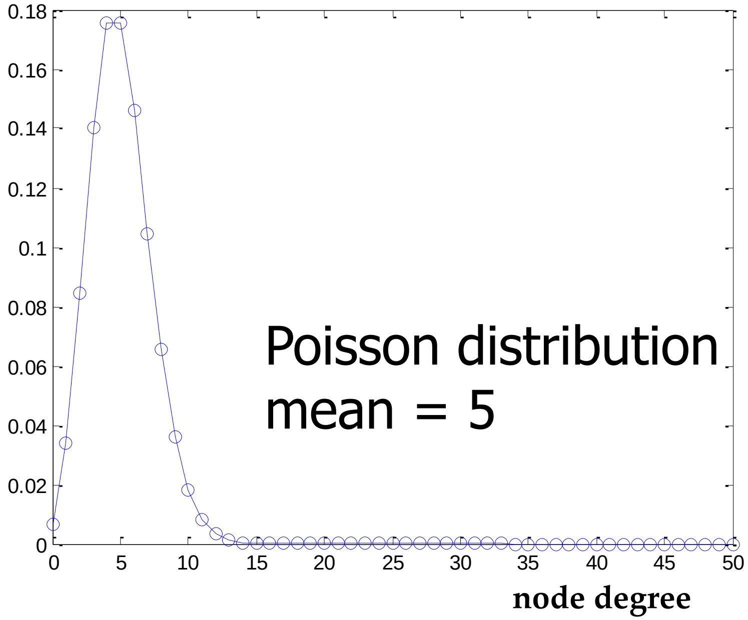
$$P(k) \sim k^{-\gamma}$$



$P(k) \sim k^{-\gamma}$
for $\gamma = 2.1$

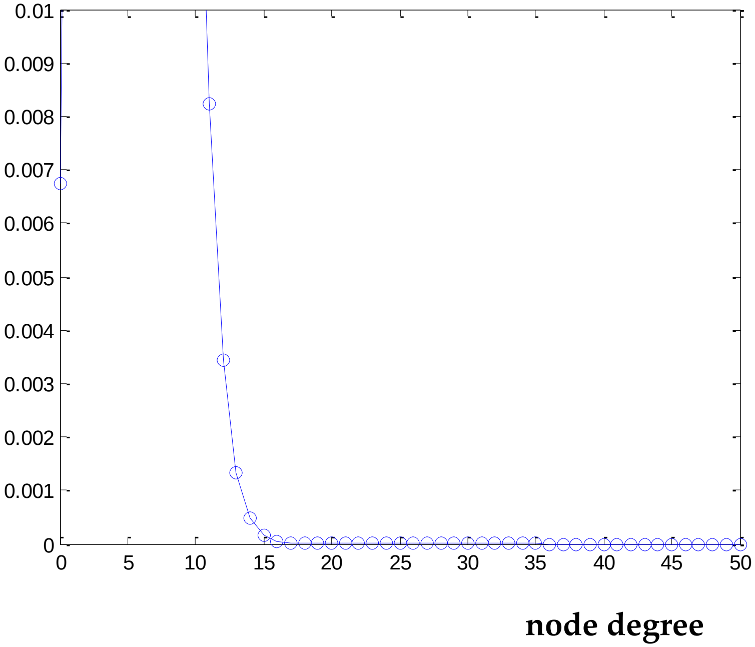
In the Erdos-Renyi models the node degrees follow a Poisson distribution

- most nodes have approximately the same number of links (~ $<k>$)
- the tail (high k region) of the degree distribution $P(k)$ decreases exponentially, which indicates that nodes that significantly deviate from the average are extremely rare

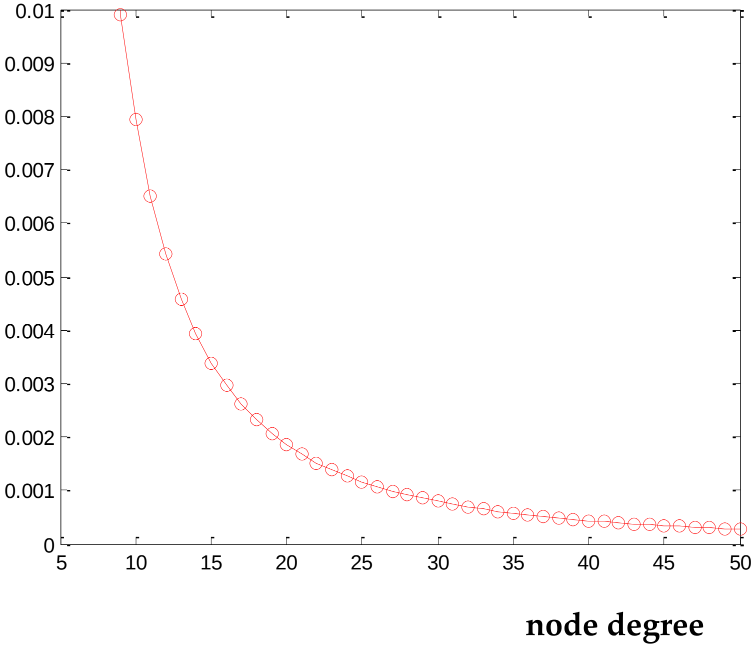


Poisson distribution
mean = 5

A. Paccanaro, 2019



Poisson distribution
mean = 5



$P(k) \sim k^{-\gamma}$
for $\gamma = 2.1$

A. Paccanaro, 2019

- **ER model**: the probability of finding a highly connected vertex (that is, a large *k*) decreases exponentially with *k*; thus, **vertices with large connectivity are practically absent**.

- **Scale Free model**: the power-law tail characterizing *P*(*k*) for the networks studied indicates that **highly connected (large *k*) vertices have a large chance of occurring, dominating the connectivity.**

A. Paccanaro, 2019

Implications for Network reliability

- This type of network is extremely robust to random destruction/malfunction of one of its components

- It is extremely **vulnerable to well-aimed attacks**

A. Paccanaro, 2019

Two mechanisms behind the generation of random networks

1. real world networks are formed by the continuous addition of new vertices to the system, thus the number of vertices n increases throughout the lifetime of the network

2. most real networks exhibit preferential connectivity.

   The probability with which a new vertex connects to the existing vertices is not uniform; there is a higher probability that it will be linked to a vertex that already has a large number of connections

A. Paccanaro, 2019

Conclusion

1. A common property of many large networks is that the vertex connectivity follows a **scale-free power-law** distribution.

2. This feature was found to be a consequence of **two generic mechanisms**:
   - (i) networks expand continuously by the addition of new vertices, and
   - (ii) new vertices attach preferentially to sites that are already well connected.

→ A model based on these two ingredients reproduces the observed stationary scale-free distributions.

A. Paccanaro, 2019

Zipf's Law

In a natural language, the frequency of any word is roughly inversely proportional to its rank in the frequency table

$$f(n) \sim n^{-a}$$

where f_n is the frequency of occurrence of the nth ranked item and *a* is close to 1

A. Paccanaro, 2019

Reading material

- Papers from which I took some figures:
  - ❑ T.J.Watts, S.H Strogatz, Nature, Vol. 393, 440, 1998
  - ❑ A.L. Barabasi, R. Albert, Science, Vol. 286, 1999

- Other relevant readings:
  - ❑ Mark Newman, Networks: An Introduction, 2nd ed, 2018
  - ❑ Albert Barabasi, Network Science, 2015

  (available to read online http://networksciencebook.com/)

Data Science

São Paulo School of Advanced Science on Learning from Data

USP

Biological Networks

Alberto Paccanaro

Department of Computer Science
Royal Holloway, University of London

www.paccanarolab.org

A. Paccanaro, 2019

PROTEINS

Movie:

https://www.youtube.com/watch?v=X_tYrnv_o6A

A. Paccanaro, 2019

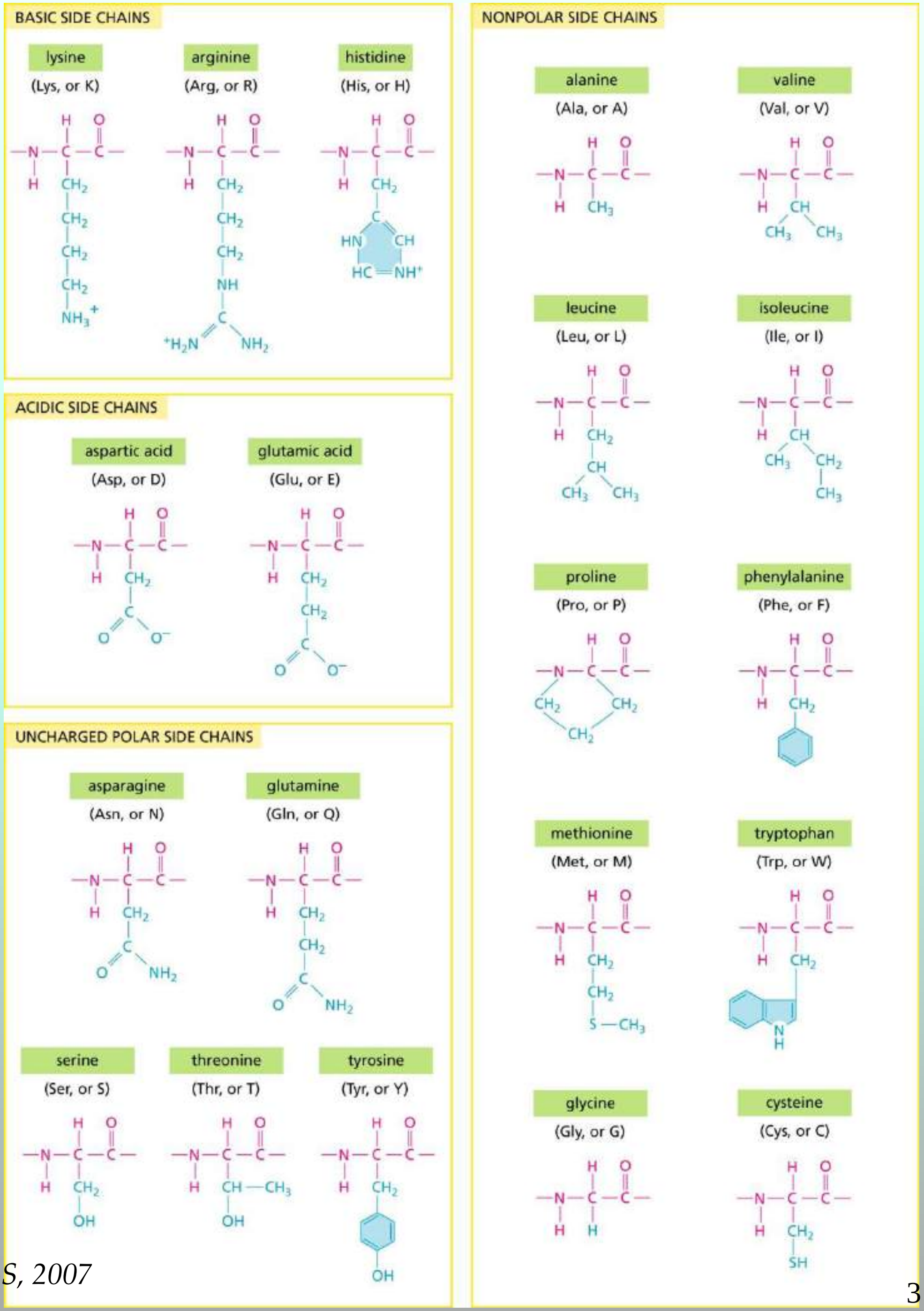
Amino acids

Proteins made out of long chains of 20 different types of aminoacids…

We need to store the sequence of aminoacids that make each protein.

We need a code for each aminoacid

We need an alphabet…

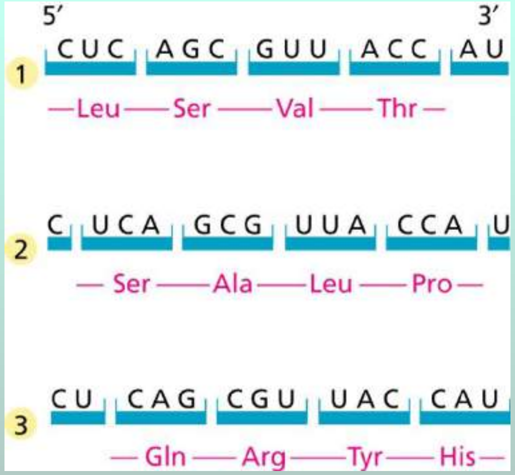


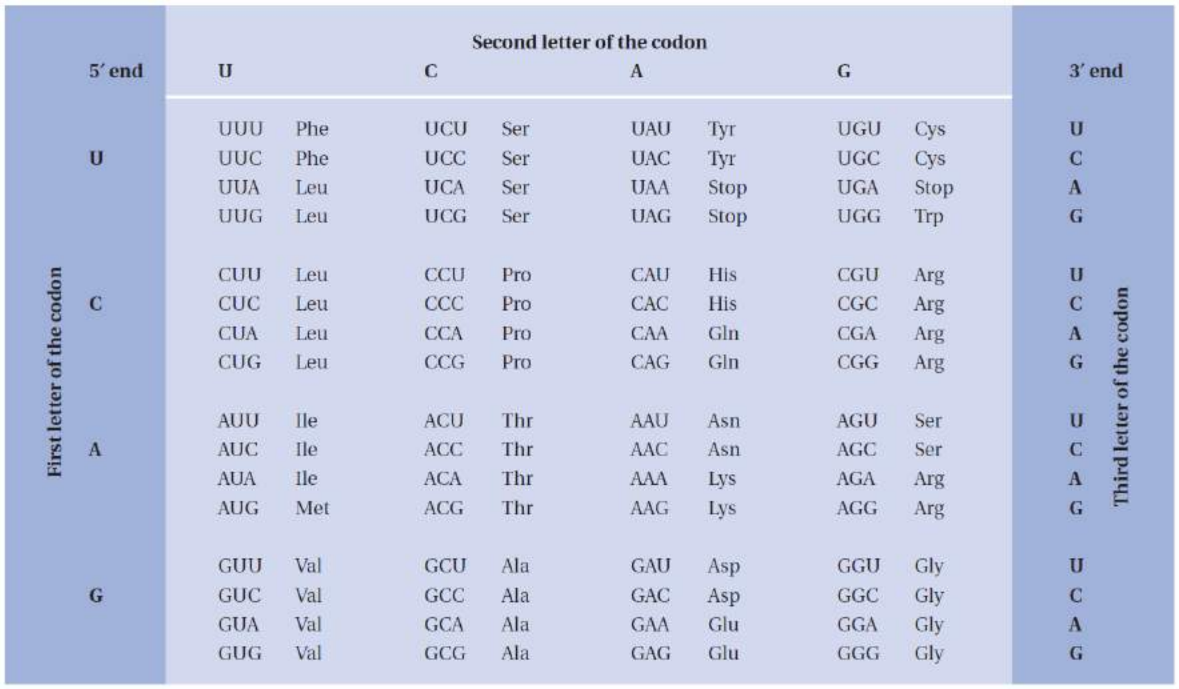
From M. Zvelebil, J. Baum, Understanding Bioinformatics, GS, 2007

A. Paccanaro, 2019

The code

- We need to code for 20 aminoacids
- We have a 4 letter alphabet…

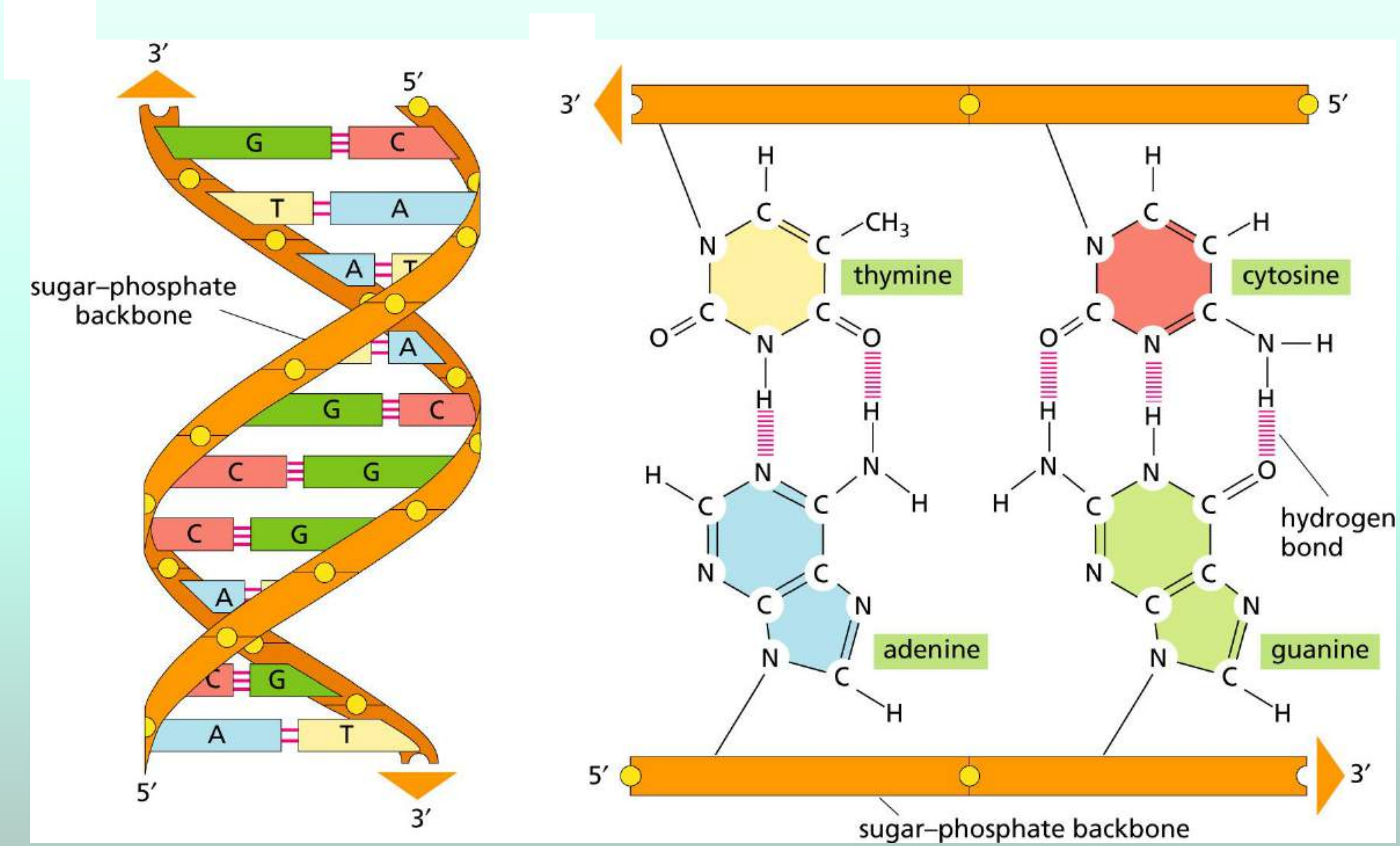




First letter of the codon	Second letter of the codon								Third letter of the codon
5′ end	U		C		A		G		3′ end
U	UUU	Phe	UCU	Ser	UAU	Tyr	UGU	Cys	U
	UUC	Phe	UCC	Ser	UAC	Tyr	UGC	Cys	C
	UUA	Leu	UCA	Ser	UAA	Stop	UGA	Stop	A
	UUG	Leu	UCG	Ser	UAG	Stop	UGG	Trp	G
C	CUU	Leu	CCU	Pro	CAU	His	CGU	Arg	U
	CUC	Leu	CCC	Pro	CAC	His	CGC	Arg	C
	CUA	Leu	CCA	Pro	CAA	Gln	CGA	Arg	A
	CUG	Leu	CCG	Pro	CAG	Gln	CGG	Arg	G
A	AUU	Ile	ACU	Thr	AAU	Asn	AGU	Ser	U
	AUC	Ile	ACC	Thr	AAC	Asn	AGC	Ser	C
	AUA	Ile	ACA	Thr	AAA	Lys	AGA	Arg	A
	AUG	Met	ACG	Thr	AAG	Lys	AGG	Arg	G
G	GUU	Val	GCU	Ala	GAU	Asp	GGU	Gly	U
	GUC	Val	GCC	Ala	GAC	Asp	GGC	Gly	C
	GUA	Val	GCA	Ala	GAA	Glu	GGA	Gly	A
	GUG	Val	GCG	Ala	GAG	Glu	GGG	Gly	G

A. Paccanaro, 2019

From M. Zvelebil, J. Baum, Understanding Bioinformatics, GS, 2007



From M. Zvelebil, J. Baum, Understanding Bioinformatics, GS, 2007

A. Paccanaro, 2019

The 2 strands can easily be separated

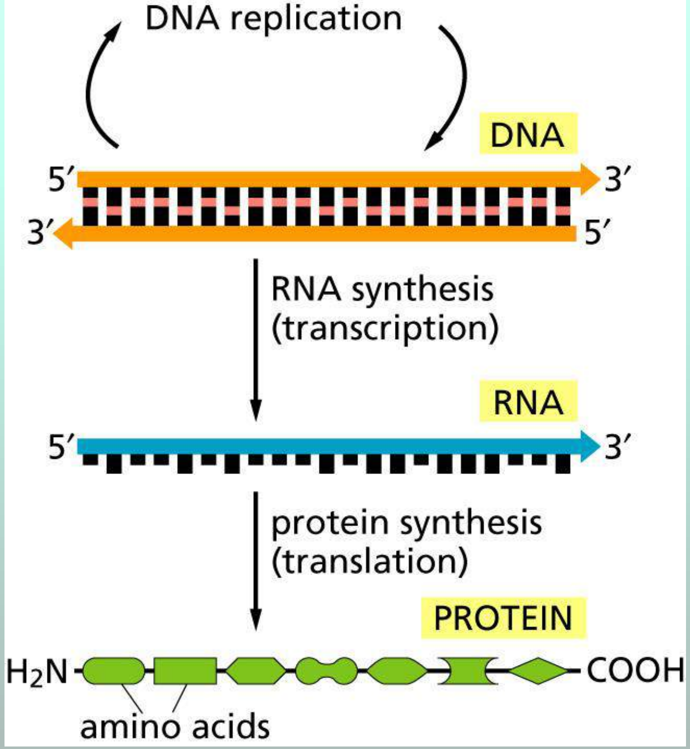
3 Fundamental Operations

1. Transcription
2. Translation
3. Replication

A. Paccanaro, 2019

The Central Dogma of Molecular Biology

- There is a single direction of flow of genetic information from DNA, through RNA into proteins.
- Genes



From M. Zvelebil, J. Baum, Understanding Bioinformatics, GS, 2007

A. Paccanaro, 2019

Note that not all genetic information encodes proteins…

A fundamental concept: the *Guilt by Association Principle*

If unknown gene/protein *i* *behaves* similarly to another gene *j*, maybe they are involved in the same/related biological process/pathway/ complex

Biomolecules rarely act in isolation, normally they work together with other cell components in order to achieve complex functions.

A. Paccanaro, 2019

Biological Networks

A. Paccanaro, 2019

Let us focus on Human

- ❑ ~ 25,000 protein coding genes
- ❑ Few thousands metabolites
- ❑ Functional RNA molecules…

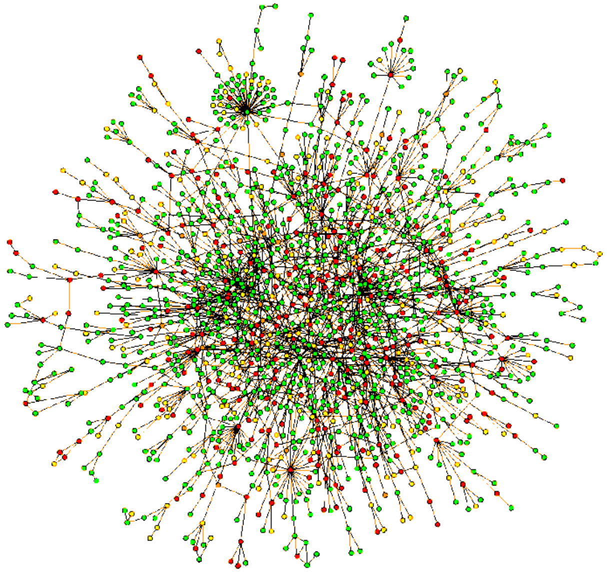
➔ total of **about 100,000 elements**

A. Paccanaro, 2019

1. Protein-Protein Interaction Networks

Nodes represent proteins and edges represent a physical interaction between two proteins.

Edges are non-directed



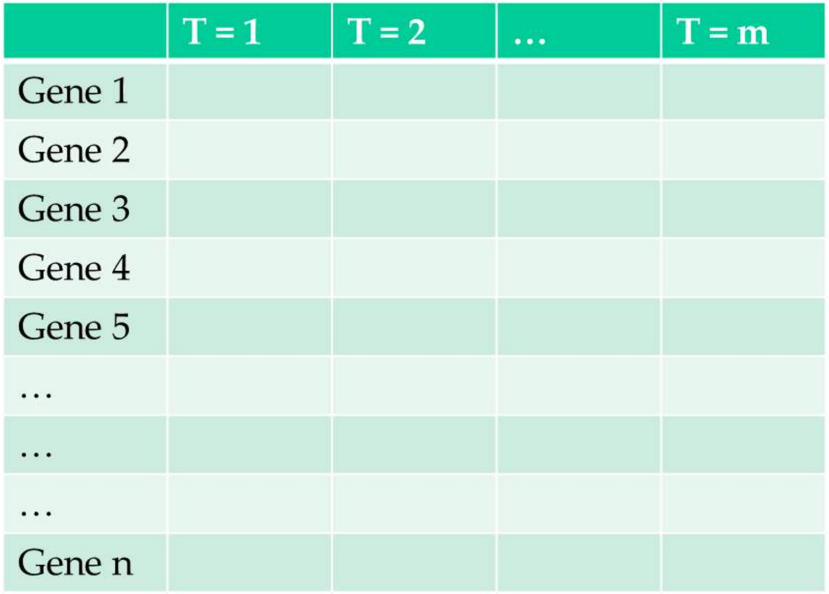
From Jeong et al, Nature 2001

A. Paccanaro, 2019

- Techniques: Y2H, AP/MS
- Databases: MIPS, BIND, MINT, DIP, Biogrid, HPRD, STRING
- ~ 40,000 known interactions in human
- 96% human protein have 3D inferred structures

2. Co-expression networks

Transcriptomics data:



	T = 1	T = 2	…	T = m
Gene 1				
Gene 2				
Gene 3				
Gene 4				
Gene 5				
…				
…				
…				
Gene n				

Fully connected network, where nodes are the genes and the links are weighted by the similarity in gene expression patterns (rows)

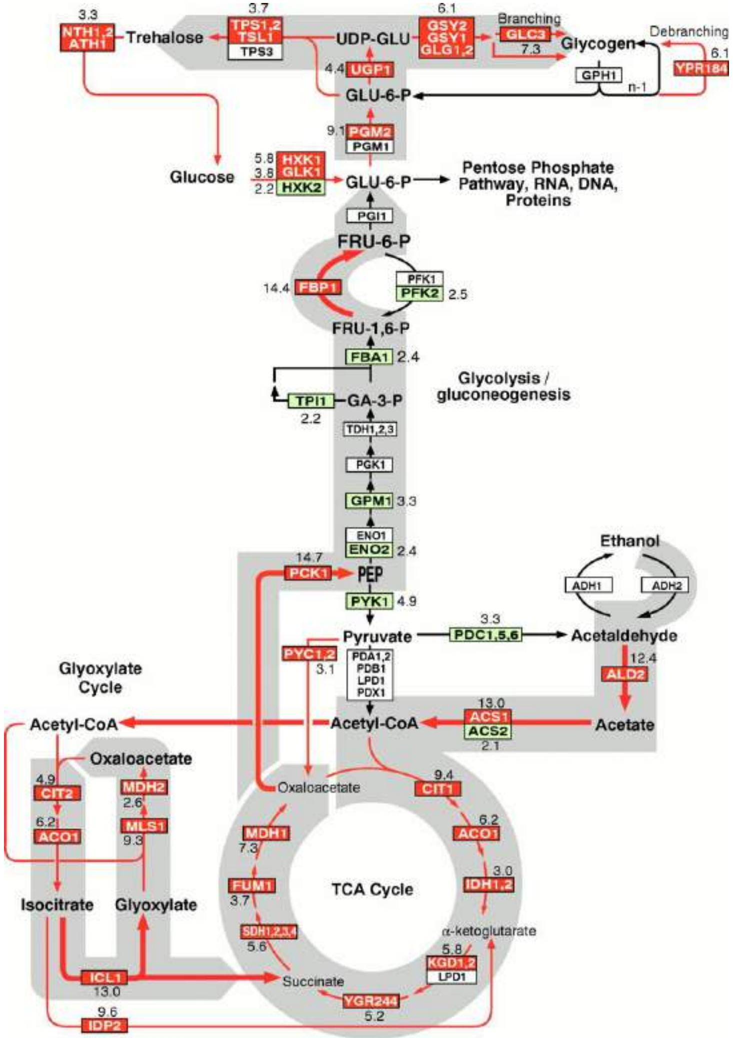
- databases: ArrayExpress, GEO

A. Paccanaro, 2019

3. Metabolic networks

Metabolic network maps attempt to comprehensively describe all possible biochemical reactions for a particular cell or organism

- databases: KEGG, BIGG
- 2766 metabolites, 3311 reactions



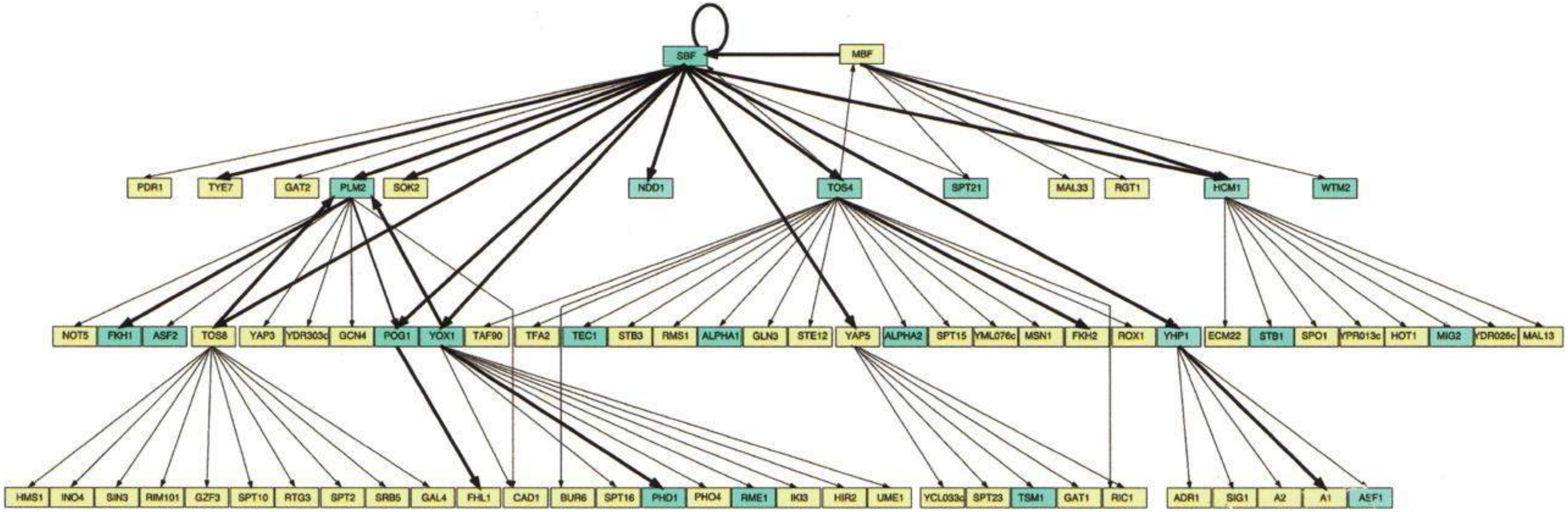
[from DeRisi, Iyer, and Brown, Science, 278:680-686]

A. Paccanaro, 2019

4. Gene Regulatory Networks

Nodes are either proteins or a putative DNA regulatory element and directed edges represent:

1. **Regulatory relationships** (the physical binding of transcription factors to regulatory elements)
   - Databases: UniPROBE, JASPAR, TRANSFAC, BCI
2. **Post-translational modifications** (e.g. kinases and its substrates)
   - Databases: PhosphoELM, PhosphoSite, PHOSIDA



From Horak, Genes & Dev. 2002

A. Paccanaro, 2019

5. RNA networks

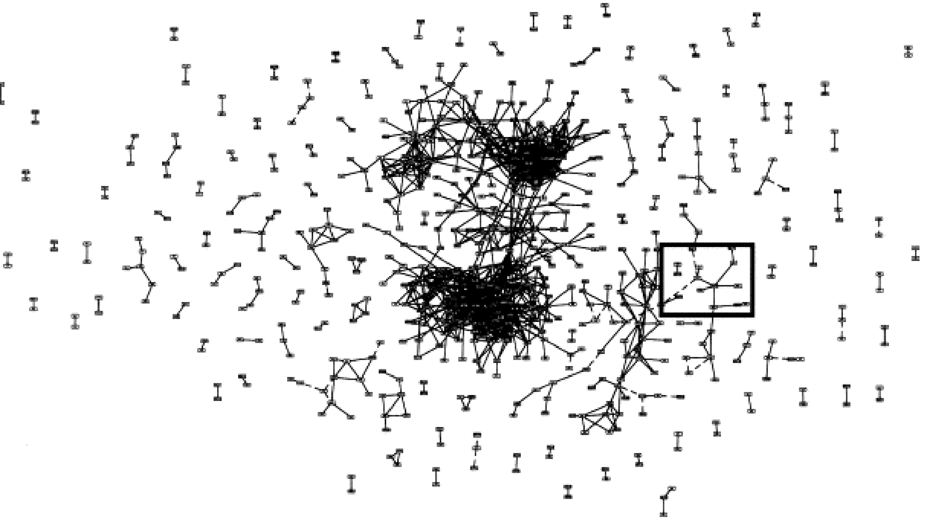
- They capture the interactions between RNAs and DNA in regulating gene expression

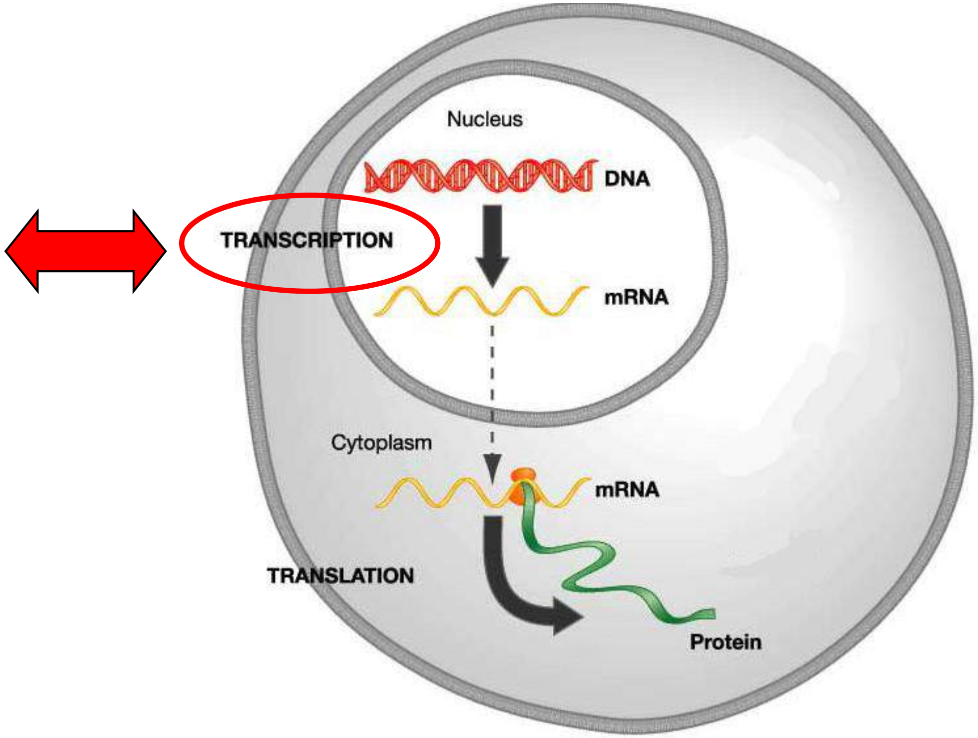
  Nodes represent small non-coding RNAs (miRNAs) or small interfering RNAs (siRNAs) and DNA regulatory elements. Links represent regulation.

- Databases:
  1. Predicted microRNA targets: TargetScan, PicTar, microRNA, miRBase, miRDB
  2. Experimentally supported targets: TarBase, miRecords

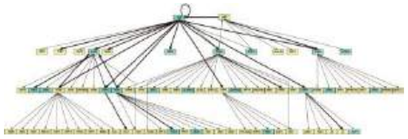
A. Paccanaro, 2019

Expression networks

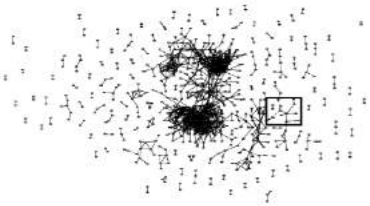




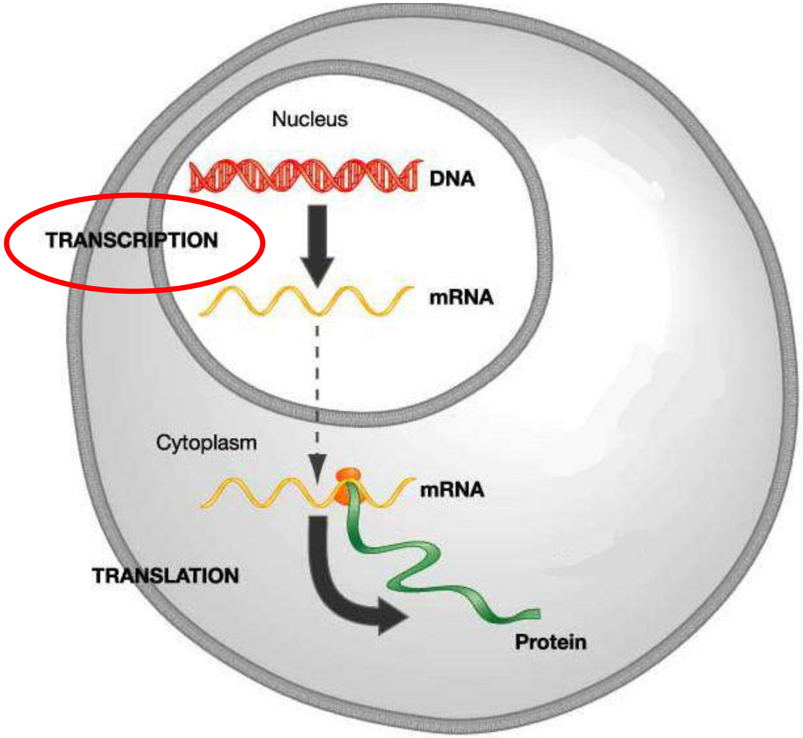
[from Qian, et al, J. Mol. Bio., 314:1053-1066]



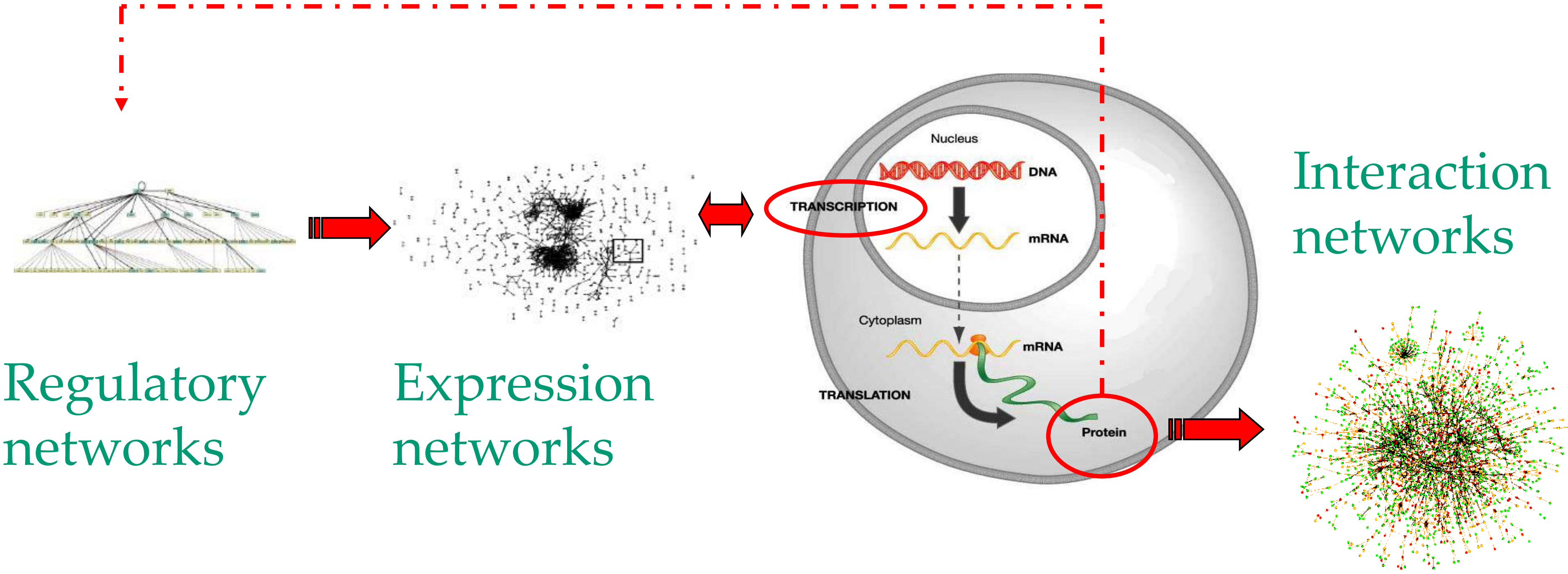
Regulatory networks



Expression networks



From Horak, Genes & Dev.; DeRisi, Science; Qian, J. Mol. Bio; Jeong, Nature



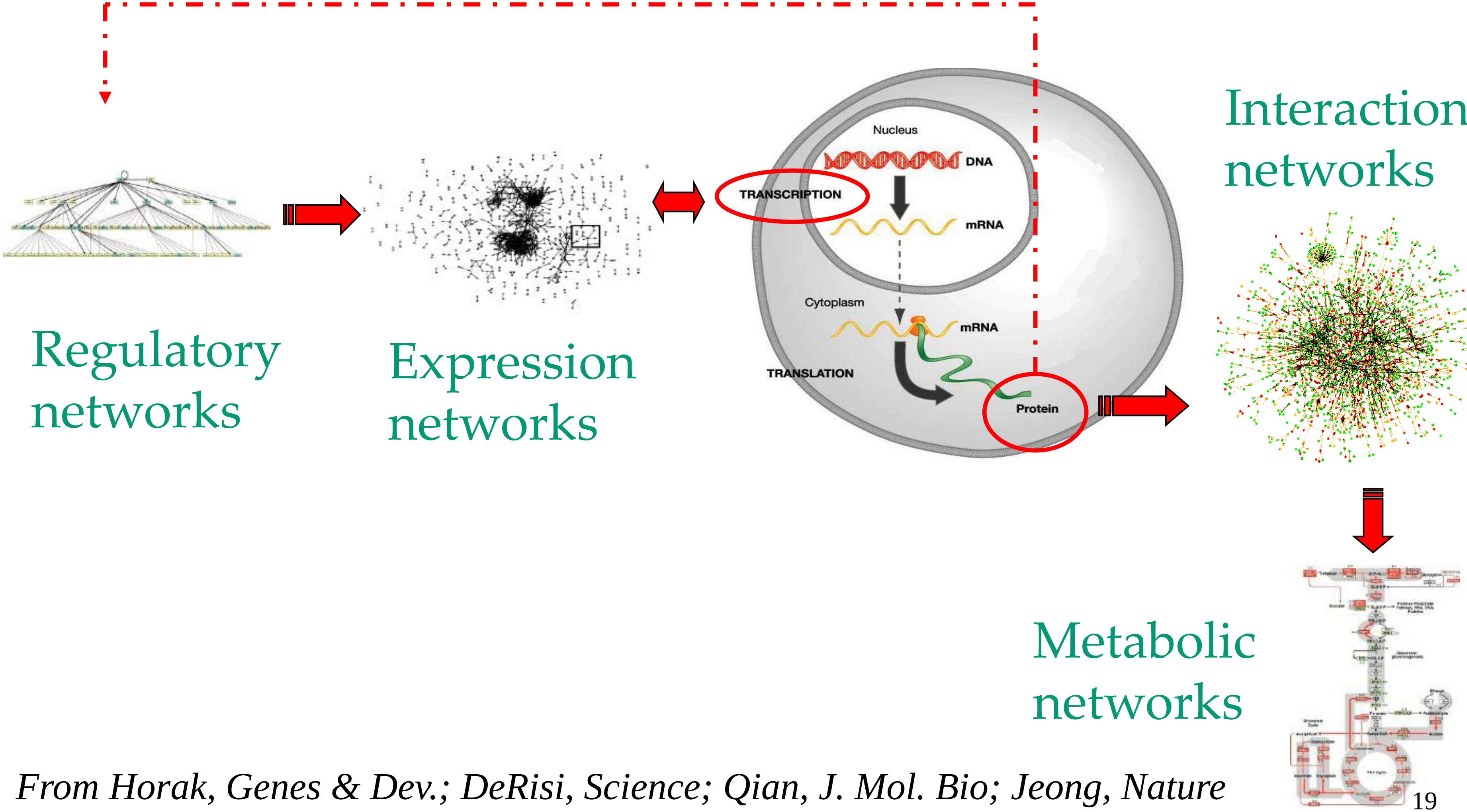
Regulatory networks

Expression networks

Interaction networks

A. Paccanaro, 2019

From Horak, Genes & Dev.; DeRisi, Science; Qian, J. Mol. Bio; Jeong, Nature



Regulatory networks

Expression networks

Interaction networks

Metabolic networks

From Horak, Genes & Dev.; DeRisi, Science; Qian, J. Mol. Bio; Jeong, Nature

A. Paccanaro, 2019

QUESTION: When I look at Human biological networks in terms of principles from network science, what do I see?

- Modules: high degree of clustering, implying the existence of topological modules that represent highly interlinked local regions in the network.

- Degree distribution: the degree distribution P(k) ~ $k^{-\gamma}$

A. Paccanaro, 2019

- Hubs: few highly connected hubs hold the whole network together.

  In protein interaction networks we have:

  - *‘party’ hubs*: interact with most of their partners simultaneously– they function inside modules and coordinate specific cellular processes
  - *‘date’ hubs*: bind different partners at different locations and times – they link together rather different processes and organize the interactome

  In protein interaction networks, **hub proteins tend to be encoded by essential genes**, and **genes encoding hubs are older and evolve more slowly** than genes encoding non-hub proteins

A. Paccanaro, 2019

- Small world phenomena: relatively **short paths** between any pair of nodes.

- Motifs: Some subgraphs (a group of nodes that link to each other, forming a small subnetwork within a network) in biological networks **appear more (or less) frequently than expected**

- Betweeness centrality: a measure of the number of shortest paths that go through each node.

  Nodes with high betweeness centrality are often called bottlenecks. In networks with directed edges, such as regulatory networks, **bottlenecks tend to correlate with essentiality**.

A. Paccanaro, 2019

Data Science

São Paulo School of Advanced Science on Learning from Data

USP

Network Medicine

Alberto Paccanaro
Department of Computer Science
Royal Holloway, University of London
www.paccanarolab.org

Genotype, phenotype & hereditary disease

Human disease cannot be explained by simple genotype-phenotype relationships

- Many genes linked to the same disease
  (e.g. hundreds of genes linked to cancer)

- One gene linked to many diseases
  (e.g. genes related to diabetes, obesity and hypertension)

A. Paccanaro, 2019

We follow an excellent review: A.L.Barabasi, et al. *Nature Review Genetics*, 2011

QUESTION: When I map our current knowledge of Human disease onto the Human biological networks, and I analyze it in terms of principles from network science, what do I see?

A. Paccanaro, 2019

Principles of Network Medicine

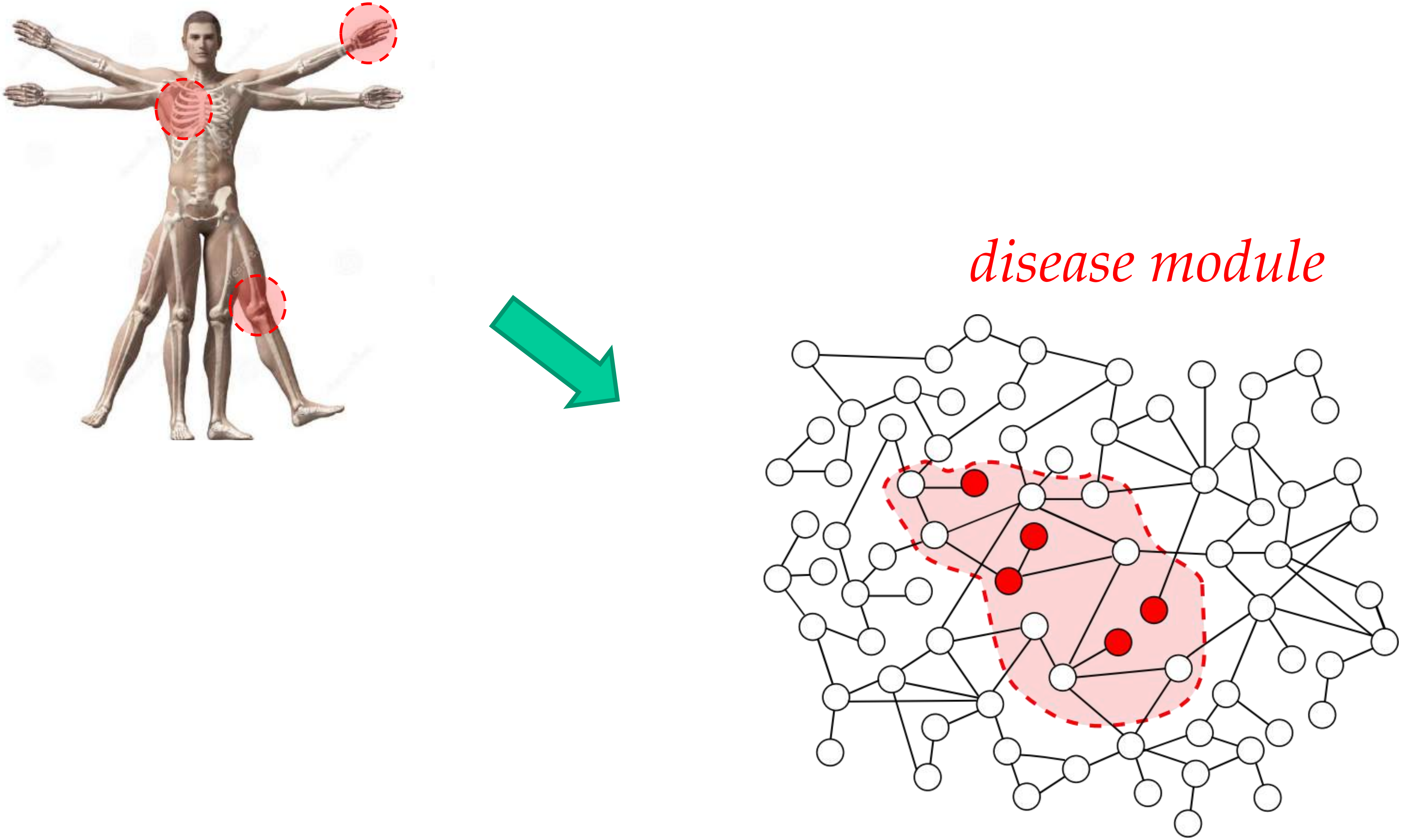
A. **Hubs: disease genes tend to avoid hubs** and segregate at the functional periphery of the interactome. In humans essential genes, not disease genes are encoded in hubs.

B. **Local hypothesis:** if a gene or molecule is involved in a disease, its **direct interactors might also be suspected to have some role** in the same disease.

➔ *Proteins involved in the same disease have an increased tendency to interact with each other.*

A. Paccanaro, 2019

Gene associated with a specific disease tend to cluster in the same neighbourhood



A. Paccanaro, 2019

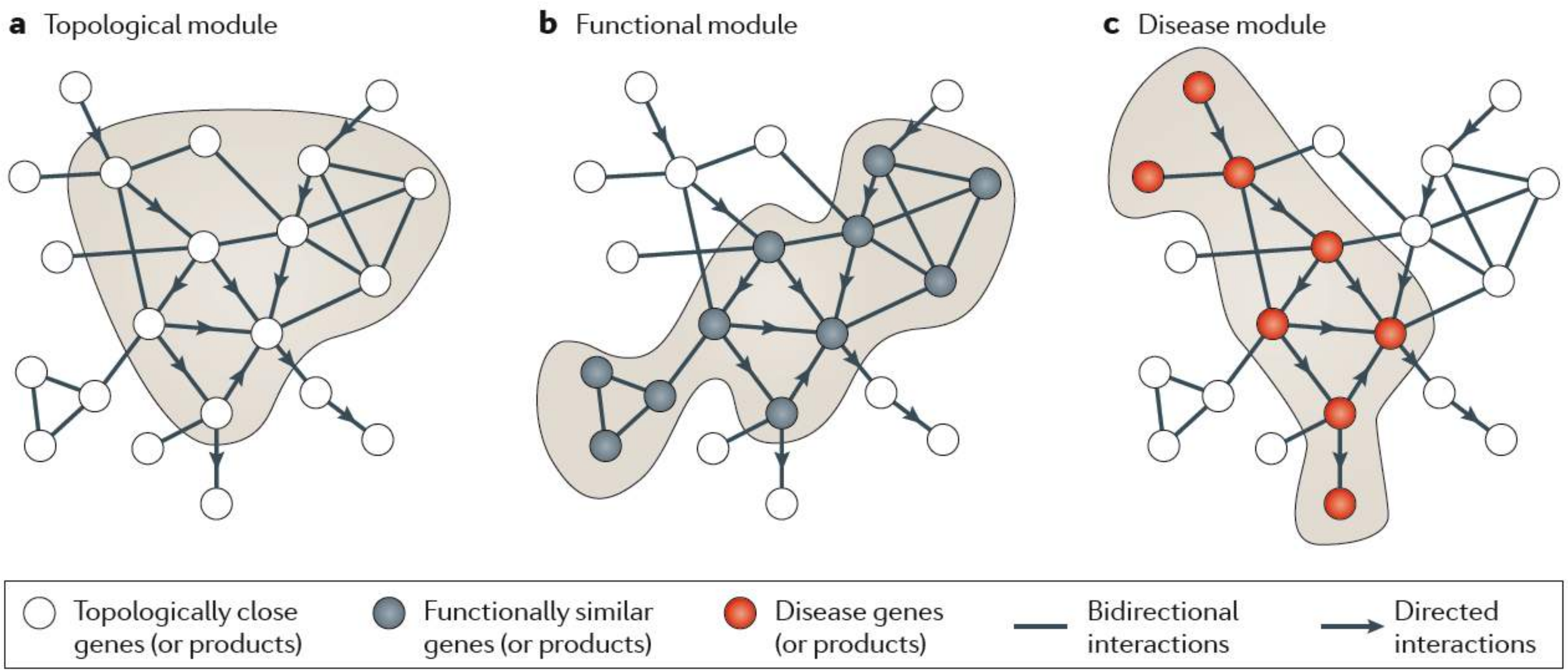
3 modules

1. **‘topological module’:** a locally dense neighbourhood in a network, such that nodes have a higher tendency to link to nodes within the same local neighbourhood than to nodes outside it.

2. **‘functional module’:** nodes of similar or related function (~phenotype) in the same network neighbourhood.

3. **‘disease module’:** a group of network components that together contribute to a cellular function and disruption of which results in a particular disease phenotype.

A. Paccanaro, 2019


a Topological module
b Functional module
c Disease module
Topologically close genes (or products)
Functionally similar genes (or products)
Disease genes (or products)
Bidirectional interactions
Directed interactions

From A.L.Barabasi, N.Gulbahce, J.Loscalzo, Nature Review Genetics, Vol. 12 (2011)

A. Paccanaro, 2019

These three concepts are interrelated

Note that…

- a **disease module** may not be identical to, but is likely to **overlap** with, the **topological** and/or **functional** modules.

- a disease module is defined in relation to a particular disease and, accordingly, **each disease has its own unique module**.

- a gene, protein or metabolite can be implicated in several disease modules, which means that different **disease modules can overlap**.

A. Paccanaro, 2019

C. **Corollary of the local hypothesis:** Mutations in interacting proteins often lead to similar disease phenotypes.

D. **Shared components hypothesis:** Diseases that share disease-associated cellular components (genes, proteins, metabolites or microRNAs) show phenotypic similarity and comorbidity.

A. Paccanaro, 2019

In other words, Network Medicine…

Network medicine: a network-based approach to human disease
Barabási et al., *Nature Review Genetics*, 2011

- **Cellular components exerts their functions through interactions with other cellular components**

- This interconnectivity means that the **impact of the abnormality in a gene is not limited to that gene**. The effects of this abnormality will be propagated to other elements in the networks which do not have abnormalities.

- An **understanding of a gene's network context is essential** in determining the phenotypic impact of defects that affect it.

A. Paccanaro, 2019

What applications have these ideas had so far?

1. Methods for disease gene prediction
2. The human diseasome
3. Implications for Network pharmacology

A. Paccanaro, 2019

1. Methods for Disease Gene Prediction

Genes in the neighbourhood of known disease genes for a given disease, are likely to be disease genes (for that disease)

1. Direct Linkage methods: predict genes that are *direct interactors* of known disease genes
2. Diffusion based methods: predict genes «*highly connected*» to known disease disease genes (more on this later)
3. Disease module-based methods: start by identifying the disease modules, and inspect their members as potential disease genes.

A. Paccanaro, 2019

2. The human diseasomes

- At the molecular level, it is difficult to consider diseases as being consistently independent of one another.

- Different disease modules can overlap.

- **Diseasome**: disease maps whose **nodes are diseases** and whose links represent various molecular relationships between the disease-associated cellular components.

A. Paccanaro, 2019

Why is this important…

- To understand **how different phenotypes**, often addressed by different medical subdisciplines, **are linked** at the molecular level

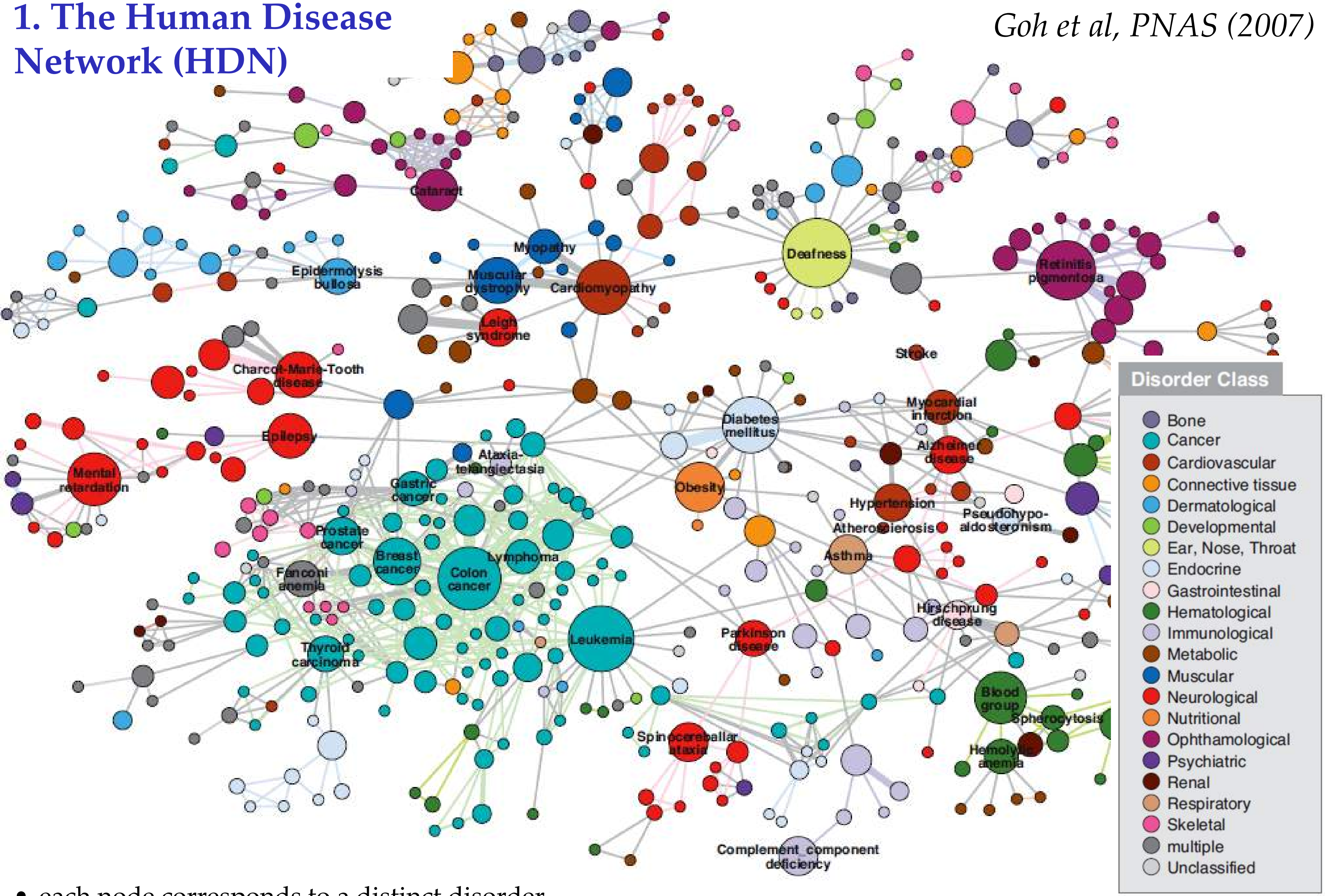
- To understand why certain groups of diseases arise together (**comorbidity**)

- To **aid drug discovery**, in particular when it comes to the use of approved drugs to treat molecularly linked diseases.

A. Paccanaro, 2019

1. The Human Disease Network (HDN)

Goh et al, PNAS (2007)



- each node corresponds to a distinct disorder
- size of each node is proportional to the number of genes participating in the corresponding disorder
- the link thickness is proportional to the number of genes shared by the disorders it connects.

2. Phenotypic disease networks (PDN)

- Phenotypic disease networks are diseasomes which are generated by **analyzing disease phenotypes**.

- Clearly, these are important when the phenotype is used to create **links which correspond to real relationships at the level of molecular network** (we will see an example of this later)

A. Paccanaro, 2019

3. Network pharmacology

- **reduce the search for therapeutic agents to those that induce detectable changes in disease module activity**.
- A **drug might have more than one binding partner** such that its efficacy is determined by its multiple interactions, leading to unwanted **side effects**
- therapies that involve multiple targets, which may be more effective than are single drugs – **drug cocktails**

Open question: can one systematically identify multiple drug targets that have an optimal impact on the disease phenotype?

A. Paccanaro, 2019

References

(from which I took some figures)

- A.L.Barabasi, N.Gulbahce, J.Loscalzo,
  Nature Review Genetics, Vol. 12 (2011)

- M. Vidal, M. E. Cusick, A.L. Barabási
  Interactome Networks and Human Disease
  Cell, Vol 144, 6, p986-998 (2011)

- X. Wang, N. Gulbahce, H. Yu
  Network-based methods for human disease gene prediction
  Briefings in Functional Genomics, Volume 10, 5 (2011)

A. Paccanaro, 2019

Data Science

São Paulo School of Advanced Science on Learning from Data

USP

Material in the following slides taken from :
Caniza, Romero, Paccanaro
Nature Scientific Reports, 2015

Quantifying the distance between disease modules on the interactome

Alberto Paccanaro

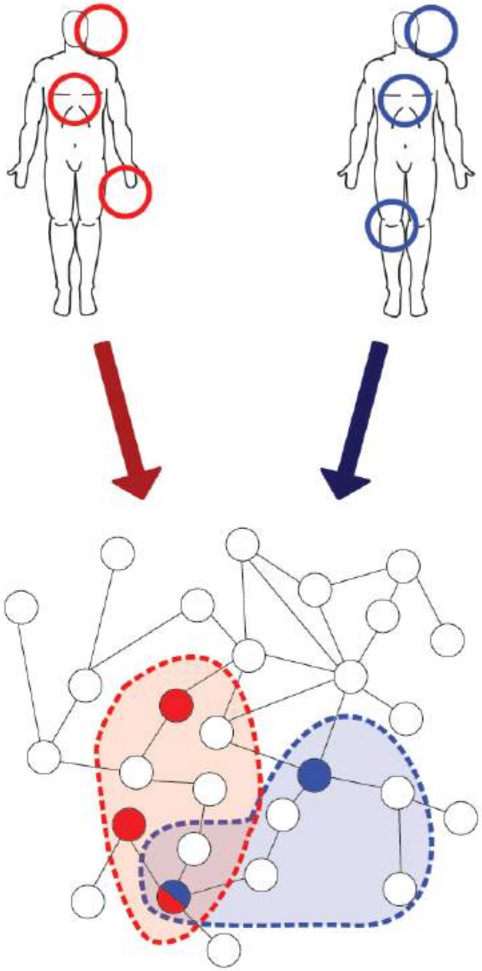
Department of Computer Science
Royal Holloway, University of London

www.paccanarolab.org

A. Paccanaro, 2019

Network Medicine: Disease as perturbations of molecular networks

Protein-protein interaction networks



Genes associated with a specific disease tend to cluster in the same neighbourhood – the disease module

The disease modules of diseases that are phenotypically similar tend to be located in closeby regions of the interactome.

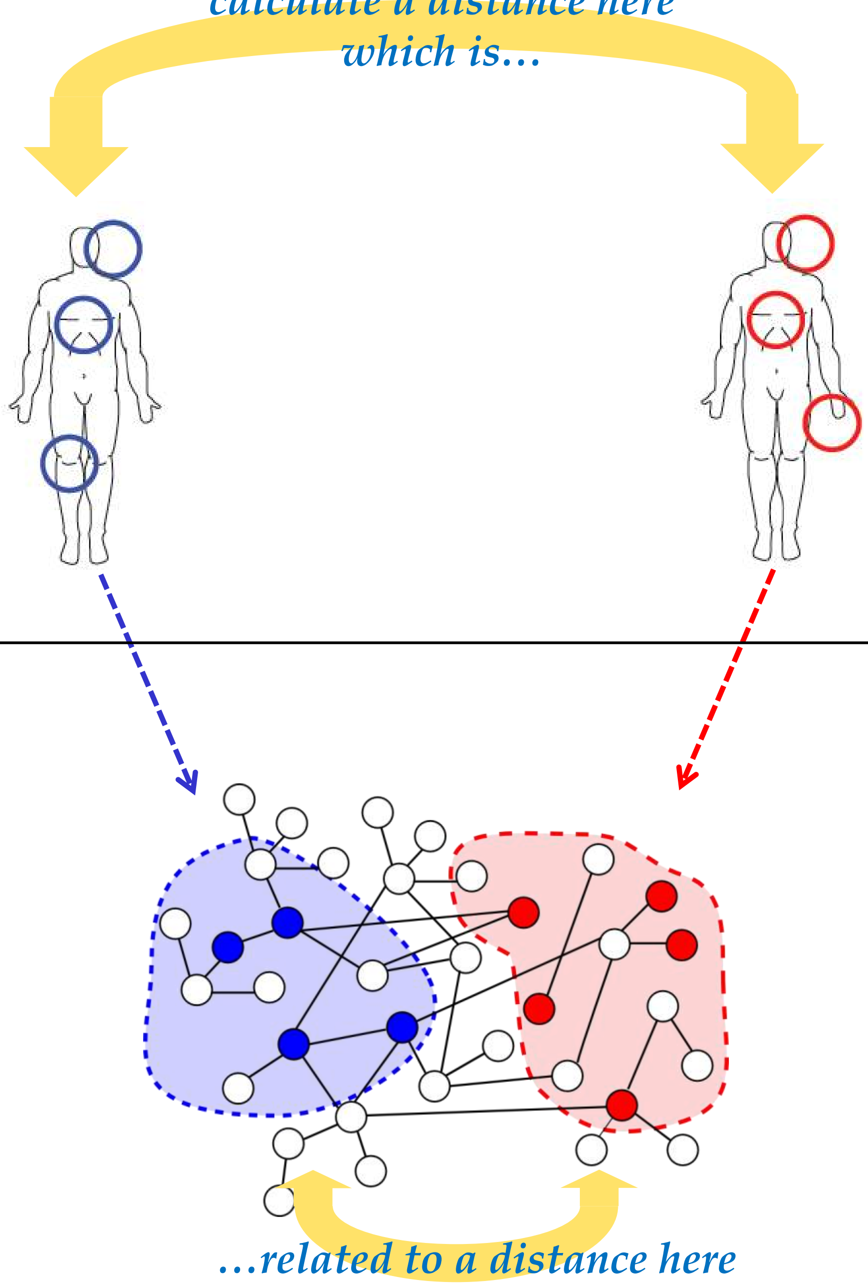
Question

*Define a "**distance**" between diseases using the disease phenotypes*

such that

it is related to the distance between disease modules

The problem



calculate a distance here which is…

Phenotype

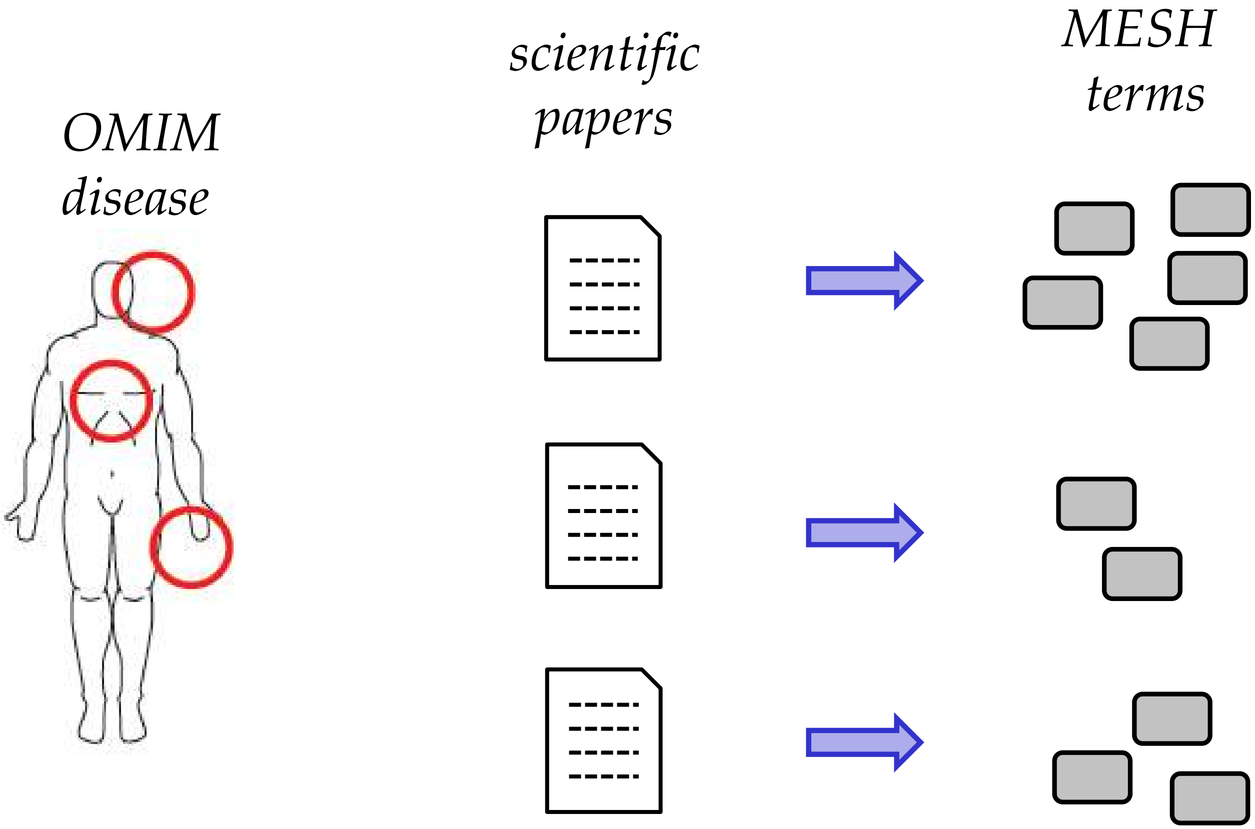
Genotype

…related to a distance here

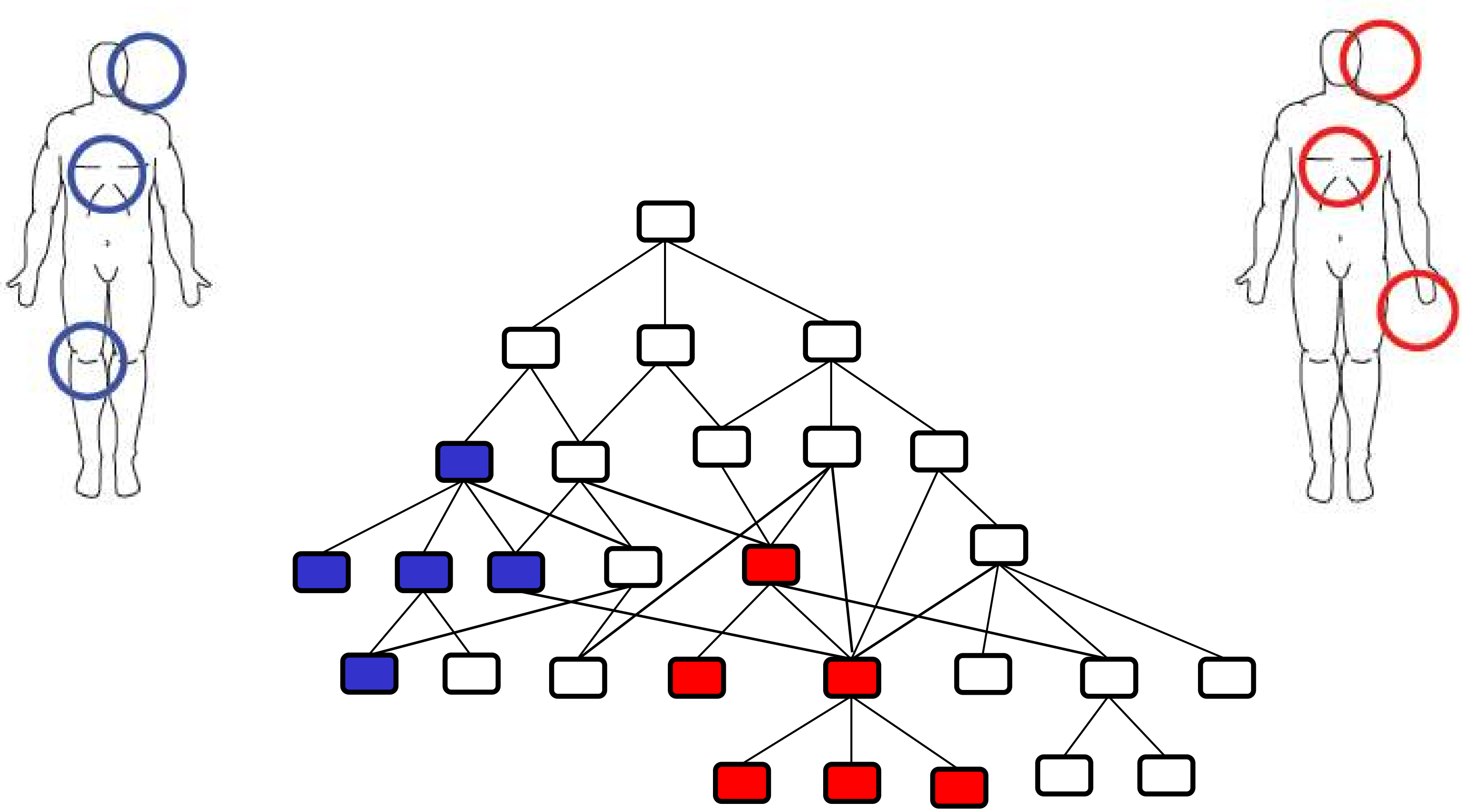
Outline of the method

[Caniza, Romero, Paccanaro, *Nature Scientific Reports*, 2015]

STEP 1: Translate a genetic disease into a set of MeSH terms



STEP 2: quantify a distance between two sets of terms on an ontology



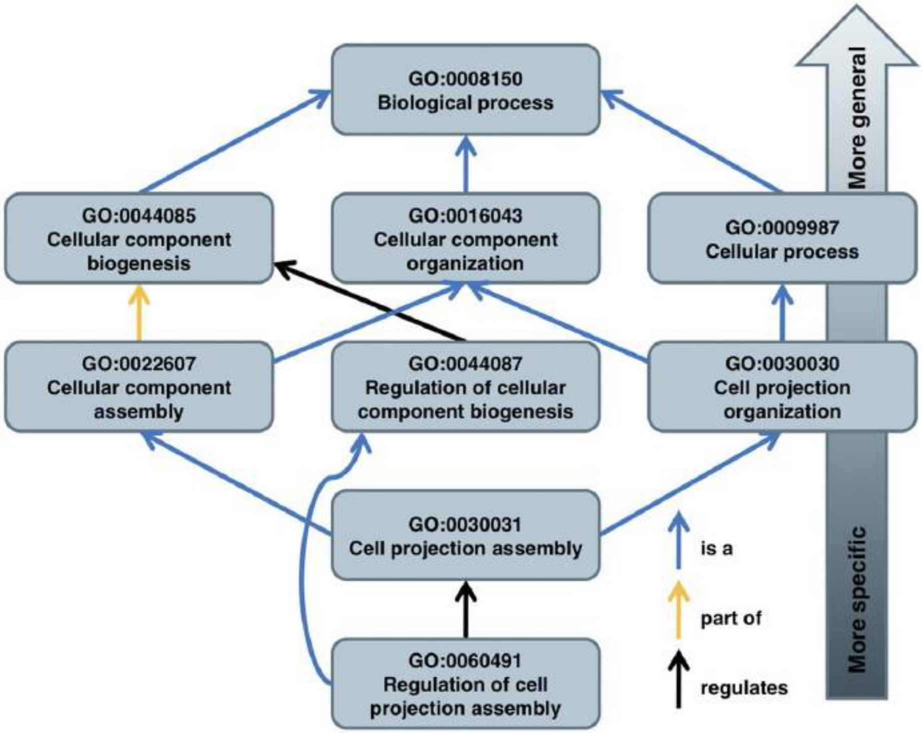
Luckily ☺ , we had developed a measure for that !

(Yang et al, *Bioinformatics*, 2012; Caniza et al, *Bioinformatics*, 2014)

Semantic Similarity on the Gene Ontology

Gene Ontology

- A structured vocabulary of functional labels
- Genes are assigned to nodes (functional labels)
- Inheritance of labels

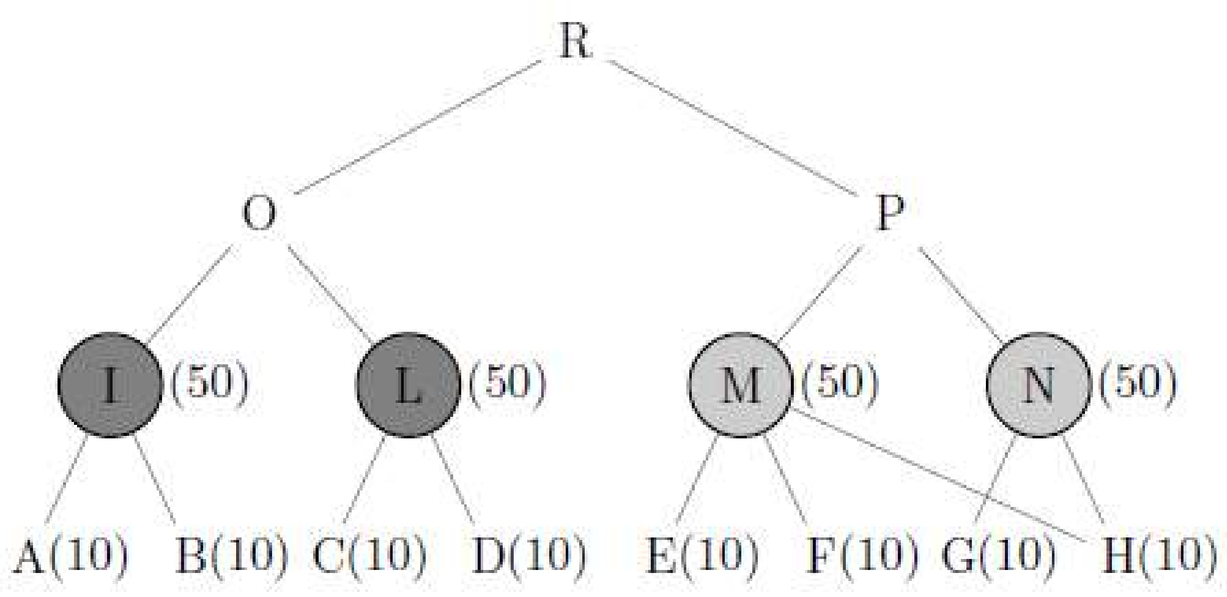

GO:0008150
Biological process
GO:0044085
Cellular component biogenesis
GO:0016043
Cellular component organization
GO:0009987
Cellular process
GO:0022607
Cellular component assembly
GO:0044087
Regulation of cellular component biogenesis
GO:0030030
Cell projection organization
GO:0030031
Cell projection assembly
GO:0060491
Regulation of cell projection assembly
is a
part of
regulates
More general
More specific

du Plessis, Brief Bioinf. 2011

Problem: evaluate the similarity between genes (or group of genes) in terms of their functional assignments

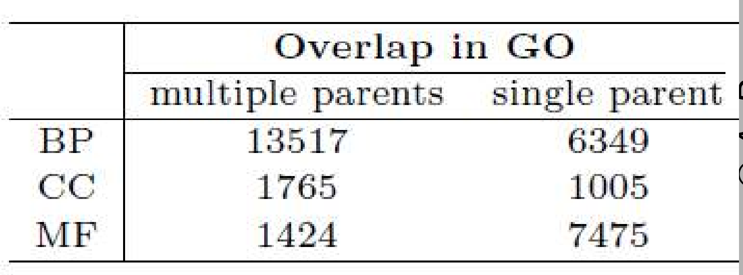
Methods use the Information Content of the Lowest Common Ancestor

The roles of descendants when calculating semantic similarities on DAGs



sim(M,N) > sim(I,L)

Yang et al, Bioinformatics, 2012
Caniza et al, Bioinformatics 2014
http://www.paccanarolab.org/gosstoweb/



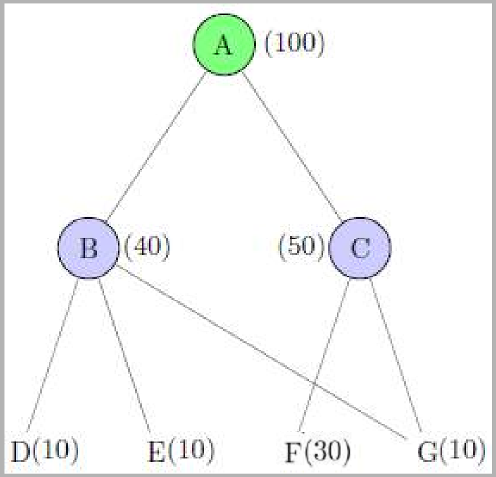
	Overlap in GO	
	multiple parents	single parent
BP	13517	6349
CC	1765	1005
MF	1424	7475

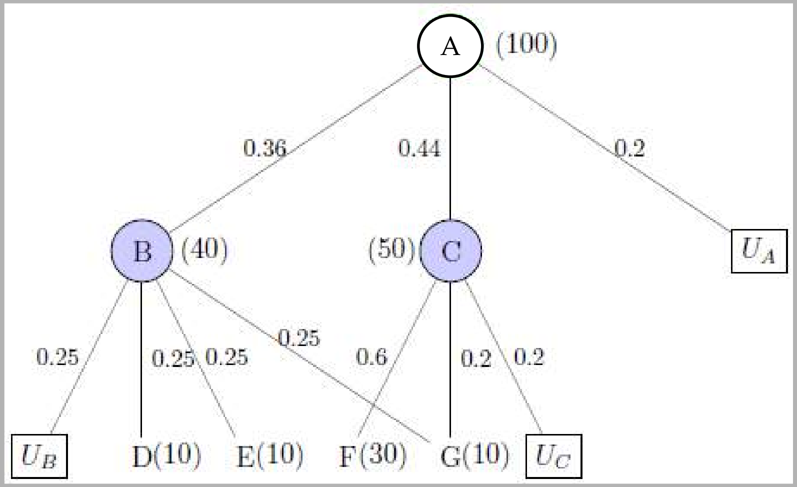
- ***Our idea**: decompose the semantic similarity of two terms into a weighted sum of the semantic similarities of their descendant leaf terms*

Host Similarity Measure, Random Walk Contribution

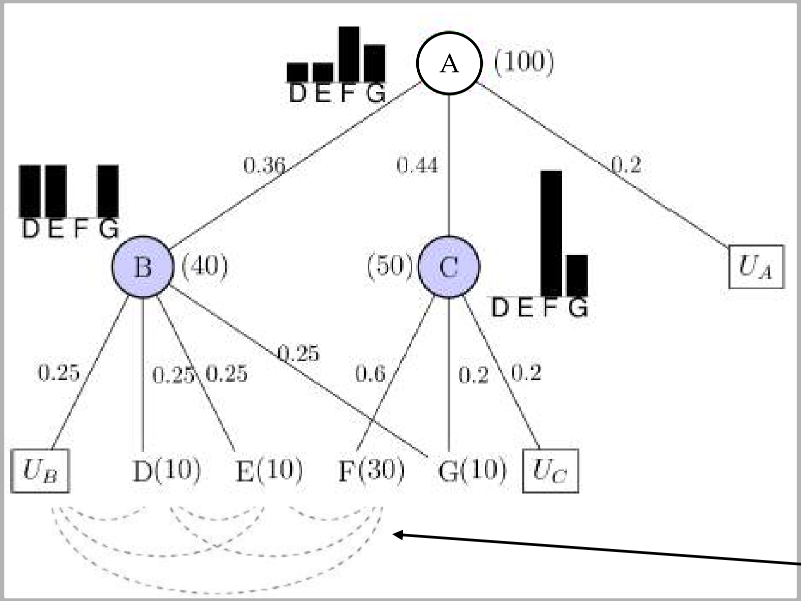
Yang et al, Bioinformatics, 2012
Caniza et al, Bioinformatics 2014
http://www.paccanarolab.org/gosstoweb/

Host Similarity Measure
HSM (upward)





Random Walk Contribution
RWC (downward)

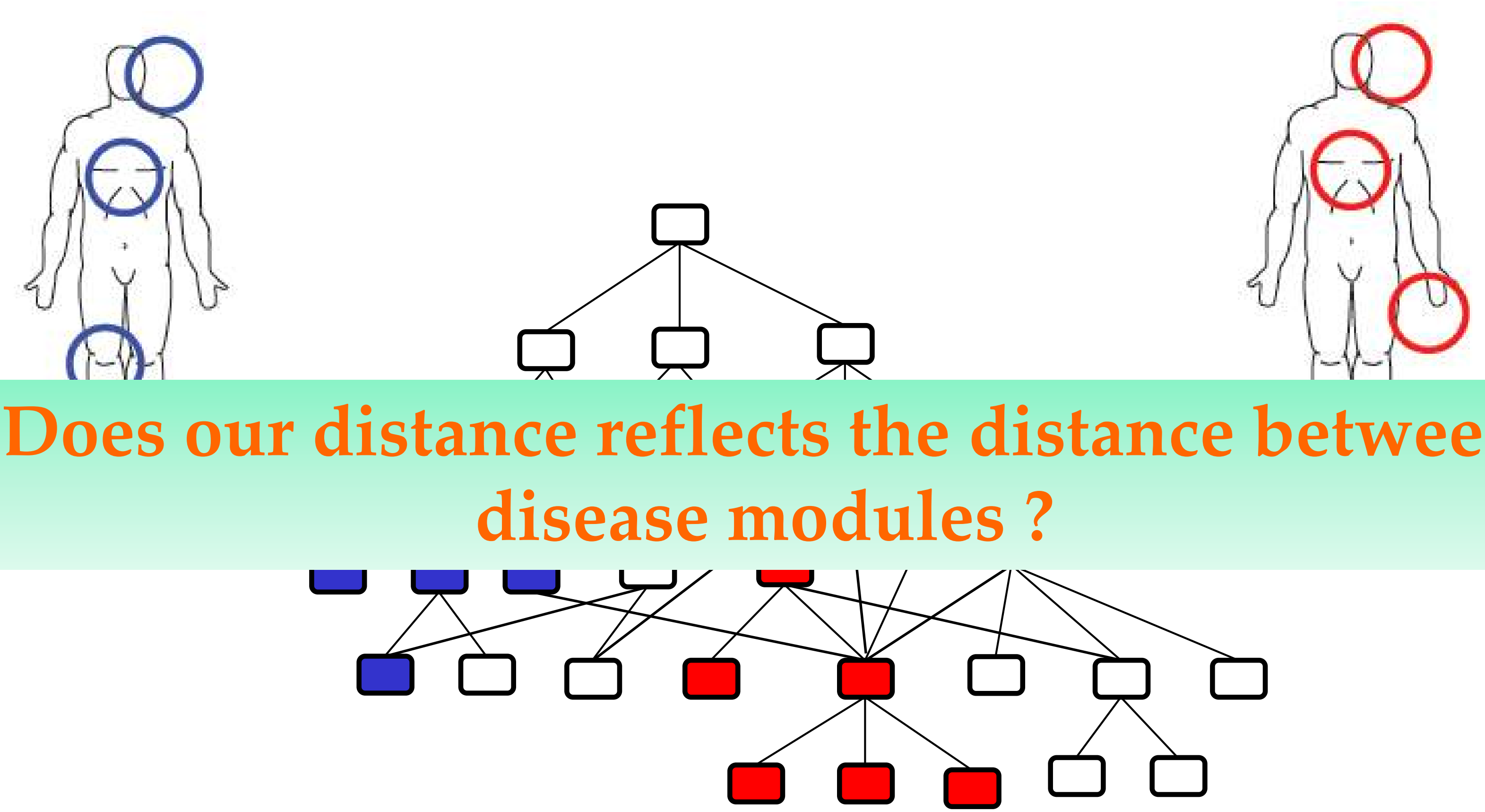


existence of common descendants
uncertainty
affect the random walk

HSM between every pair of leaves weighted by their probabilities

© A. Paccanaro, 2019

STEP 2: quantify a distance between two sets of terms on an ontology



Does our distance reflects the distance between disease modules ?

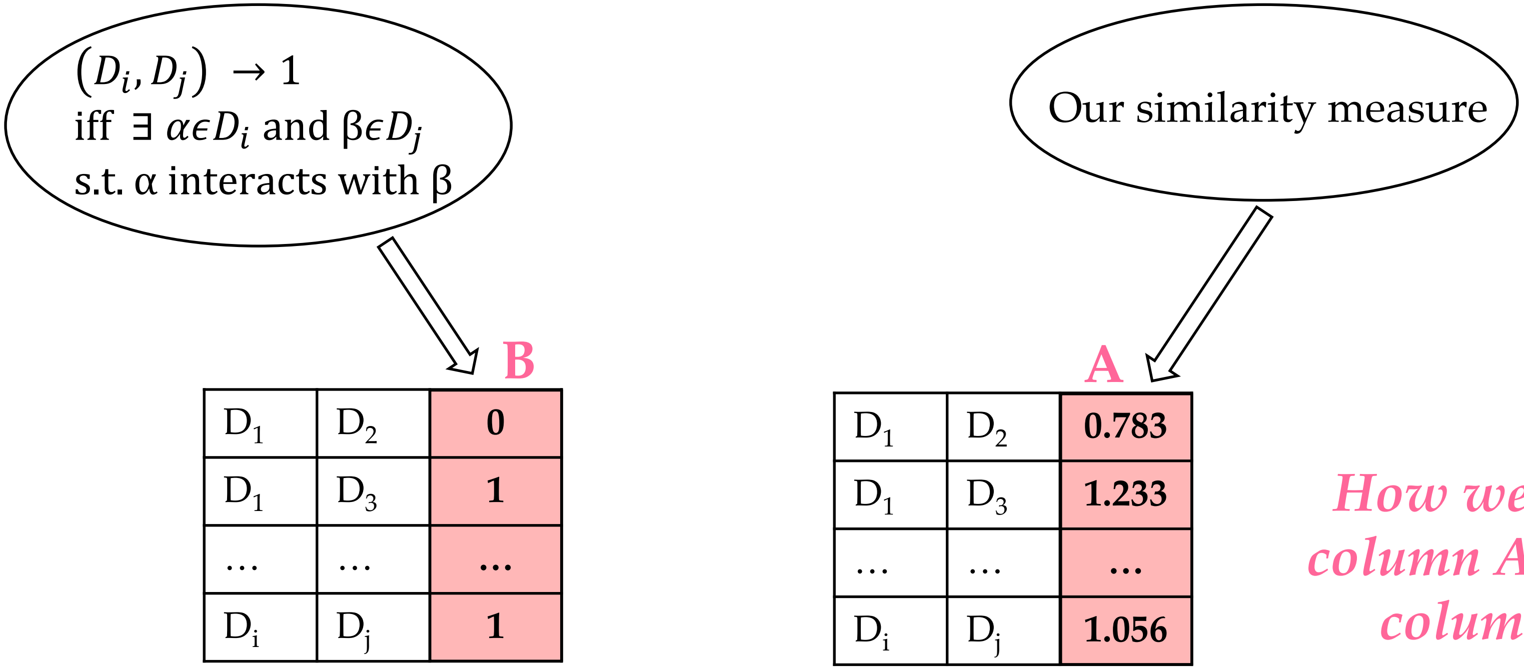
© A. Paccanar

Luckily ☺ , we had developed a measure for that !

(Yang et al, *Bioinformatics*, 2012; Caniza et al, *Bioinformatics*, 2014)

1. Evaluation as a prediction problem

A. **Diseases related by <u>physical interactions</u> (PPI) of diseases proteins**



$(D_i, D_j) \rightarrow 1$
iff $\exists\ \alpha \epsilon D_i$ and $\beta \epsilon D_j$
s.t. α interacts with β

Our similarity measure

B

D_1	D_2	**0**
D_1	D_3	**1**
…	…	**…**
D_i	D_j	**1**

A

D_1	D_2	**0.783**
D_1	D_3	**1.233**
…	…	**…**
D_i	D_j	**1.056**

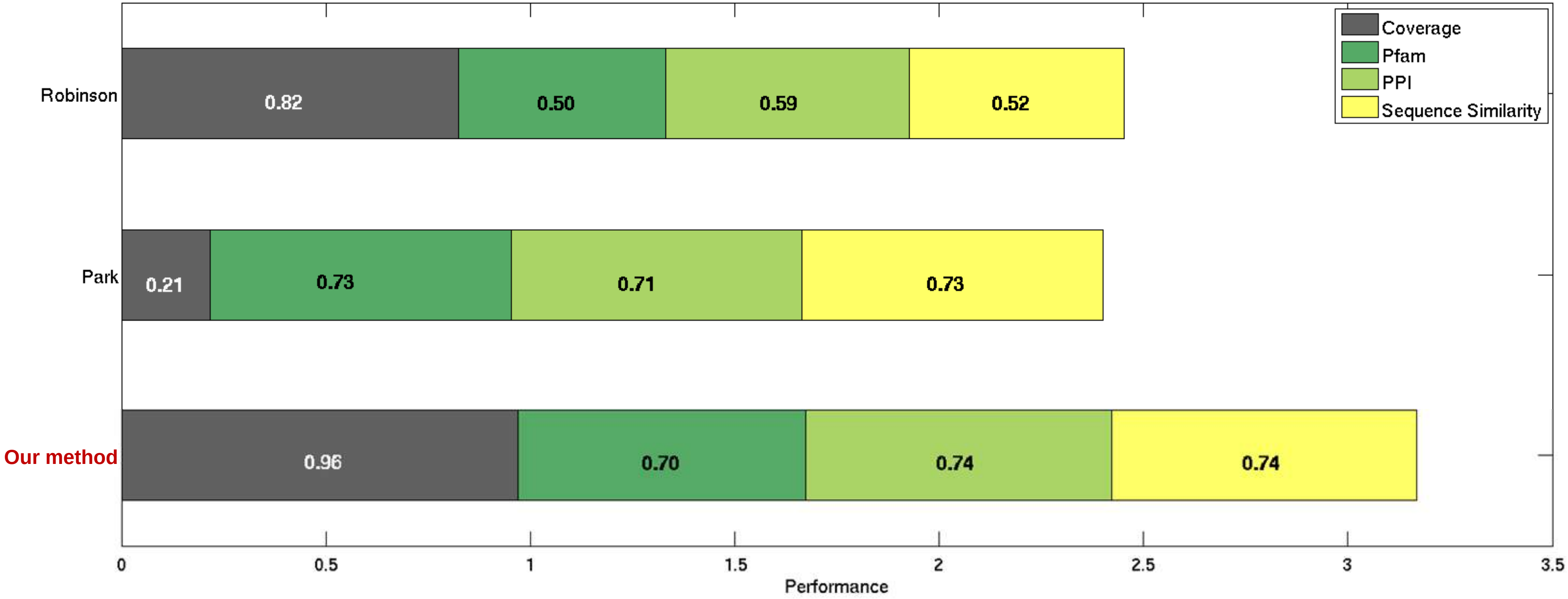
How well does column A predict column B?

B. **Diseases related by <u>sequence similarity</u> of disease proteins**

C. **Diseases related by <u>evolutionary relatedness of disease proteins</u> (Pfam)**

D. **<u>Coverage</u> (% of OMIM diseases)**

Results of AUC analysis



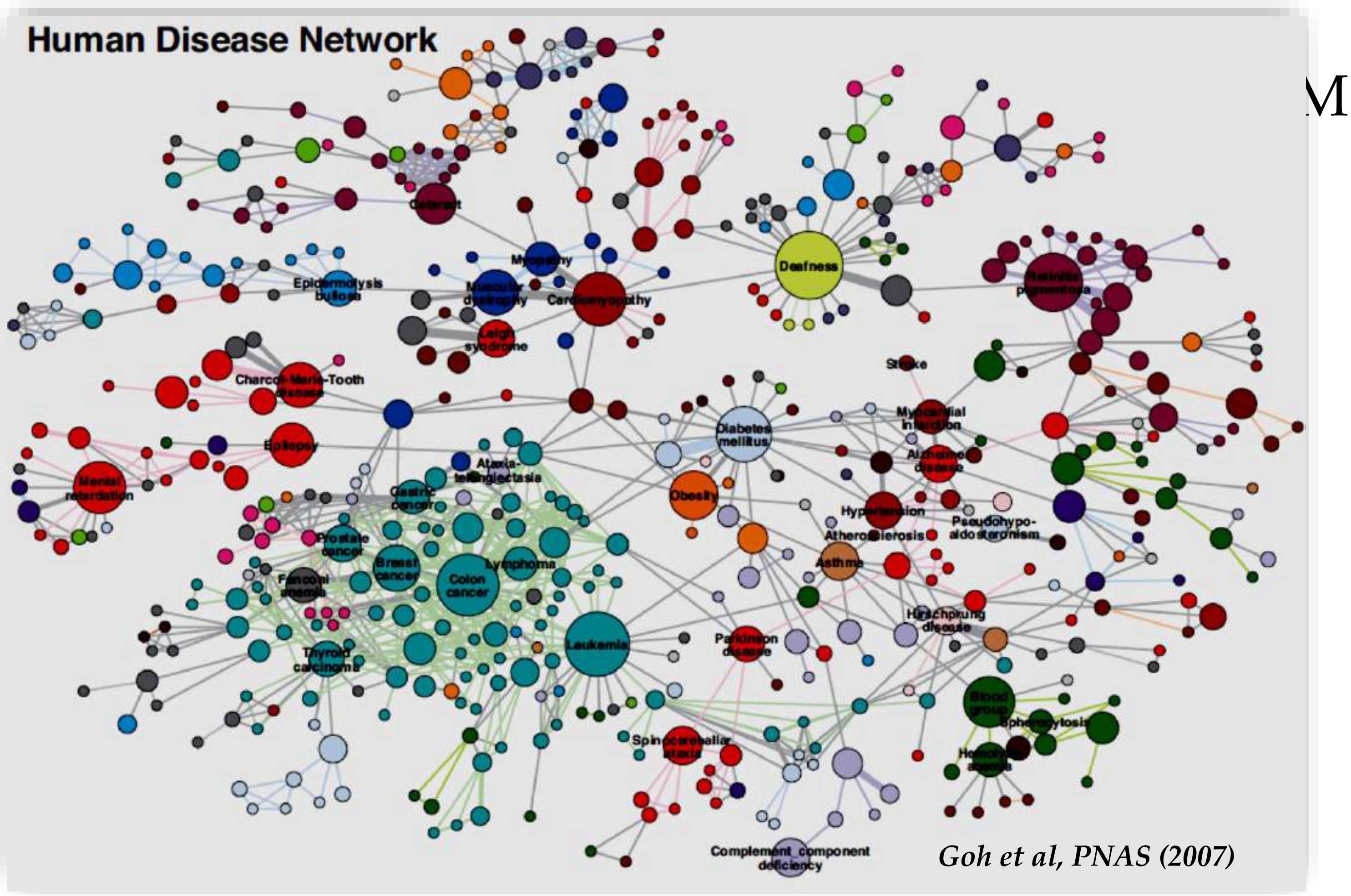
Robinson : builds and ad-hoc diseases ontology (**Human Phenotype Ontology**) and then calculates a distance on it (Köhler et al, NAR, 2013)

Park : similarity between two diseases is determined by an association score based on the **cellular co-localisation** of their disease proteins (Park et al, Mol. Sys. Bio. 2011)

2. Embedding diseases in low dimensional space

1) M

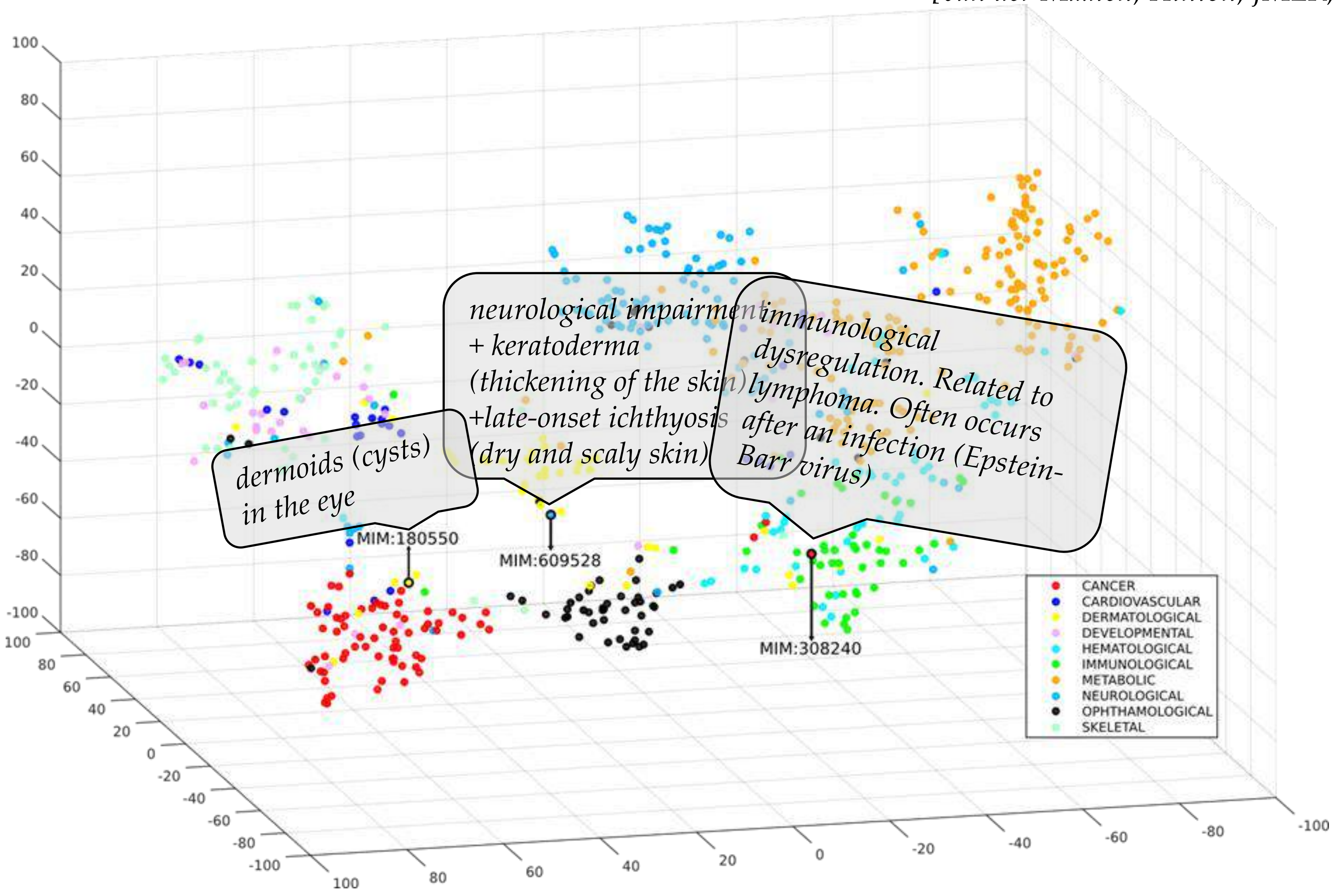
2)


Human Disease Network

Goh et al, PNAS (2007)

Embedding diseases in 3D using *t*-SNE

[van der Maaten, Hinton, JMLR, 2008]



dermoids (cysts) in the eye

neurological impairment + keratoderma (thickening of the skin) +late-onset ichthyosis (dry and scaly skin)

immunological dysregulation. Related to lymphoma. Often occurs after an infection (Epstein-Barr virus)

MIM:180550 - Ring Dermoid of Cornea – cancer/dermatological/ophthalmological

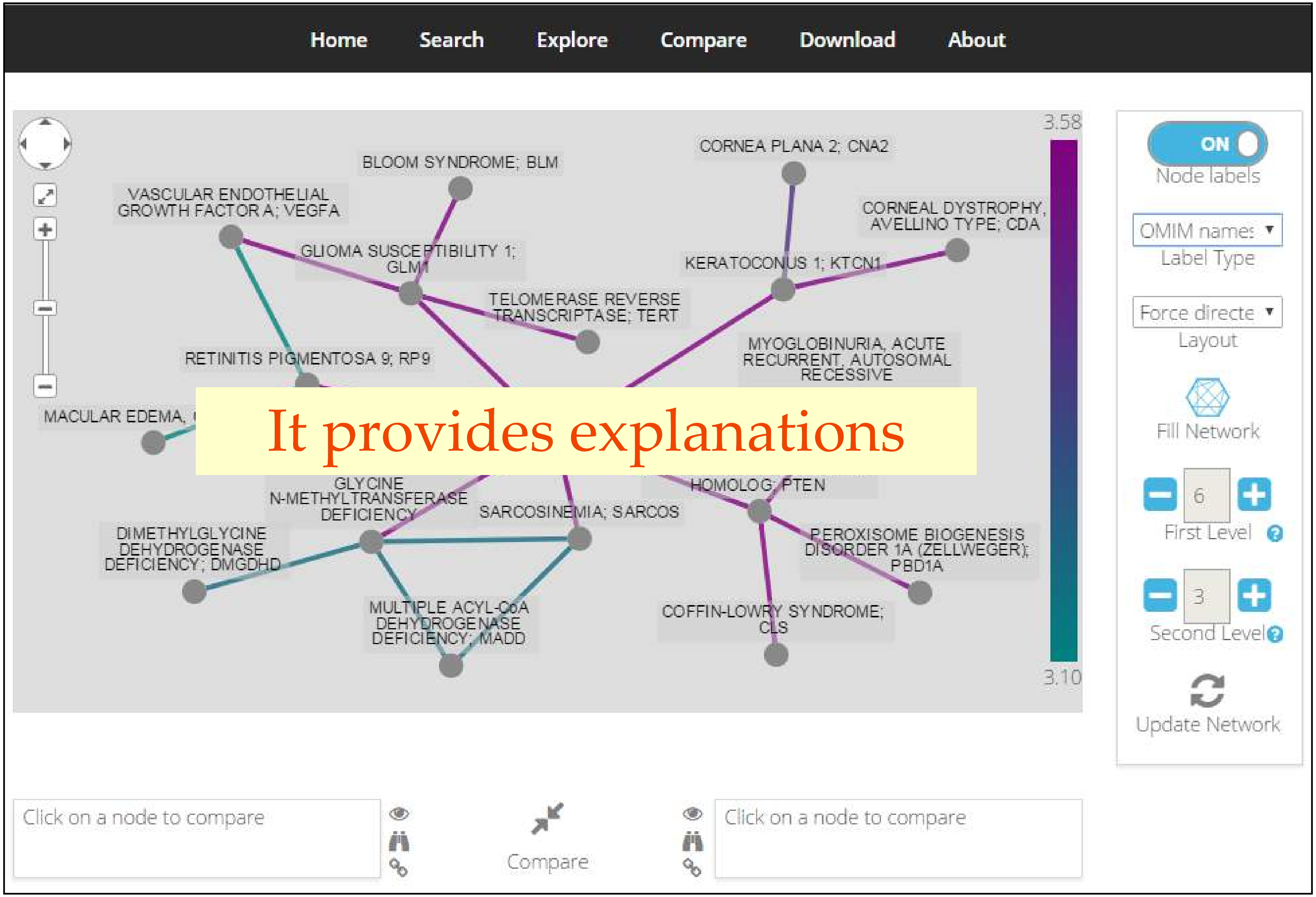
MIM:609528 - Cerebral dysgenesis, neuropathy, ichthyosis, and palmoplantar keratoderma syndrome – neurol./dermatol.

MIM:308240 - Lymphoproliferative syndrome – cancer/immunological

Landis – the Landscape of Disease Similarities

http://www.paccanarolab.org/landis

Differential diagnoses



Home
Search
Explore
Compare
Download
About
3.58
3.10
BLOOM SYNDROME; BLM
CORNEA PLANA 2; CNA2
VASCULAR ENDOTHELIAL GROWTH FACTOR A; VEGFA
CORNEAL DYSTROPHY, AVELLINO TYPE; CDA
GLIOMA SUSCEPTIBILITY 1; GLM1
KERATOCONUS 1; KTCN1
TELOMERASE REVERSE TRANSCRIPTASE; TERT
MYOGLOBINURIA, ACUTE RECURRENT, AUTOSOMAL RECESSIVE
RETINITIS PIGMENTOSA 9; RP9
MACULAR EDEMA, (
GLYCINE N-METHYLTRANSFERASE DEFICIENCY
HOMOLOG; PTEN
SARCOSINEMIA; SARCOS
DIMETHYLGLYCINE DEHYDROGENASE DEFICIENCY; DMGDHD
PEROXISOME BIOGENESIS DISORDER 1A (ZELLWEGER); PBD1A
MULTIPLE ACYL-CoA DEHYDROGENASE DEFICIENCY; MADD
COFFIN-LOWRY SYNDROME; CLS
Fill Network
Update Network
Click on a node to compare
Compare
Click on a node to compare

Node labels: ON

Label Type: OMIM names

Layout: Force directe

First Level: 6

Second Level: 3

It provides explanations

Data Science

São Paulo School of Advanced Science on Learning from Data

USP

Material in the following slides taken from :
J.J. Caceres, A. Paccanaro,
***PLoS Comp. Biology*, 2019**

Using disease distances to predict disease genes for Uncharted Diseases

Alberto Paccanaro

Department of Computer Science
Royal Holloway, University of London

www.paccanarolab.org

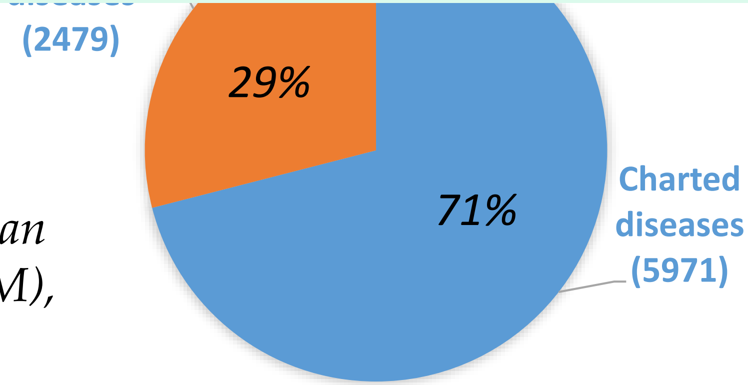
A. Paccanaro, 2019

Disease gene prediction

- **Charted** diseases: some disease genes are known
- **Uncharted diseases:** no known disease genes

Disease gene prediction for charted diseases: search in a neighbourhood of known disease genes

Can we use our disease similarity measure for predicting disease genes for uncharted diseases ?

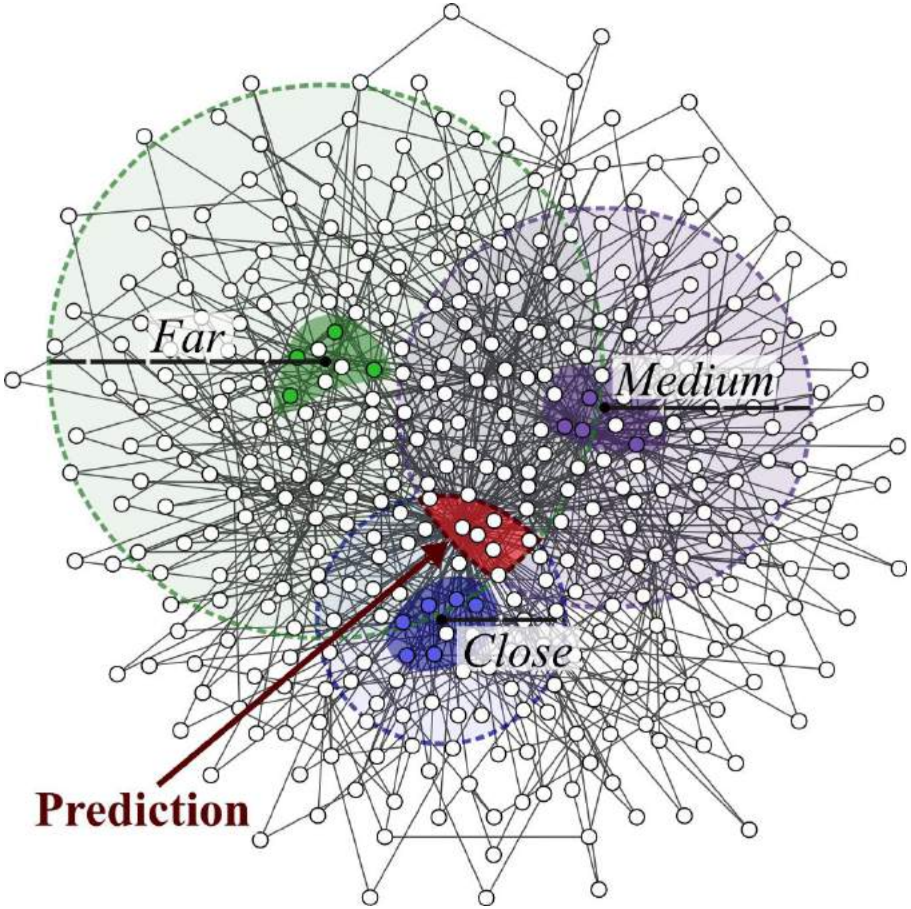


Data from Online Mendelian Inheritance in Man (OMIM), Sept 2018

A. Paccanaro, 2019

Predicting genes for *uncharted* diseases – the idea

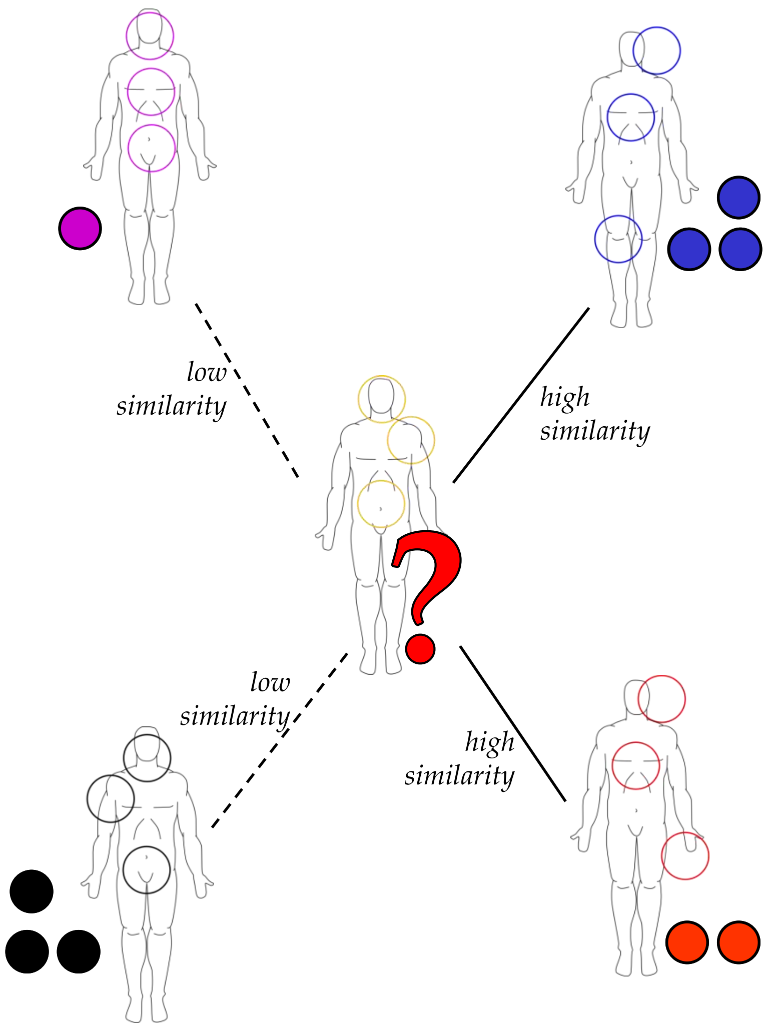
<u>Triangulation</u>: a mobile phone is detected within a radius from each of the towers.

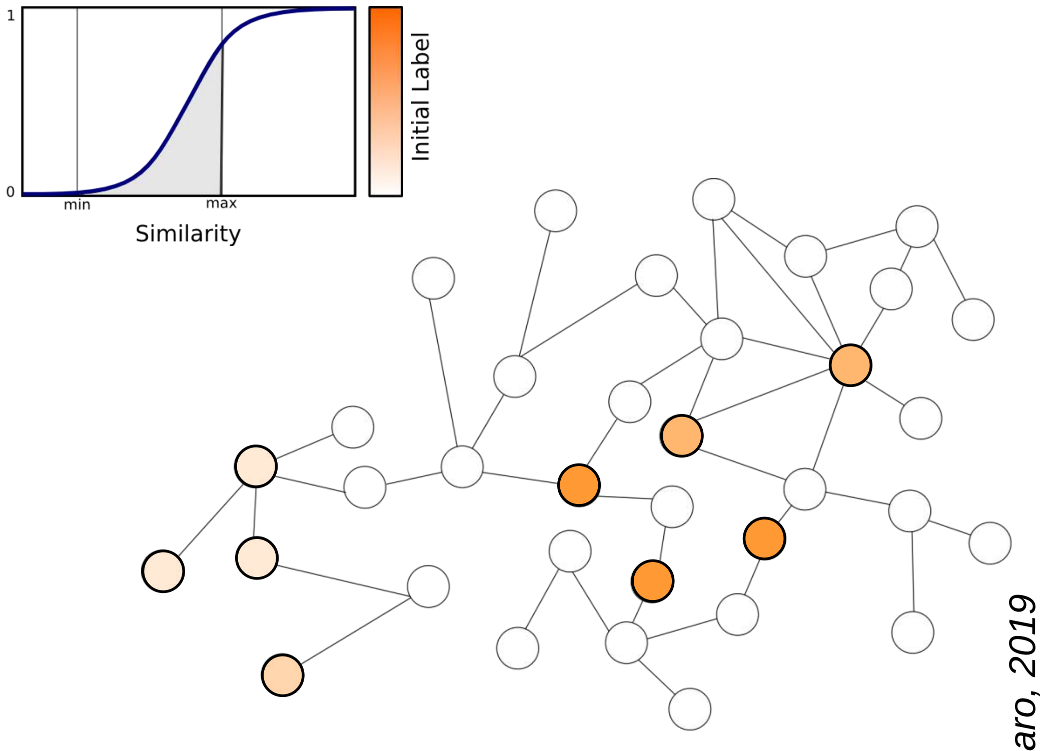


A. Paccanaro, 2019

A new disease gene prediction algorithm

soft labels + diffusion





1. Calculate the similarity between our uncharted disease and each charted disease
2. Place known genes in the interactome.
3. Learn a *similarity-to-label* mapping
4. Assign a *"soft"* label to the disease genes
5. Diffuse the soft labels

A. Paccanaro, 2019

Diffusing soft labels (semi-supervised learning)

For a given disease, the soft label is related to the ***probability for that gene to be a disease gene for that disease.***

Interacting nodes have similar labels

Preserve initial labelling

$$F^* = \arg\min_{F} Q(F)$$

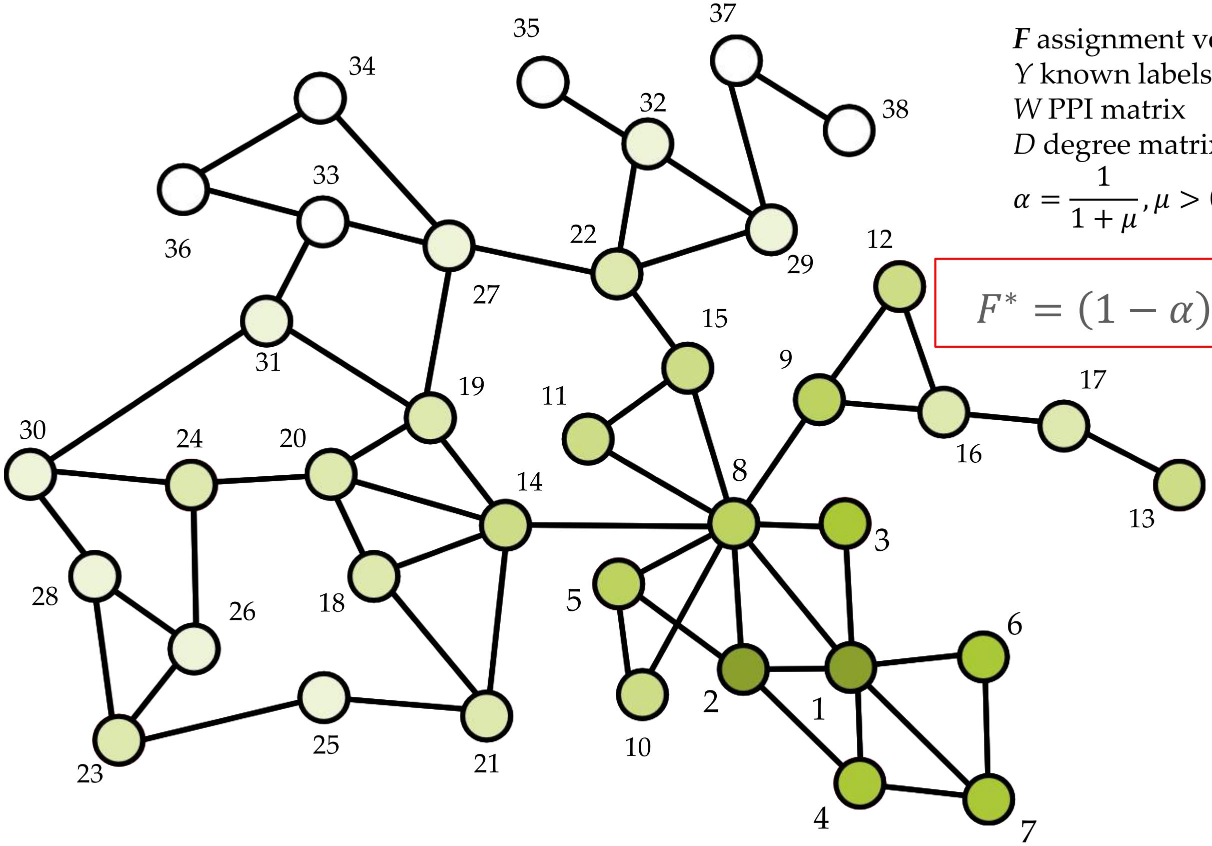
$$Q(F) = \frac{1}{2}\left(\sum_{i,j=1}^{n} W_{ij}\left\|\frac{1}{\sqrt{D_{ii}}}F_i - \frac{1}{\sqrt{D_{jj}}}F_j\right\|^2 + \mu\sum_{i=1}^{n}\|F_i - Y_i\|^2\right)$$

F assignment vector
Y known labels
W PPI matrix
D degree matrix of W

$$\alpha = \frac{1}{1+\mu}, \mu > 0$$

(Zhou et al, NIPS 2004, "Consistency" method)

$$\boxed{F^* = (1-\alpha)\left(I - \alpha D^{-1/2} W D^{-1/2}\right)^{-1} Y}$$



A. Paccanaro, 2019

Testing Setup

Disease categories

Uncharted diseases

Currently there are no known disease genes

Charted diseases

Some disease genes are know

Experiment types

Prospective evaluations

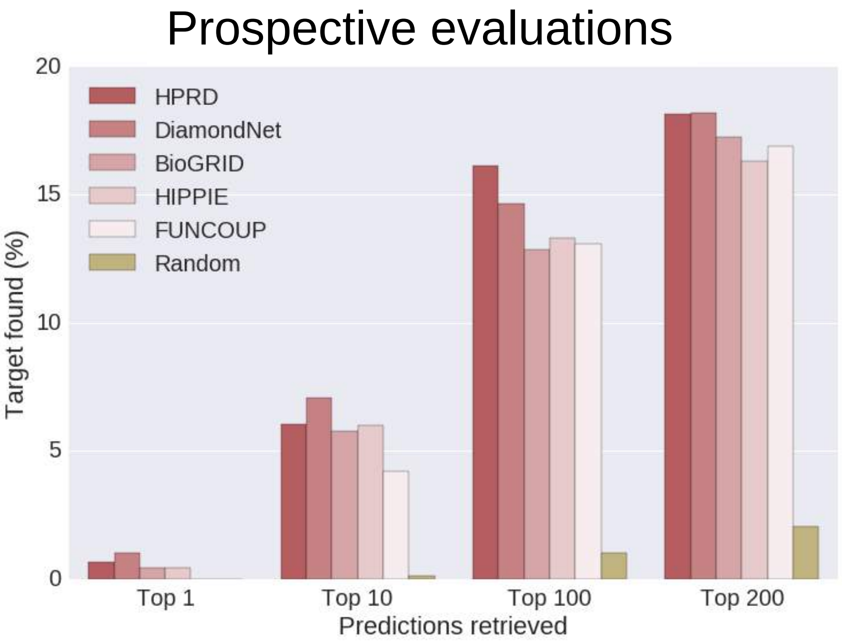
Using information from 2013, predict new disease genes known in 2018

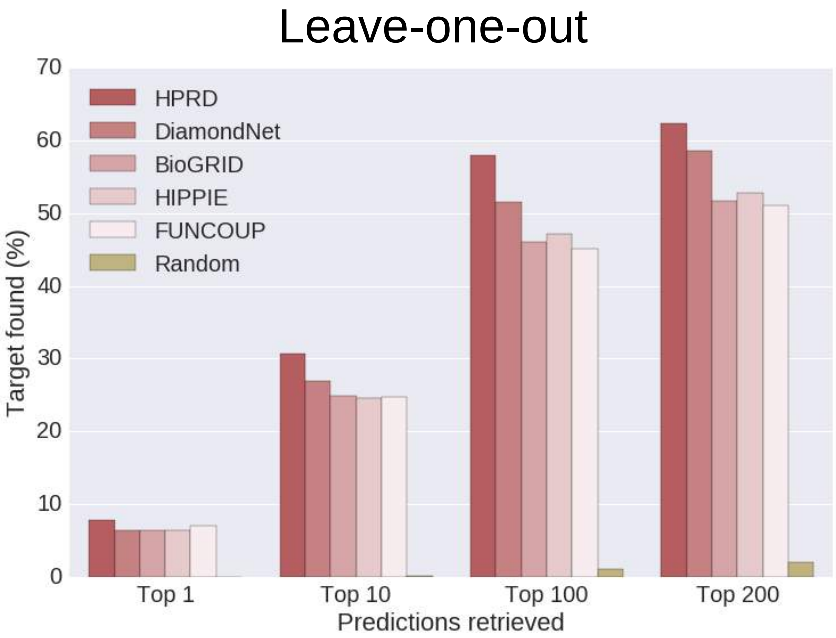
Leave-one-out

Using data from 2018, a single association is removed and is predicted back

A. Paccanaro, 2019

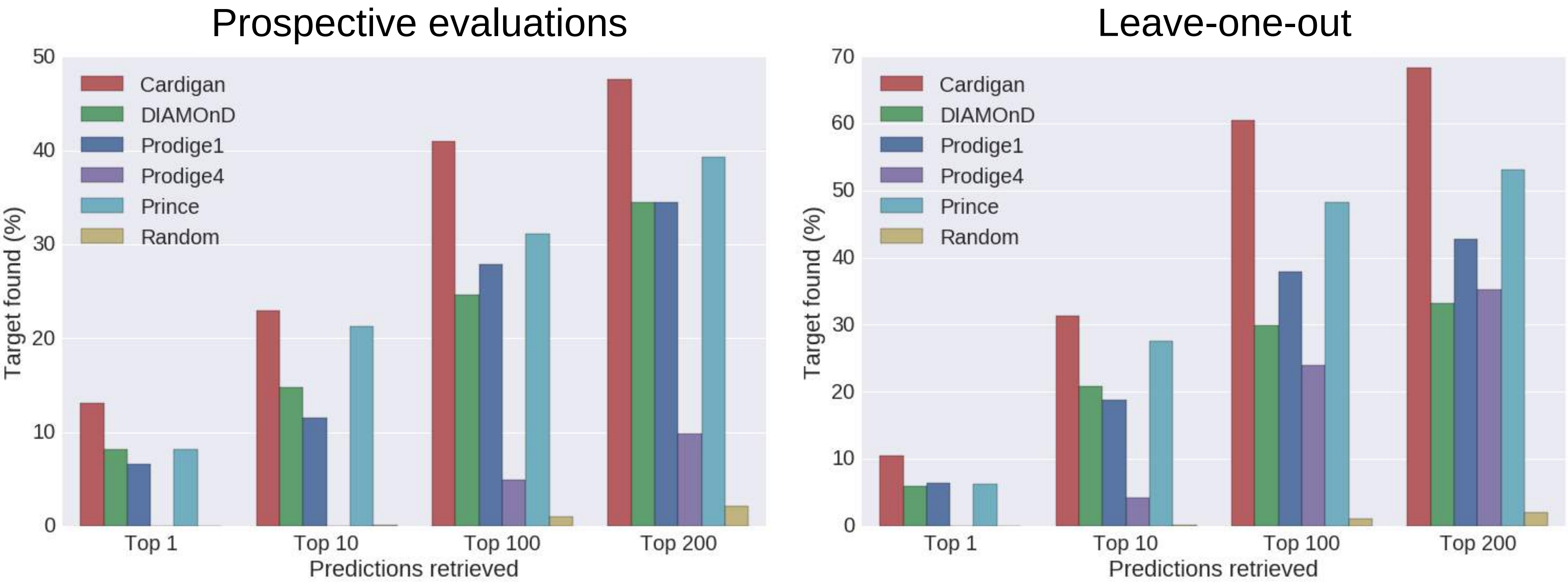
Performance – uncharted diseases





A. Paccanaro, 2019

Performance – charted diseases



Prospective evaluations

Leave-one-out

DIAMOnD -- *Ghiassian, Menche, Barabasi, PLoS Comp Bio 2015*
Prodige1,4 -- *Mordelet, Vert, BMC Bioinformatics, 2011*
Prince -- *Vanunu, Magger, Ruppin, Shlomi, Sharan, PLoS Comp Bio 2010*

Prospective evaluation -- Examples

Disease	2013 Status	Gene	Our Ranking	Paper
Familial Retinal Arteriolar Tortuosity (MIM:180000)	Uncharted	COL4A1	5	*Zenten J. et al. , Graefe's Arch. Clin. Exp Ophthalmology 252, 2014*
Ablepharon-macrostomia syndrome (MIM:200110)	Uncharted	TWIST2	10	*Marchegiani et al., American J. of Human Genetics 97, 2015*
Fetal Akinesia Deformation Sequence (MIM:208150)	Charted	MUSK	1	*Tan-Sindhunata et al. , Eur. J. Human Genetics 23, 2015*
Schimmelpenning-Feuerstein-Mims syndrome (MIM:163200)	Charted	NRAS	1	*Lim et al. , Human molecular genetics 23, 2014*

A. Paccanaro, 2019

Conclusions

- ✓ A distance between disease modules on the interactome which uses exclusively disease phenotype information.

- ✓ How diffusion methods + our disease similarity measure can be used to infer disease genes for uncharted diseases.

- ✓ These methods can provide **explanations**

A. Paccanaro, 2019

References

(from which I took some figures)

- **H. Caniza, A. E. Romero, A. Paccanaro**

  A network medicine approach to quantify distance between hereditary disease modules on the interactome

  Scientific Reports, vol. 5, 17658 (2015)

- **J.J. Cáceres, A. Paccanaro**

  Disease gene prediction for molecularly uncharacterized diseases

  PLoS Computational Biology, vol. 15 (2019)

A. Paccanaro, 2019

Data Science

São Paulo School of Advanced Science
on Learning from Data

USP

A brief intro to Recommender Systems

Alberto Paccanaro

Department of Computer Science

Royal Holloway, University of London

www.paccanarolab.org

What is the goal of a RecSys?

Predicting relevant items to users (e.g. movies)

As in Netflix, to predict the rating value 1,2,3,4, or 5 for each movie.

NETFLIX ORIGINAL

STRANGER THINGS

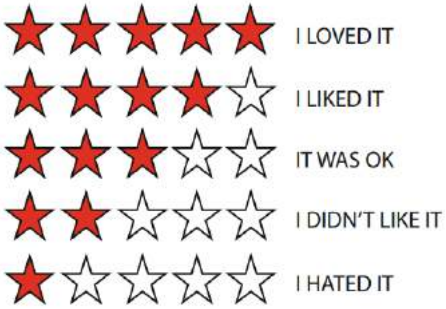
95% Match 2016 1 Season 4K Ultra HD 5.1

When a young boy vanishes, a small town uncovers a mystery involving secret experiments, terrifying supernatural forces and one strange little girl.

Winona Ryder, David Harbour, Matthew Modine
TV Shows, TV Sci-Fi & Fantasy, Teen TV Shows

Popular on Netflix

Recently Watched



A. Paccanaro, 2019

Brief History: The Netflix Prize

- Year: 2006
- Competition for the best collaborative filtering algorithm
- Data: 480,189 users x 17,770 movies with 100,480,507 ratings (~ 1.7% density).
- Prize: US$1,000,000

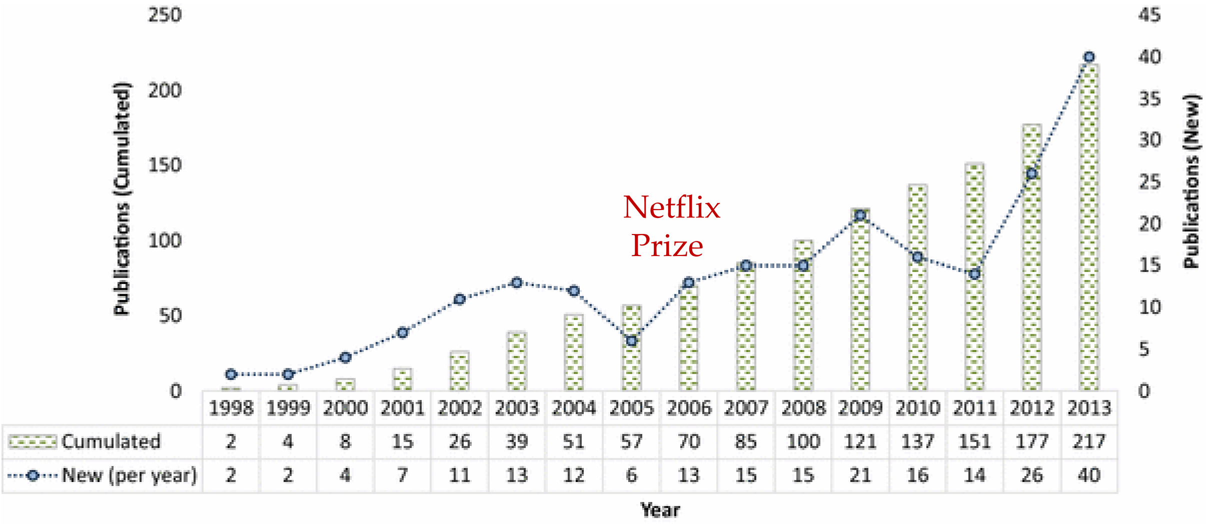
Over 40,000 teams registered
from 186 countries
Growing interest in the field

Robert Bell, Yehuda Koren
Pragmatic Chaos

NETFLIX
2009
DATE 09-21-09
PAY TO THE ORDER OF BellKor's Pragmatic Chaos
$1,000,000 00
AMOUNT ONE MILLION 00/100
FOR The Netflix Prize

A. Paccanaro, 2019

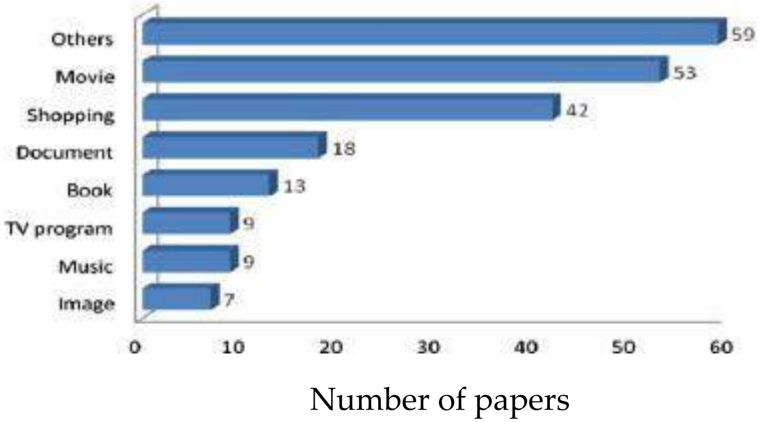
Growing interest in RecSys



Netflix Prize

	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
▨ Cumulated	2	4	8	15	26	39	51	57	70	85	100	121	137	151	177	217
···●··· New (per year)	2	2	4	7	11	13	12	6	13	15	15	21	16	14	26	40

Application fields



From Beel, Joeran, et al. "Research-paper recommender systems: a literature survey." *International Journal on Digital Libraries* 17.4 (2016): 305-338.

A. Paccanaro, 2019

Topics

- Content-based Filtering
- Collaborative Filtering
  - ❑ Neighbourhood-Based CF
  - ❑ Model-Based CF
    - » Latent factor models
    - » Matrix decomposition
    - » Non-negative matrix factorization
    - » Modelling user and item biases
    - » Implicit feedback

A. Paccanaro, 2019

Content-based Filtering

- Assumption/Scenario: we do not have access to other users ratings.

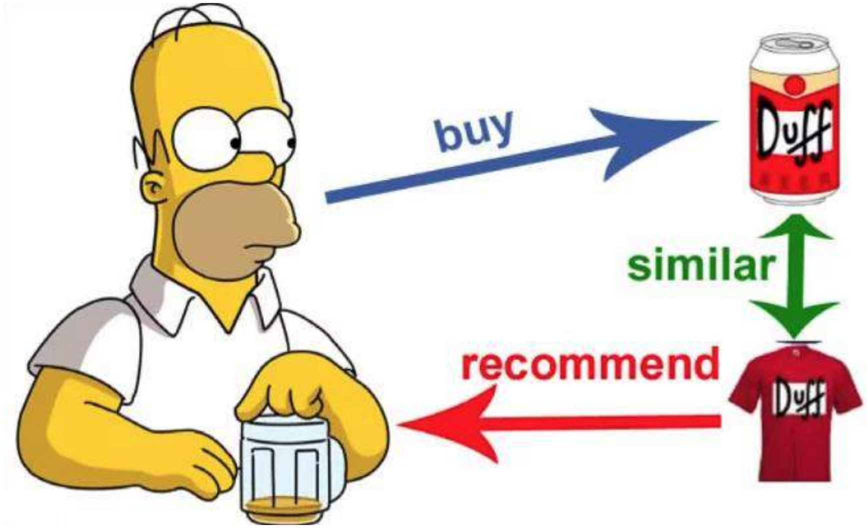
- Profiles for users and movies
  - Movie: genre, actors, box office popularity, plot, etc.
  - Users: demographic information, age, sex, etc.

- Example:
  - John liked Terminator.
  - Terminator has similar genre keywords as Alien and Predator.
  - Recommend Alien and Predators to John.

Aggarwal, Charu C. *Recommender systems*. Cham: Springer International Publishing, 2016.

A. Paccanaro, 2019

Content-based Filtering



https://medium.com/building-ibotta/ibottas-recommender-system-7a4034773bf9

A. Paccanaro, 2019

The rest of this lecture

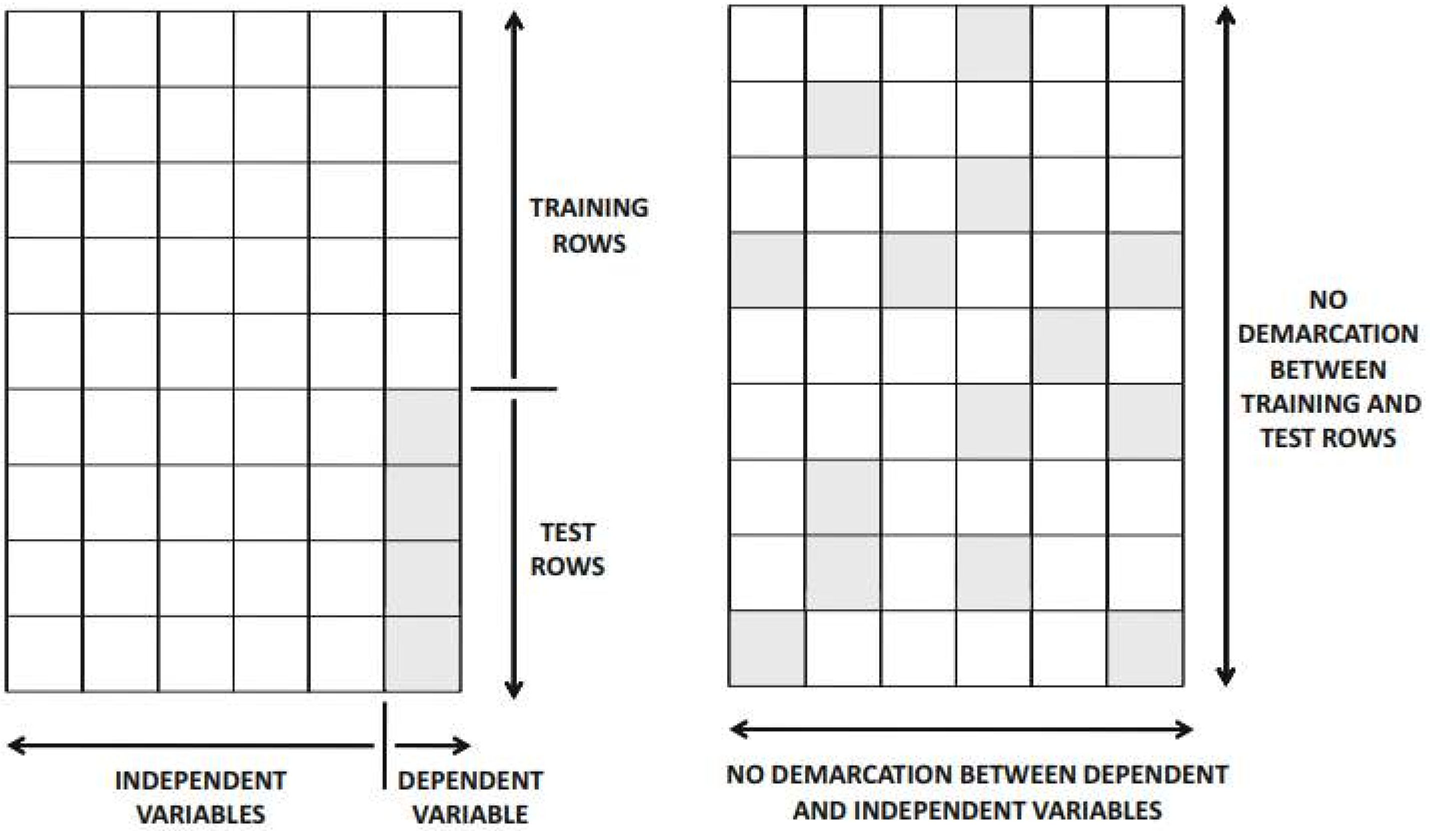
- ~~Content-based Filtering~~
- Collaborative Filtering
  - ❑ Neighbourhood-Based CF
  - ❑ Model-Based CF
    - » Latent factor models
    - » Matrix decomposition
    - » Non-negative matrix factorization
    - » Modelling user and item biases
    - » Implicit feedback

A. Paccanaro, 2019

Collaborative Filtering

Past users behaviour is available – e.g. previous ratings – without requiring the creation of explicit profiles

How Collaborative Filtering is different from classification?


TRAINING ROWS
TEST ROWS
INDEPENDENT VARIABLES
DEPENDENT VARIABLE
NO DEMARCATION BETWEEN TRAINING AND TEST ROWS
NO DEMARCATION BETWEEN DEPENDENT AND INDEPENDENT VARIABLES

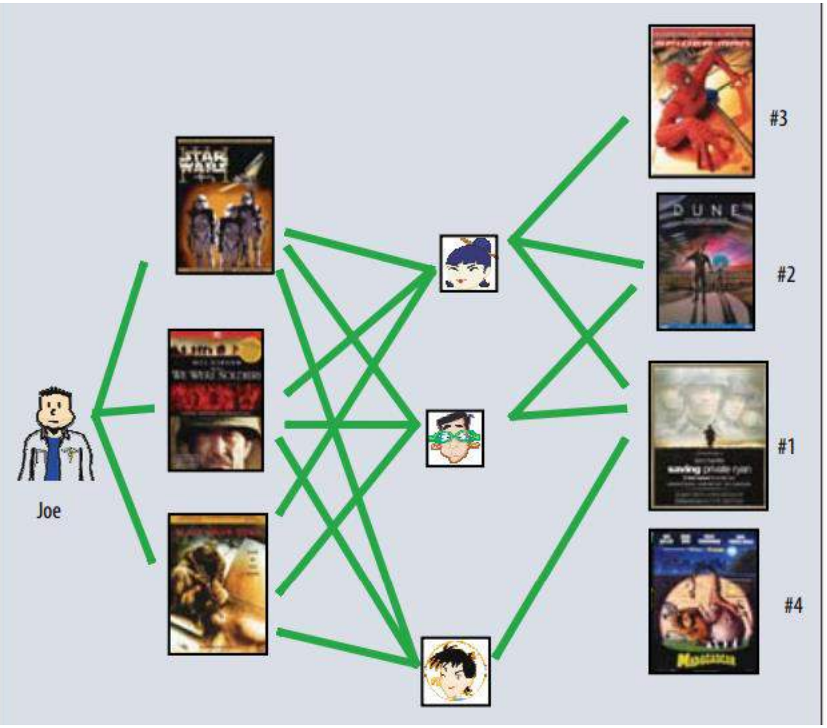
(a) Classification

(b) Collaborative filtering

A. Paccanaro, 2019

Neighbourhood-based models

- User-based: deliver recommendation by finding *similar* users
- Item-based: deliver recommendations by finding *similar* items (movies)



From Koren, Bell,Volinsky, *Computer* (2009): 30-37.

A. Paccanaro, 2019

How do we define similarities?

- Pearson correlation

$$\frac{\sum_{k\in I_u\cap I_v}(r_{uk}-\mu_u)\cdot(r_{vk}-\mu_v)}{\sqrt{\sum_{k\in I_u\cap I_v}(r_{uk}-\mu_u)^2}\cdot\sqrt{\sum_{k\in I_u\cap I_v}(r_{vk}-\mu_v)^2}}$$

- Cosine similarity

$$\frac{\sum_{k\in I_u\cap I_v} r_{uk}\cdot r_{vk}}{\sqrt{\sum_{k\in I_u\cap I_v} r_{uk}^2}\cdot\sqrt{\sum_{k\in I_u\cap I_v} r_{vk}^2}}$$

u, v: two given users.
$R = [r_{uj}]$ matrix of $n \times m$ containing ratings for n users and m movies
I_u, I_v: set of movies indices rated by user u and v, respectively.
μ_u, μ_v: mean rating for user u and v, respectively.
$P_u(j)$: set of k closest users to target user u.

A. Paccanaro, 2019

Strengths and weaknesses

- Strengths:
  - Simple and intuitive
  - Interpretable

- Weakness:
  - Impractical in large-scale settings
  - Computationally expensive: need to compute all pairwise similarities between users or items

A. Paccanaro, 2019

The rest of this lecture

- ~~Content-based Filtering~~
- Collaborative Filtering
  - ~~❑ Neighbourhood-Based CF~~
  - ❑ Model-Based CF
    - » Latent factor models
    - » Matrix decomposition
    - » Non-negative matrix factorization
    - » Modelling user and item biases
    - » Implicit feedback

A. Paccanaro, 2019

Latent Factor Models

- Goal: to find "hidden" factors in the user-movie rating matrix that explains user preferences.

- These factors can be thought of as modelling movie genres and user preferences, e.g. thriller, sci-fi, etc.

Latent Factor (matrix decomposition) models – the idea

Movies (q)

Users (p)

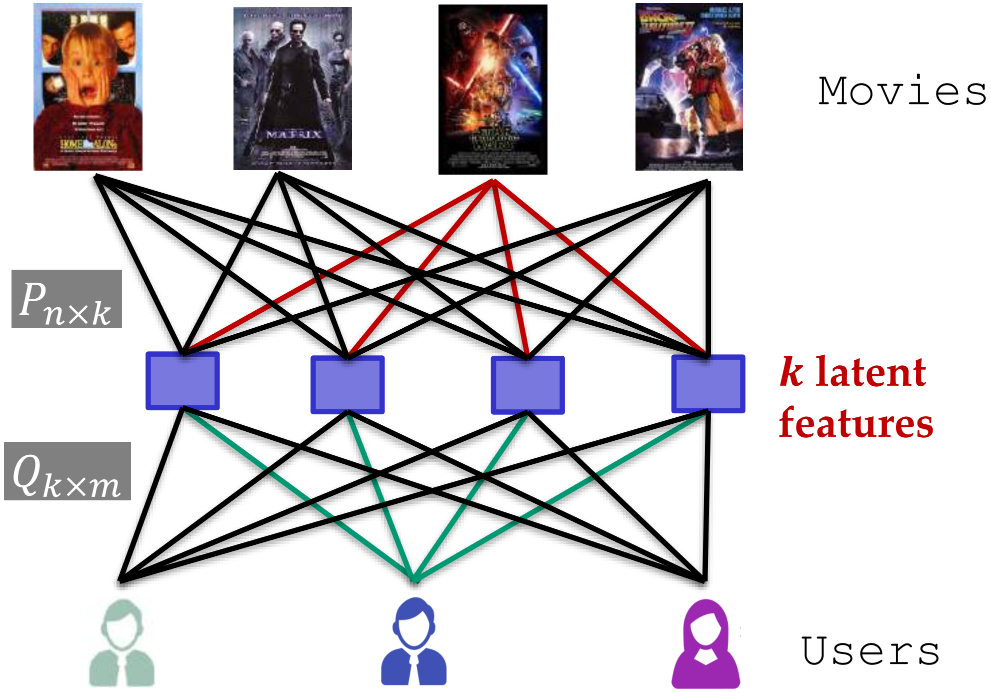
1	0	0	0	2	0	3	0	4	0
4	0	0	3	0	4	0	0	0	1
0	0	4	0	0	0	1	0	3	0
5	0	0	0	0	5	0	1	0	0
0	0	4	0	3	0	2	0	0	0

Y

Matrix decomposition models are useful for very sparse datasets with potential **latent features**

$$Y_{i,j} \approx \boldsymbol{p}_i^T \cdot \boldsymbol{q}_j$$

$$Y_{n\times m} \approx P_{n\times k} \cdot Q_{k\times m}$$



A. Paccanaro, 2019

Matrix decomposition

- User u: low-dimensional feature vector $q_u \in \mathbb{R}^k$.
- Movie j: low-dimensional feature vector $p_j \in \mathbb{R}^k$.
- **Rating prediction:** $\boldsymbol{\hat{r}_{uj} = q_u \cdot p_j}$

These are learned by minimising:

$$\min_{q^*,p^*} \sum_{(u,i)\in\kappa} (r_{ui} - q_i^T p_u)^2 + \lambda(\| q_i \|^2 + \| p_u \|^2)$$

It can be solved by stochastic gradient descent:

$$q_i \leftarrow q_i + \gamma \cdot (e_{ui} \cdot p_u - \lambda \cdot q_i)$$
$$p_u \leftarrow p_u + \gamma \cdot (e_{ui} \cdot q_i - \lambda \cdot p_u)$$

$$e_{ui} \overset{def}{=} r_{ui} - q_i^T p_u.$$

A. Paccanaro, 2019

Matrix decomposition

- Matrix form:
  - $R \in \mathbb{R}^{n \times m}$: ratings of n users and m movies
  - $P \in \mathbb{R}^{n \times k}$: users latent factors (each row is a user).
  - $Q \in \mathbb{R}^{k \times m}$: movies latent factors (each column is a movie).
  - Ω: set of observed entries in R.

$$\boxed{\text{Model} \qquad \hat{R} \simeq PQ}$$

Learned by minimising the cost function:

$$\boxed{\min_{P,Q} \zeta(P,Q) = \frac{1}{2} \|\Omega \circ (R - PQ)\| + \frac{\lambda}{2} (\|P\| + \|Q\|)}$$

Fits model
to observed entries

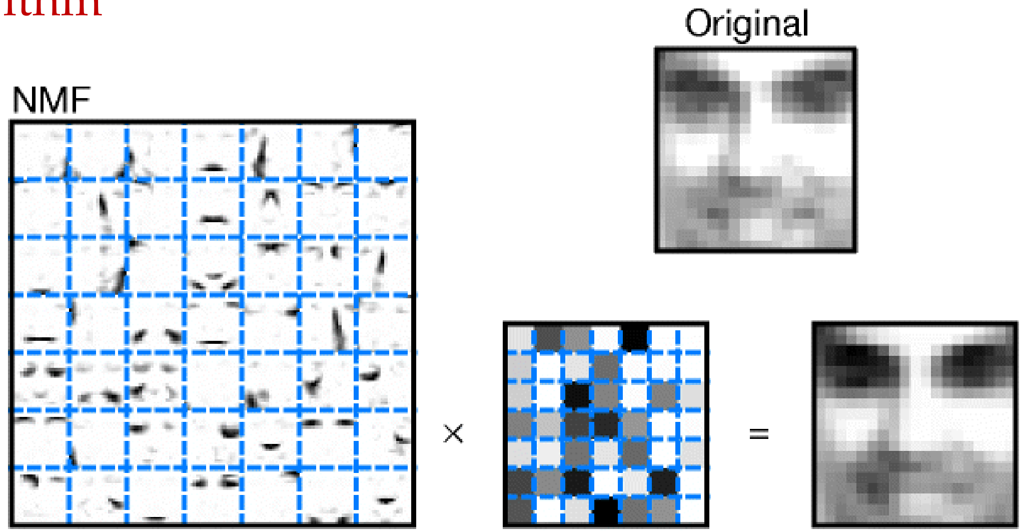
Regularization to
prevent overfitting

A. Paccanaro, 2019

Non-negative matrix decomposition (NMF)

- Additional non-negative constraint: $P, Q \geq 0$.

- Why NMF is interesting?
  - Model interpretability
  - Efficient Multiplicative algorithm



A. Paccanaro, 2019

From Lee, Daniel D., and H. Sebastian Seung. "Learning the parts of objects by non-negative matrix factorization." *Nature* 401.6755 (1999): 788.

Modelling users and item biases

- There are users who tend to rate always high (above mean rating) or low (below mean rating).

$$\hat{r}_{ui} = \mu + b_i + b_u + q_i^T p_u$$

μ: mean rating of all users.
b_i: bias of item i
b_u: bias of user u

We need to learn also b_i and b_u!

Learned by minimising the cost:

$$\min_{p\cdot,q\cdot,b\cdot} \sum_{(u,i)\in\kappa} (r_{ui} - \mu - b_u - b_i - p_u^T q_i)^2 + \lambda$$

$$(\| p_u \|^2 + \| q_i \|^2 + b_u^2 + b_i^2)$$

A. Paccanaro, 2019

The rest of this lecture

- ~~Content-based Filtering~~
- Collaborative Filtering
  - ~~❑ Neighbourhood-Based CF~~
  - ~~❑ Model-Based CF~~
    - ~~» Latent factor models~~
    - ~~» Matrix decomposition~~
    - ~~» Non-negative matrix factorization~~
    - ~~» Modelling user and item biases~~
    - » Implicit feedback

A. Paccanaro, 2019

Implicit Feedback

- Implicit feedback: additional information about users, e.g. which movies were clicked (plots read).
- These additional information can be integrated into the model.

A. Paccanaro, 2019

References
(from which I took some figures)

- Aggarwal, Charu C. *Recommender systems*. Cham: Springer International Publishing, 2016.
- Koren, Yehuda, Robert Bell, and Chris Volinsky. "Matrix factorization techniques for recommender systems." *Computer* (2009): 30-37.
- Herlocker, Jonathan L., et al. "Evaluating collaborative filtering recommender systems." *ACM Transactions on Information Systems (TOIS)* 22.1 (2004): 5-53.
- Melville, Prem, and Vikas Sindhwani. "Recommender systems." *Encyclopedia of Machine Learning and Data Mining*(2017): 1056-1066.
- Ricci, Francesco, Lior Rokach, and Bracha Shapira. "Recommender systems: introduction and challenges." *Recommender systems handbook*. Springer, Boston, MA, 2015. 1-34.

A. Paccanaro, 2019

Data Science

São Paulo School of Advanced Science on Learning from Data

USP

Material in the following slides taken from :
Galeano, Paccanaro 2019
<u>BioRxiv</u> 594465, doi: 10.1101/594465

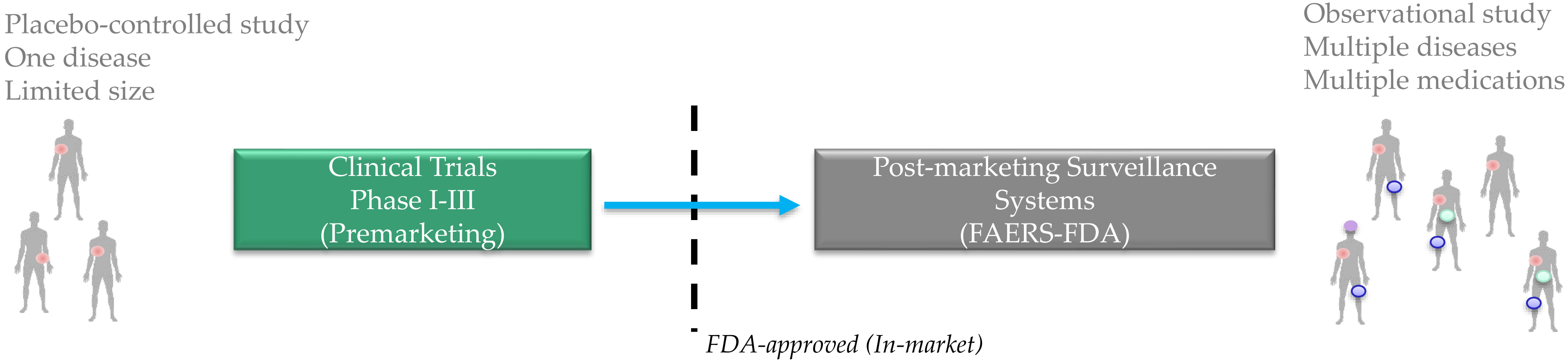
A collaborative model for predicting the frequency of drug side effects

Alberto Paccanaro
Department of Computer Science
Royal Holloway, University of London
www.paccanarolab.org

Drugs side effects

A drug-side effect association in humans can be:

Very rare:	< 0.01%
Rare:	< 0.1%
Infrequent:	< 1%
Frequent:	< 10%
Very frequent:	> 10%



Placebo-controlled study
One disease
Limited size

Clinical Trials
Phase I-III
(Premarketing)

Post-marketing Surveillance
Systems
(FAERS-FDA)

Observational study
Multiple diseases
Multiple medications

FDA-approved (In-market)

A. Paccanaro, 2019

Question

Can we predict the frequency of drug side effects ?

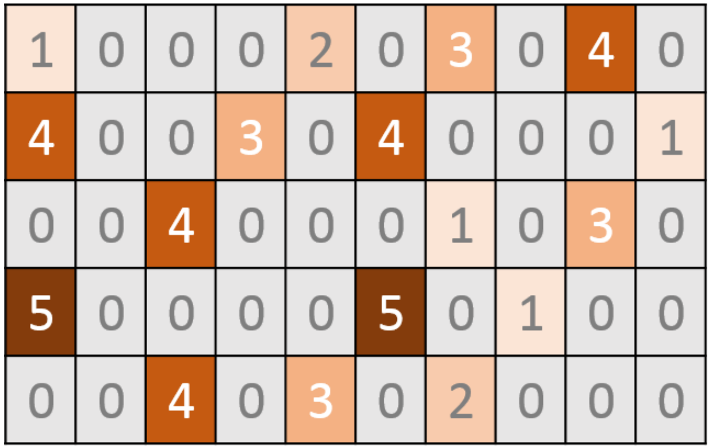
Few methods exists which are aimed at predicting the presence/absence of side effects. These exploit molecular or cellular features.

A. Paccanaro, 2019

The data

996 side effect terms

760 drugs



1	0	0	0	2	0	3	0	4	0
4	0	0	3	0	4	0	0	0	1
0	0	4	0	0	0	1	0	3	0
5	0	0	0	0	5	0	1	0	0
0	0	4	0	3	0	2	0	0	0

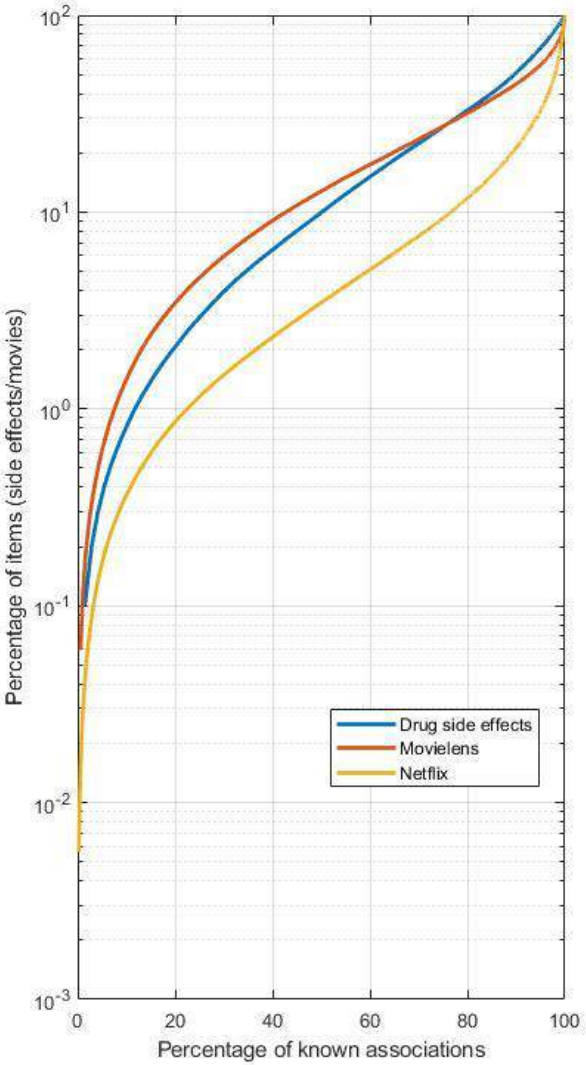
density ~ 5% (sparse)

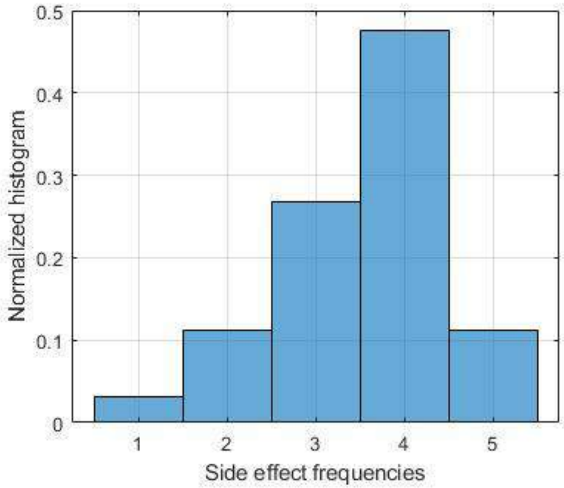
Very rare = 1
Rare = 2
Infrequent = 3
Frequent = 4
Very Frequent = 5

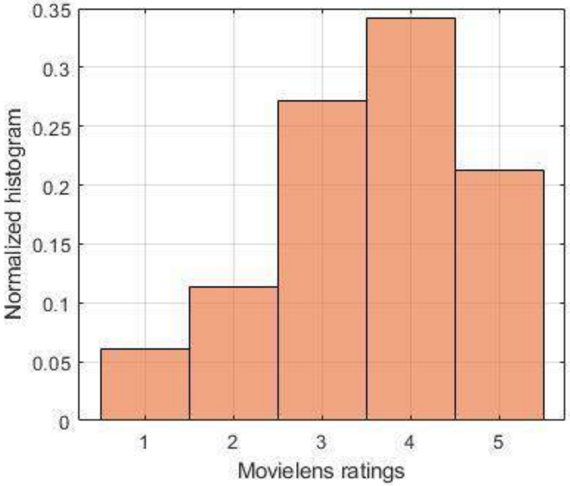
A. Paccanaro, 2019

The Side Effect Resource (SIDER) 4.1 [Khun et al., 2015]

Let’s look at the data…







A. Paccanaro, 2019

How do we predict (recommend) movies?

Movies (q)

Users (p)

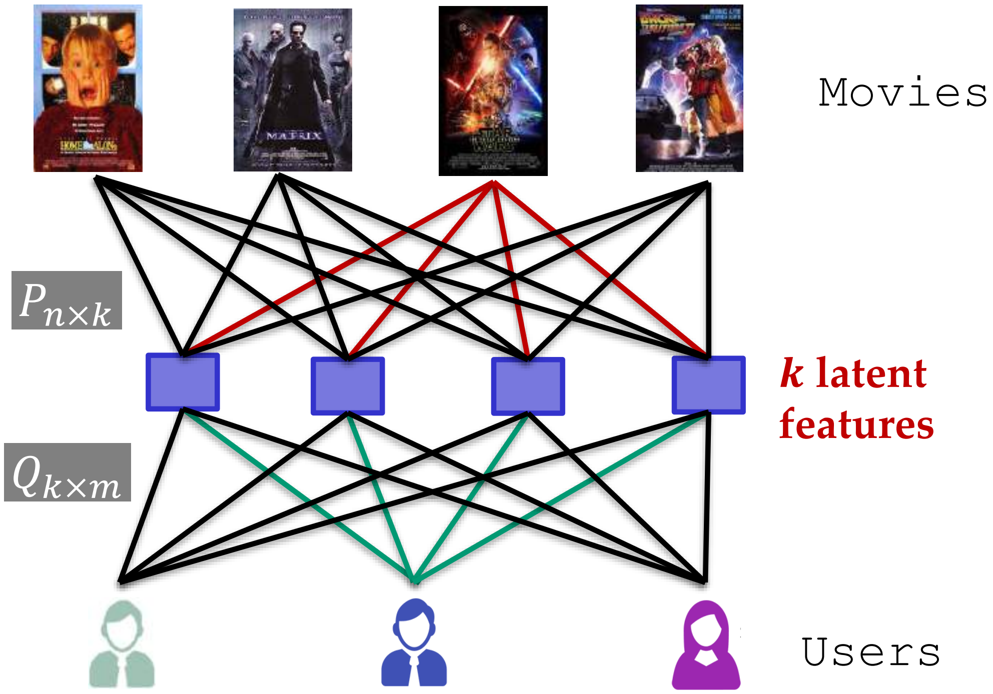
1	0	0	0	2	0	3	0	4	0
4	0	0	3	0	4	0	0	0	1
0	0	4	0	0	0	1	0	3	0
5	0	0	0	0	5	0	1	0	0
0	0	4	0	3	0	2	0	0	0

Y

Matrix decomposition models are useful for very sparse datasets with potential **latent features**

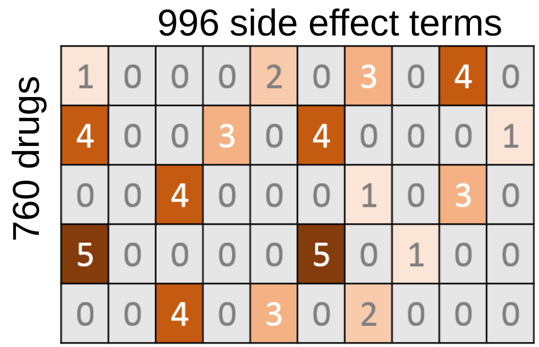
$$Y_{i,j} \approx \boldsymbol{p}_i^T \cdot \boldsymbol{q}_j$$

$$Y_{n\times m} \approx P_{n\times k} \cdot Q_{k\times m}$$



A. Paccanaro, 2019

Our idea: recommending side effects to drugs



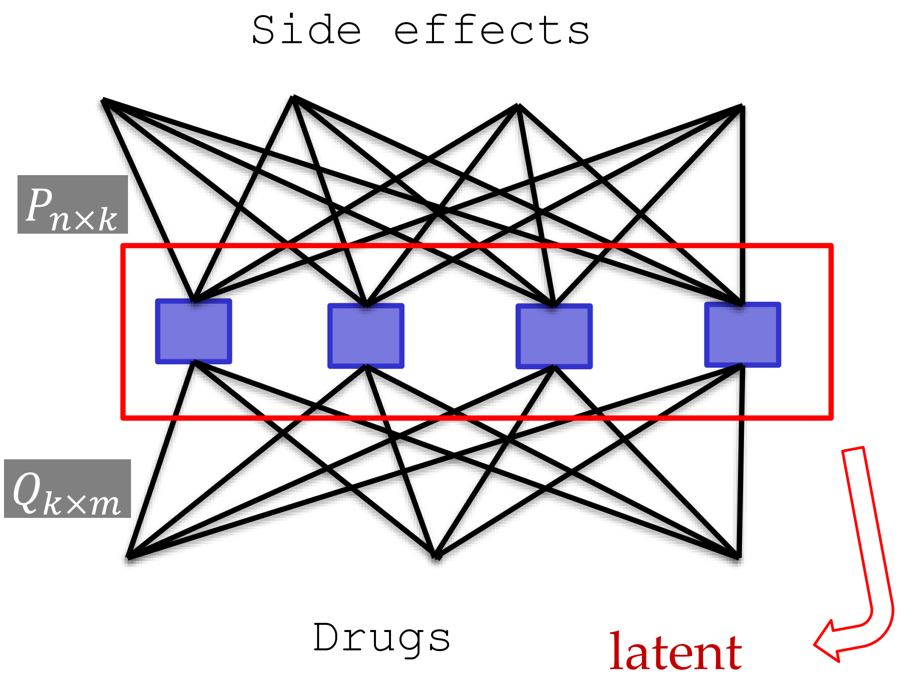
996 side effect terms

760 drugs

1	0	0	0	2	0	3	0	4	0
4	0	0	3	0	4	0	0	0	1
0	0	4	0	0	0	1	0	3	0
5	0	0	0	0	5	0	1	0	0
0	0	4	0	3	0	2	0	0	0

$Y_{n \times m}$

```
Very rare = 1
Rare = 2
Infrequent = 3
Frequent = 4
Very Frequent = 5
```



latent representations *(drug signatures)*

A. Paccanaro, 2019

$$Y_{i,j} \approx \boldsymbol{p}_i^T \cdot \boldsymbol{q}_j$$

$$Y_{n \times m} \approx P_{n \times k} \cdot Q_{k \times m}$$

Learning the latent representations

$$\min_{P,Q} \; J(P,Q) = \frac{1}{2} \| Y - PQ \|_F^2 + \frac{\lambda}{2} (\| P \|_F^2 + \| Q \|_F^2)$$

Low-rank representation of the data

Regularization to prevent overfitting

subject to: $P_{i,j} \geq 0, Q_{i,j} \geq 0$

in order to increase **interpretability**

We learn this with a multiplicative rule (similar to NMF) or with Conjugate Gradient Descent + projections

… it does not work ☹

A. Paccanaro, 2019

Our new cost function

$$\min_{W,H\geq 0} J(W,H) = \frac{1}{2}\sum_{Y_{i,j}\in\{1,2,3,4,5\}} (Y_{i,j} - (WH)_{i,j})^2 + \frac{\alpha}{2}\sum_{Y_{i,j}=0} ((WH)_{i,j})^2$$

Fits clinical trials frequency data

Fits unobserved associations with confidence α_{null}

$Y_{n\times m}$ of n drugs and m side effects
$W_{n\times k}$: drug signatures
$H_{k\times m}$: side effect signatures
$0 \leq \alpha \leq 1$

A. Paccanaro, 2019

We are confident on clinical trials data (values 1-5) but only α-confident on the unobserved associations (0s)

Our model uses the large amount of zeros as a regularization

- Small α allows the weights in W and H to grow
- Large α keeps the weights in W and H small and induces sparsity.

Multiplicative Learning algorithm

Our cost function *converges to a local optimum* using the update rules (satisfy the Karush-Kuhn-Tucker conditions):

$$W \leftarrow W \circ \frac{P_{\Omega}(\mathrm{Y})H^{T}}{\left(P_{\Omega}(\mathrm{WH}) + \alpha \;\; P_{\Omega}^{\neg}(\mathrm{WH})\right)H^{T}}$$

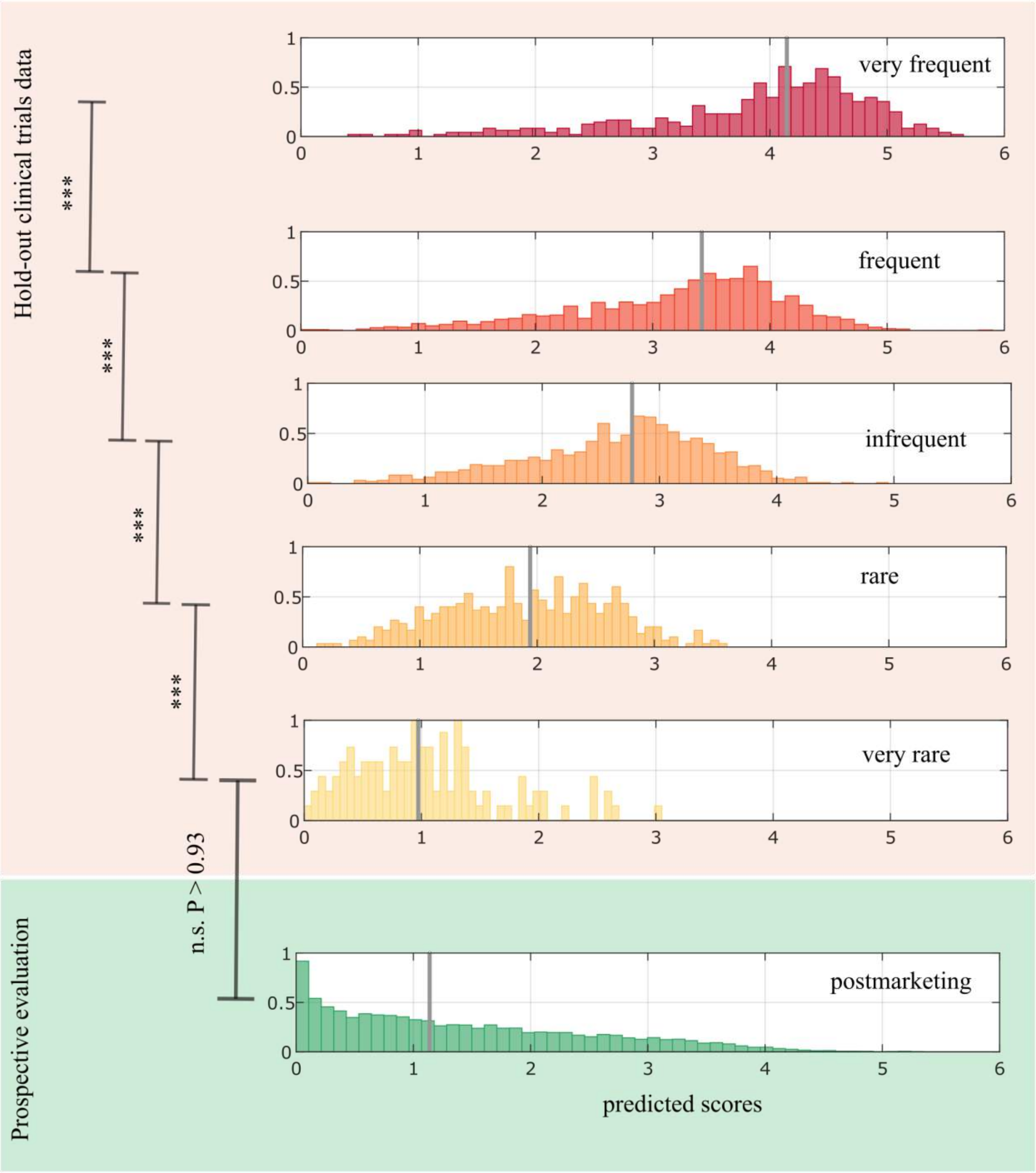
$$H \leftarrow H \circ \frac{W^{T}P_{\Omega}(\mathrm{Y})}{W^{T}\left(P_{\Omega}(\mathrm{WH}) + \alpha \;\; P_{\Omega}^{\neg}(\mathrm{WH})\right)}$$

P_{Ω}: selection function for entries {1,2,3,4,5}
$P_{\Omega}^{\neg}$: selection function for entries {0}
∘ is the Hadamard product

A. Paccanaro, 2019

Multiplicative learning rule – no learning rate, no projection function

Inspired by non-negative matrix factorization (NMF) [Lee, Seung, Nature, 1999]



Prediction on Test Set

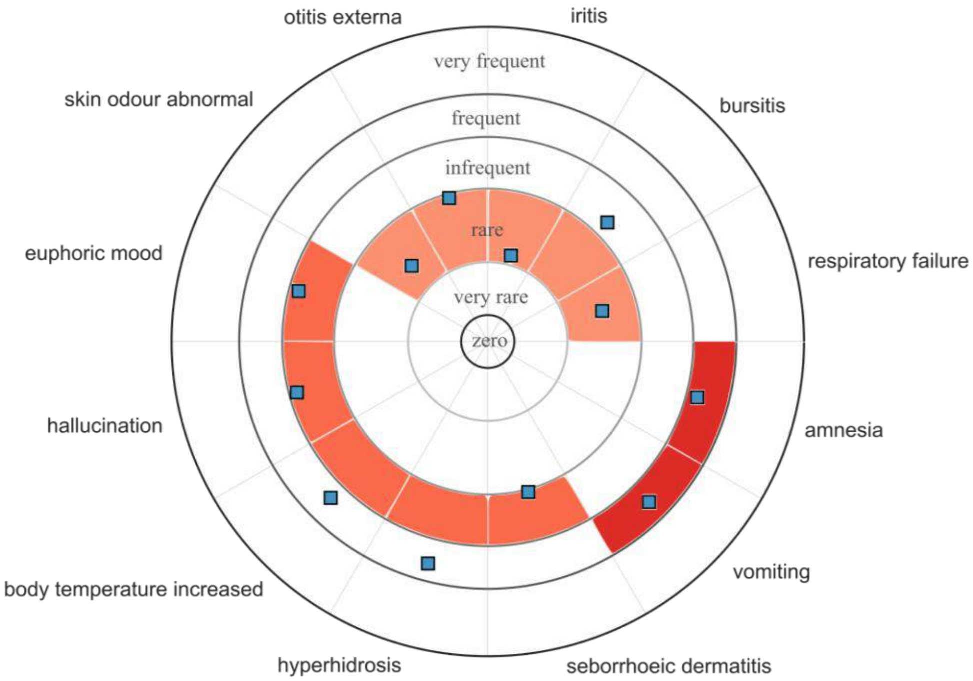
Higher predicted values correspond to higher side effect frequencies

No significant differences between the predicted scores for the **very rare** side effects and the **post-marketing** side effects

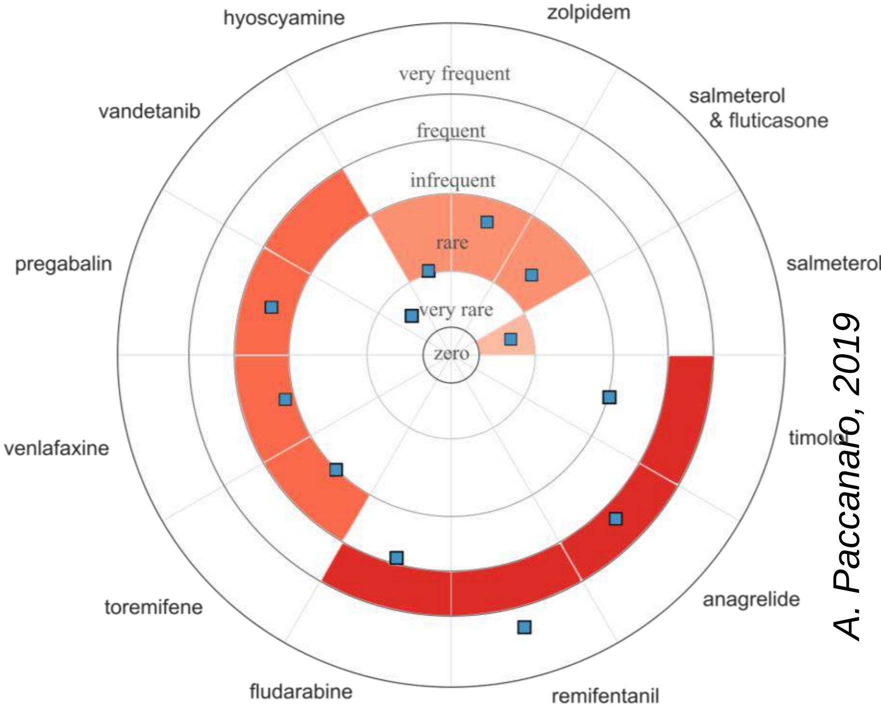
A. Paccanaro, 2019

Examples

Gabapentin
(anticonvulsant drug)



Arrhythmia
(cardiovascular side effect)

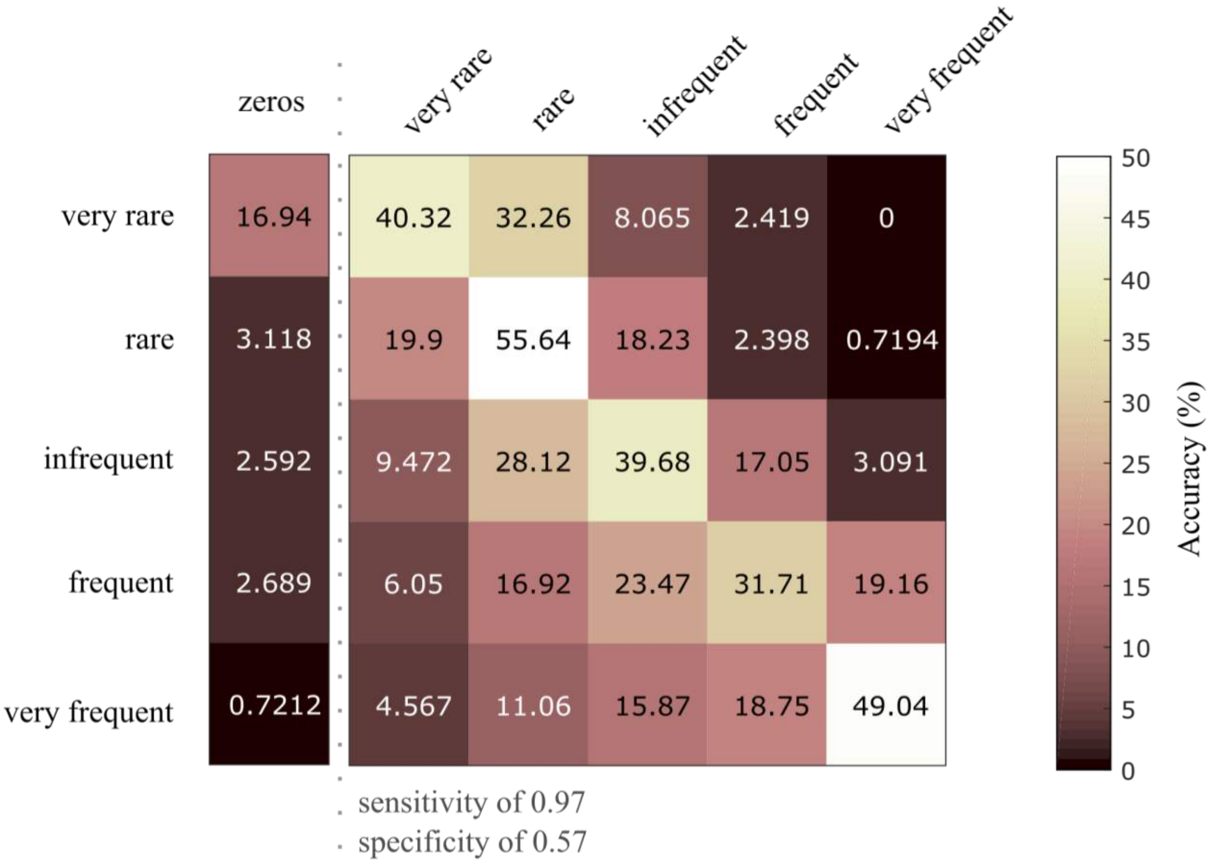


A. Paccanaro, 2019

Percentage of accuracy at predicting the frequency class of drug side effects

Predicted Class

True Class



	zeros	very rare	rare	infrequent	frequent	very frequent
very rare	16.94	40.32	32.26	8.065	2.419	0
rare	3.118	19.9	55.64	18.23	2.398	0.7194
infrequent	2.592	9.472	28.12	39.68	17.05	3.091
frequent	2.689	6.05	16.92	23.47	31.71	19.16
very frequent	0.7212	4.567	11.06	15.87	18.75	49.04

sensitivity of 0.97

specificity of 0.57

A. Paccanaro, 2019

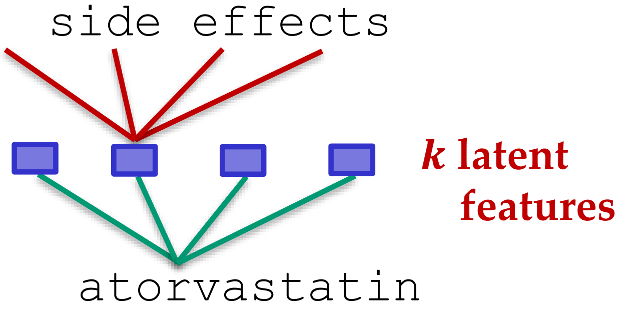
Very rare:	<0.01%
Rare:	<0.1%
Infrequent:	<1%
Frequent:	<10%
Very frequent:	>10%

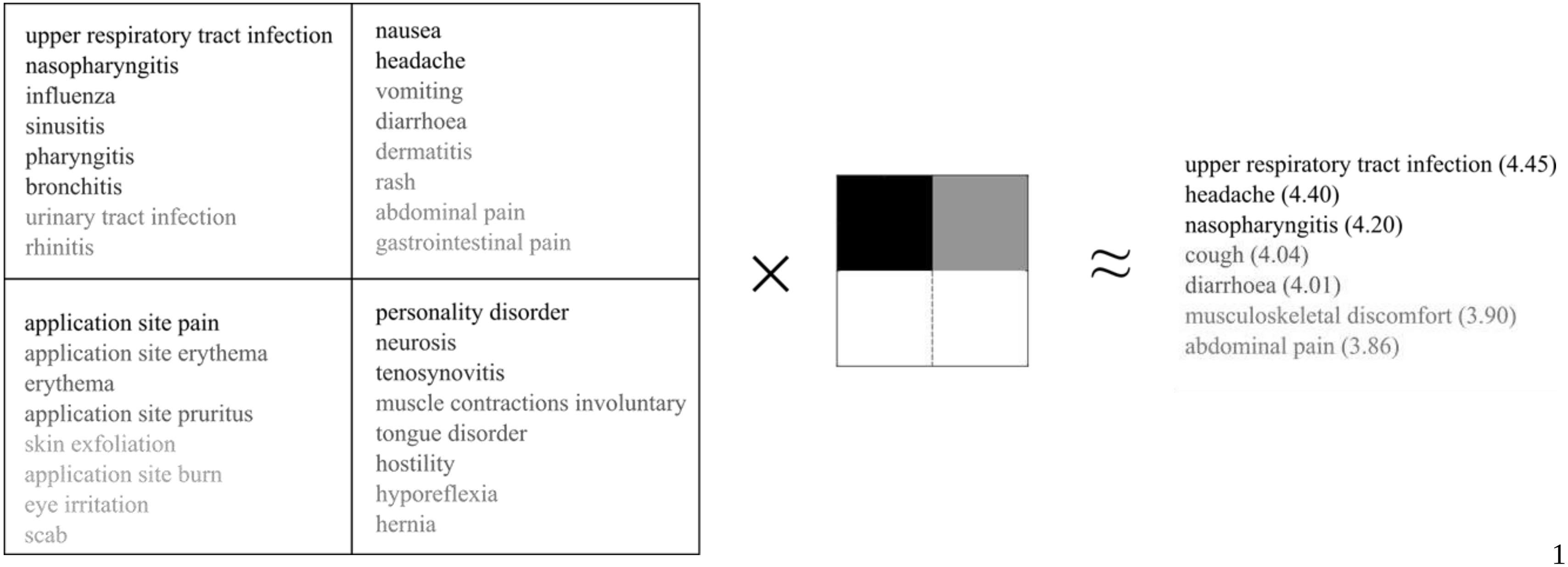
Question: can we “explain” how the prediction works ?

A. Paccanaro, 2019

Predictions can be *explained* in terms of the latent features

Example: *Atorvastatin is known to cause frequent respiratory and thoracic-related side effects*





upper respiratory tract infection nasopharyngitis influenza sinusitis pharyngitis bronchitis urinary tract infection rhinitis	nausea headache vomiting diarrhoea dermatitis rash abdominal pain gastrointestinal pain
application site pain application site erythema erythema application site pruritus skin exfoliation application site burn eye irritation scab	personality disorder neurosis tenosynovitis muscle contractions involuntary tongue disorder hostility hyporeflexia hernia

upper respiratory tract infection (4.45)
headache (4.40)
nasopharyngitis (4.20)
cough (4.04)
diarrhoea (4.01)
musculoskeletal discomfort (3.90)
abdominal pain (3.86)

A. Paccanaro, 2019

Question: do the latent representations tell us something about the *biology* of the problem?

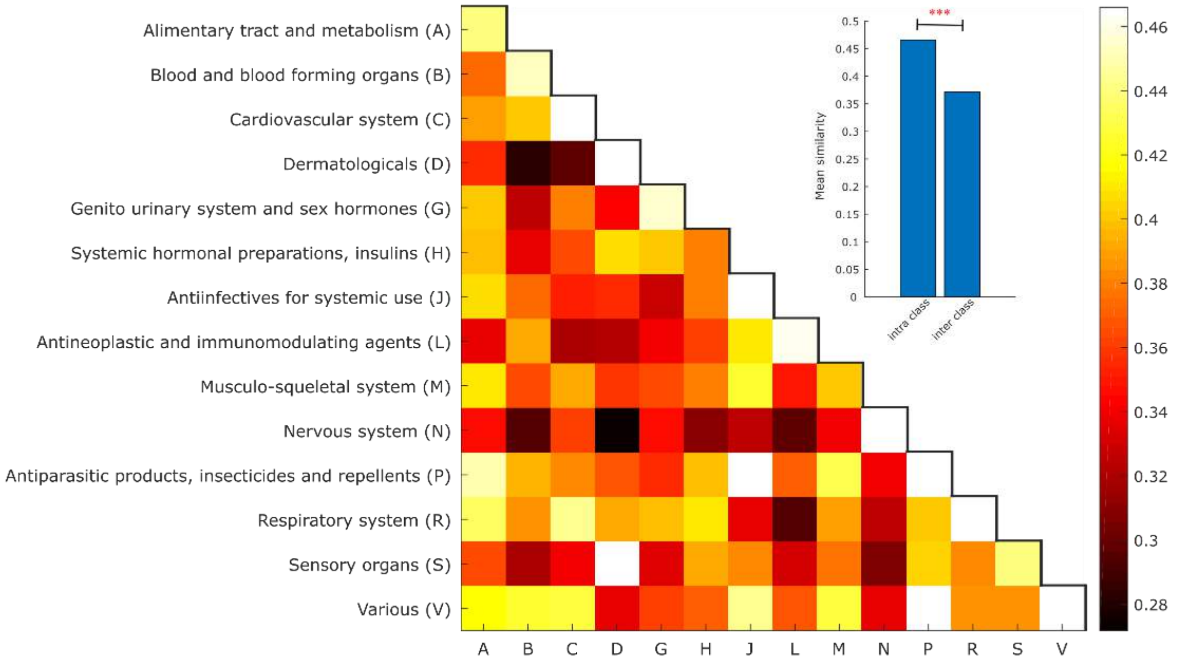
A. Paccanaro, 2019

Drug signature are related to clinical activity of the drug

Hierarchical categorization of drugs according to ATC (from WHO):

1. Anatomical
2. Therapeutic
3. Pharmacological
4. Chemical

Anatomical class



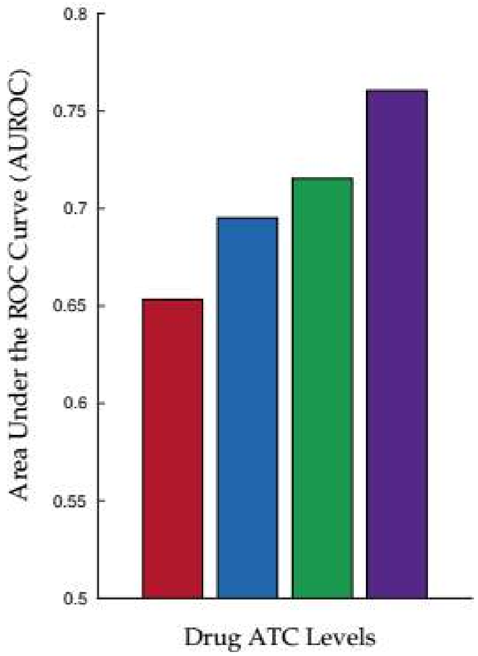
A. Paccanaro, 2019

Drug signature similarity predicts drug clinical activity

🟥 Anatomical class 🟦 Therapeutic subclass 🟩 Pharmacological subclass 🟪 Chemical subclass

Hierarchical categorization of drugs according to ATC (from WHO):

1. Anatomical
2. Therapeutic
3. Pharmacological
4. Chemical



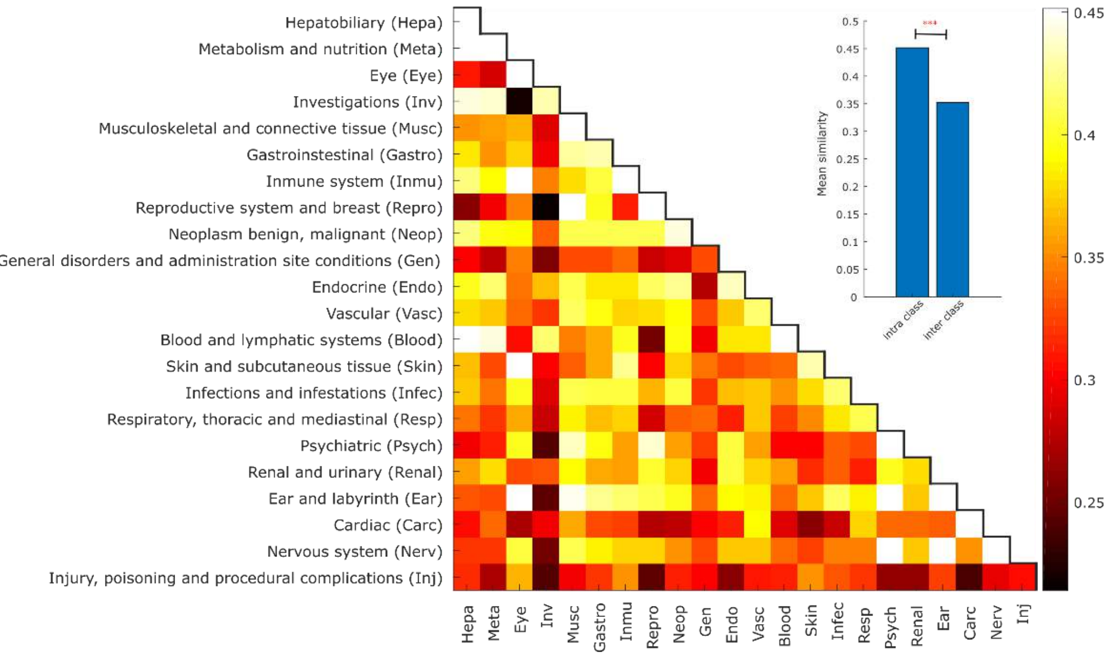
Predicting if ***2 drugs share the same category*** *using the drug signature similarity.*

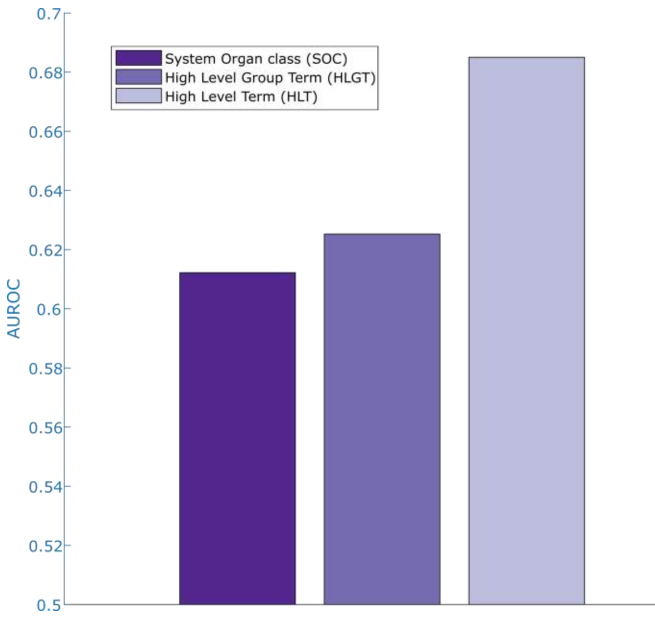
A. Paccanaro, 2019

Side-effect signatures are related to phenotypes

Medical Dictionary for Regulatory Activities (MedDRA) classification of side effects

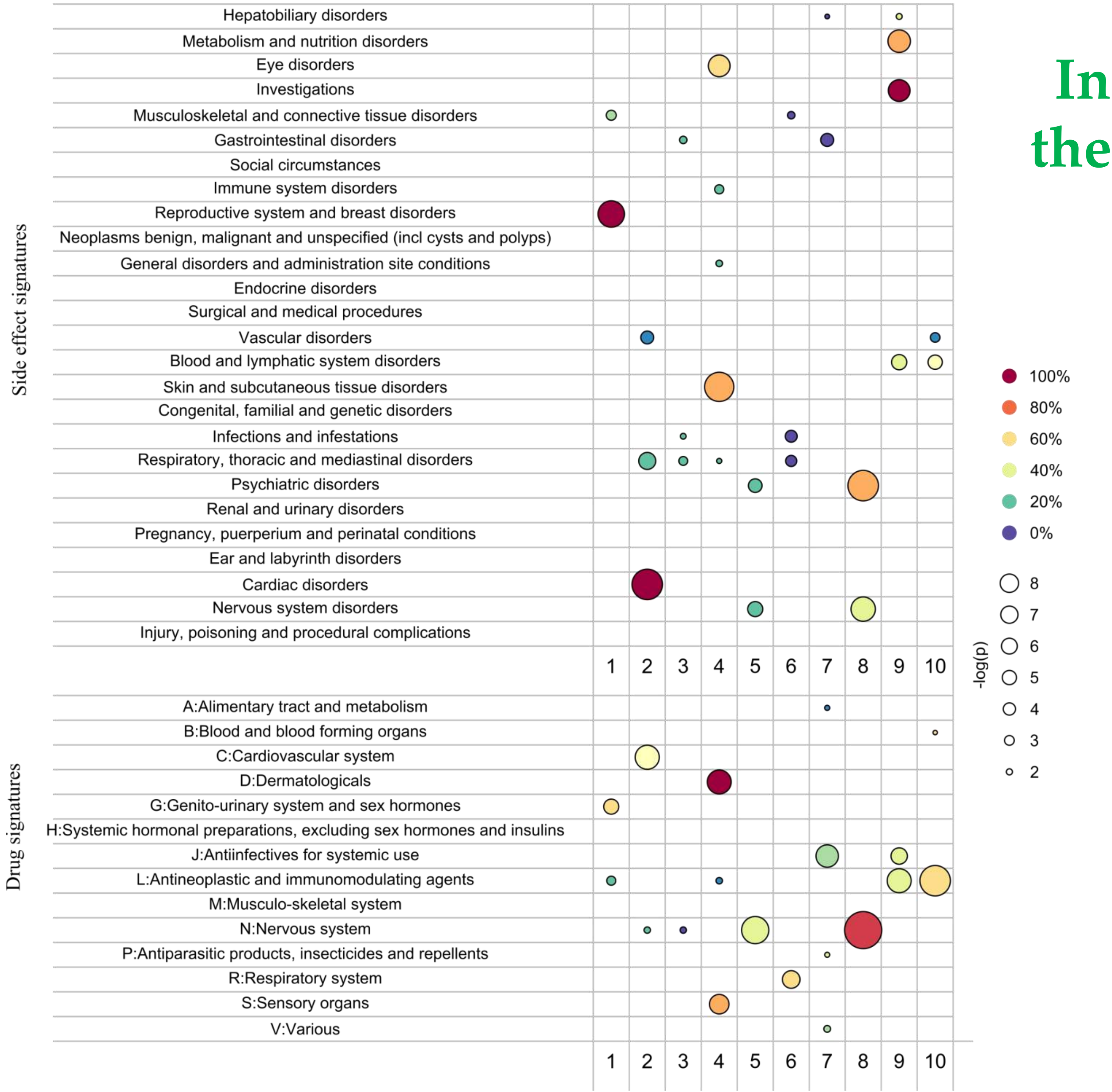
1. System Organ class (anatomy and physiology)
2. High level group term
3. High level term





A. Paccanaro, 2019

Interpreting the signatures

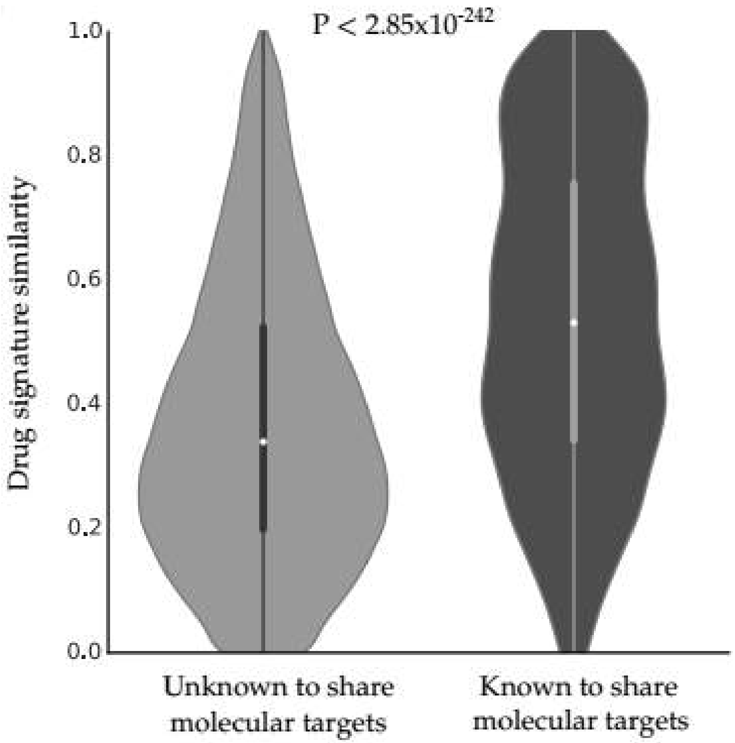


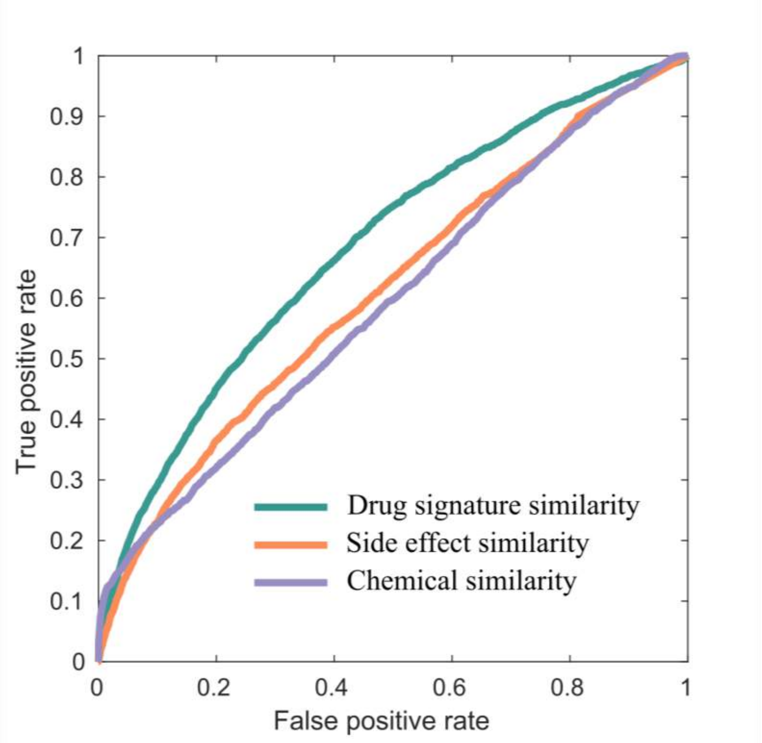
A. Paccanaro, 2019

Question: can we exploit the latent representations for predictions in pharmacology?

A. Paccanaro, 2019

Drug latent representations predict shared targets





A. Paccanaro, 2019

There is a significant difference in the cosine similarity between drug ***signatures for pairs that share targets***

Prediction of whether 2 drugs share molecular targets using similarity between drug signatures

Conclusions

- ✓ A method for predicting the frequency of side-effects in the population.
- ✓ It tells us something about the biology of the problem
- ✓ It can be used for directing clinical trials.
- ✓ It can provide **explanations**

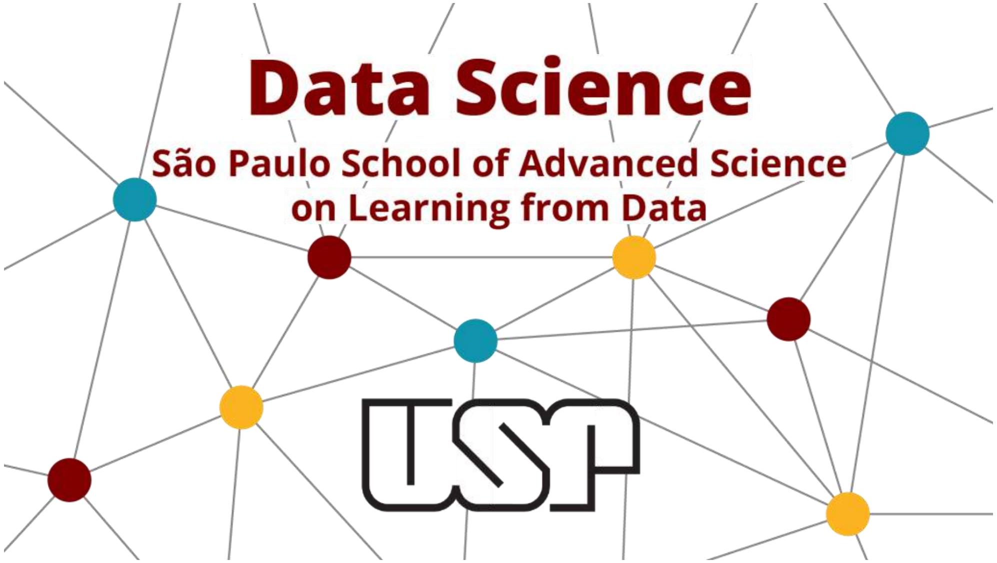
A. Paccanaro, 2019

Reading Material

D. Galeano, A. Paccanaro (2019)

BioRxiv 594465, doi: 10.1101/594465

A. Paccanaro, 2019



Clustering

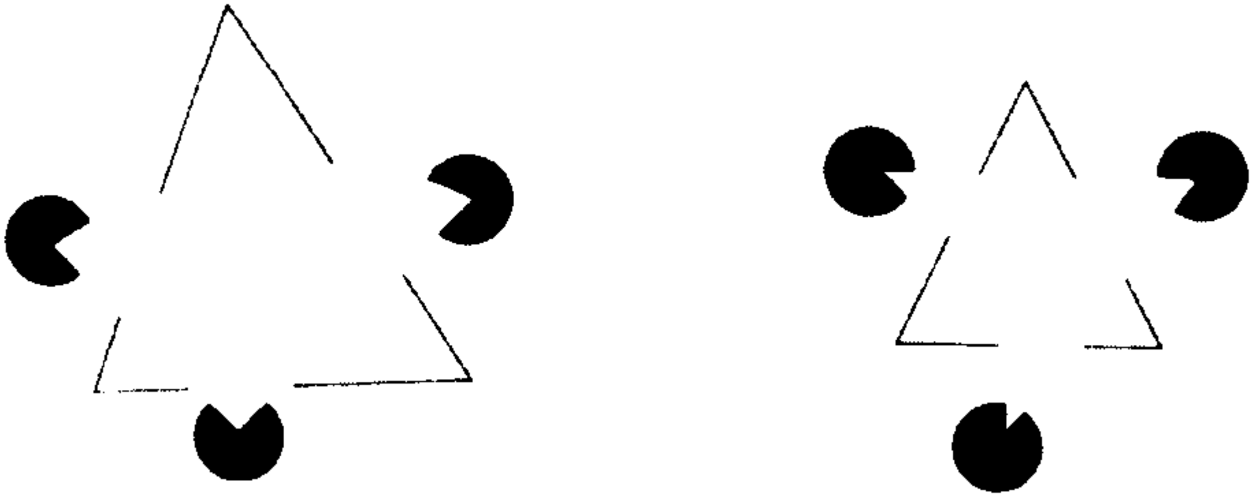
Alberto Paccanaro
Department of Computer Science
Royal Holloway, University of London
www.paccanarolab.org

What is clustering ?

Clustering is grouping things that "go together"

A. Paccanaro, 2019

A. Paccanaro, 2019



A. Paccanaro, 2019

Two questions need to be answered

1. **What do I want to get out of my clustering**
   - Objects in an image
   - Genes with the same function
   - Homologous proteins
   - …

2. **What is that I can measure in the data** (which I hope can answer 1.)
   - Difference in colour between pixels
   - Correlation in gene expression data
   - Sequence distance
   - …

A. Paccanaro, 2019

Clustering Methods

- “Statistical” clustering methods: assume a probabilistic model that generates the observed data points

- “Pairwise” clustering methods: define a similarity function between pairs of points (distance) and then formulates an optimality criterion that the clustering must optimize.

  (the optimality criteria quantify the intuitive notion that points in the same cluster are similar while points in different clusters are dissimilar)

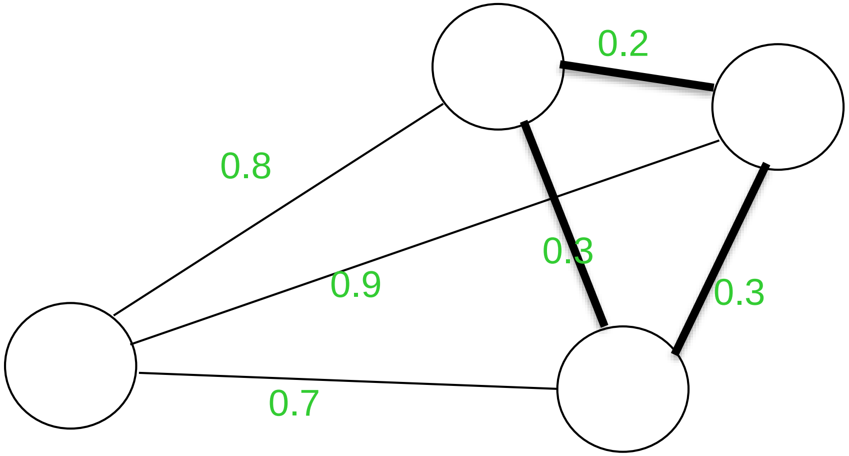
A. Paccanaro, 2019

Clustering as “segmenting” a graph

Pairwise distances between the datapoints as representing the adjacency matrix of a fully connected graph, where:

- nodes are datapoints
- the links are weighted by the distances

clustering → finding areas in the graph which are more “tightly” connected



A. Paccanaro, 2019

Which algorithms we will look at

1. K-means clustering
2. Hierarchical clustering
   - *Single linkage*
   - *Complete linkage*
   - *Average linkage*
3. Connected Components Analysis
4. ClusterONE
5. Spectral clustering

Which algorithms we will look at

1. ~~K-means clustering~~

2. ~~Hierarchical clustering~~
   - ~~*Single linkage*~~
   - ~~*Complete linkage*~~
   - ~~*Average linkage*~~

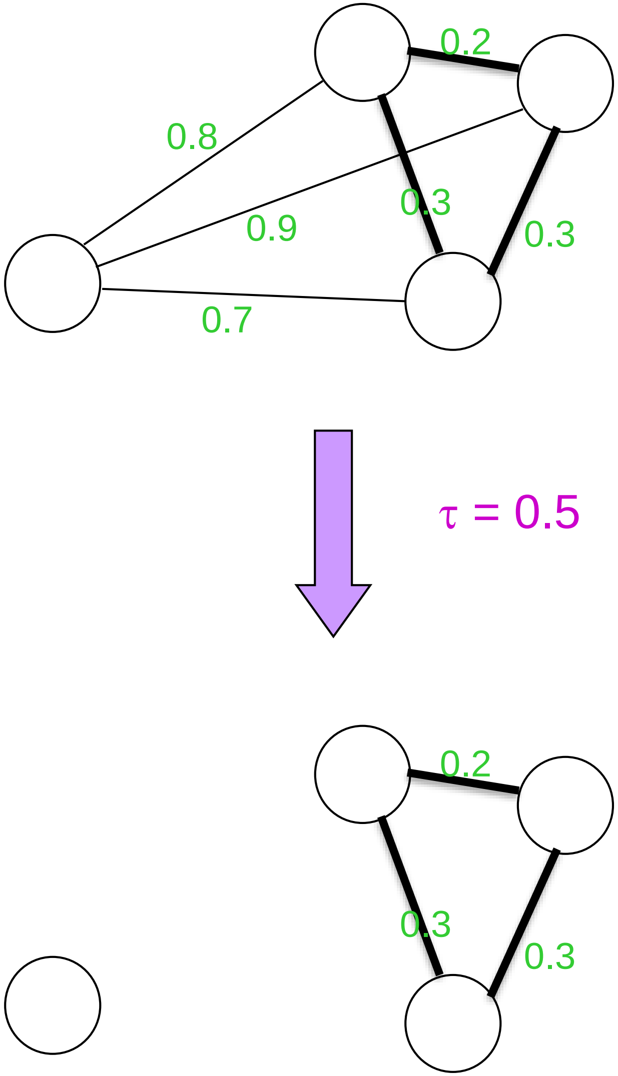
3. Connected Components Analysis

4. ClusterONE

5. Spectral clustering

A. Paccanaro, 2019

Connected Component Analysis



A. Paccanaro, 2019

The CCA algorithm

Think of the problem in terms of a graph where:

- The graph is fully connected
- Each datapoint in your problem is a node
- Links are labelled with the distance between the datapoints

1. Select a threshold τ
2. Erase every link in the graph whose label is greater than τ
3. The clusters are the parts of the graph which are still connected

A. Paccanaro, 2019

Which algorithms we will look at

1. ~~K-means clustering~~

2. ~~Hierarchical clustering~~
   - ~~*Single linkage*~~
   - ~~*Complete linkage*~~
   - ~~*Average linkage*~~

3. ~~Connected Components Analysis~~

4. ClusterONE

5. Spectral clustering

A. Paccanaro, 2019

ClusterONE – Clustering with Overlapping Neighborhood Expansion

The material in the following slides is taken from:
T. Nepusz, H. Yu, A. Paccanaro,
Nature Methods, 2012

- Main features:
  - Can take into account network **weights**
  - Creates **overlapping** clusters
  - Extremely **fast** – it can be applied to large scale networks

Over 16,000 downloads !

- Implementation available from the lab website: www.paccanarolab.org/cluster-one

- Current release uses multiple CPU cores and can now scale up to graphs containing millions of vertices and edges (has been used on 9 million nodes and nearly 100 million edges on a server containing 80 CPU cores and 96 GB of memory).

- We developed it for **detecting protein complexes** from protein interaction networks. It has now become the state-of-the-art method for this problem.

- Other research groups have successfully applied ClusterONE and proved its usefulness in several different domains:
  - Clustering a genome-scale network obtained by integrating SNP array, gene expression microarray, array-CGH, CGH, GWAS and gene mutation data. This study was aimed at identifying key functional modules in **lung adenocarcinoma**.
  - Associating drugs with protein domains in the context of **myocardial infarction**.
  - Studying the mechanisms of adverse **side effects of Torcetrapib,** a drug being developed to treat hypercholesterolemia (elevated cholesterol levels) and prevent cardiovascular disease (its development was halted in 2006).
  - **Detecting communities in Social Networks**.

Human soluble protein complexes

[Havugimana et al, *Cell*, 2012]

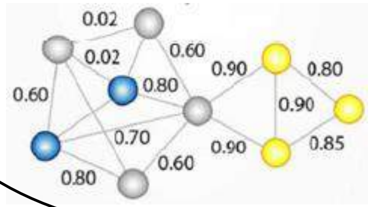
Experimental data

HeLa soluble protein extracts
Fractionated with different techniques
Coeluting proteins identified by MS

A. Emili lab, Un. of Toronto

Functional genomics data
E. Marcotte lab
Un. Texas, Austin

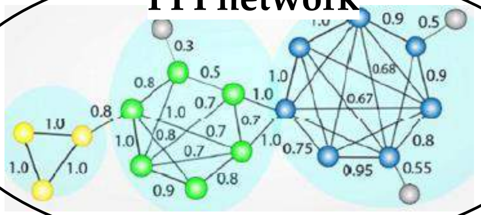
Preliminary PPI network



GO semantic similarity

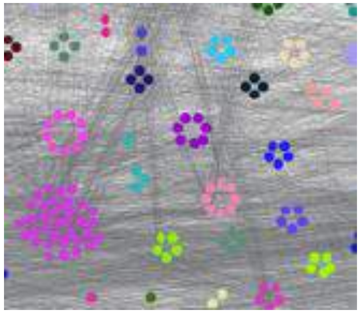
De-noising (diffusion)

PPI network



Finding the complexes (ClusterONE)

Protein Complexes



A. Paccanaro, 2019

The problem

- Cluster a large graph
- Edges are undirected
- Edges are **weighted**
- Nodes can appear in more than one cluster – **overlapping clustering**

A. Paccanaro, 2019

The ClusterONE algorithm – 3 phases

1. **Cluster Growth**: Cluster candidates are grown from selected seed nodes, independently of each other. Growth is driven by the greedy maximisation of a **goal function**.

2. **Cluster Merging:** Highly similar cluster candidates are merged into larger clusters.

3. **Cluster post-processing:** Cluster candidates are finally post-processed using several simple criteria (size, density, etc.)

A. Paccanaro, 2019

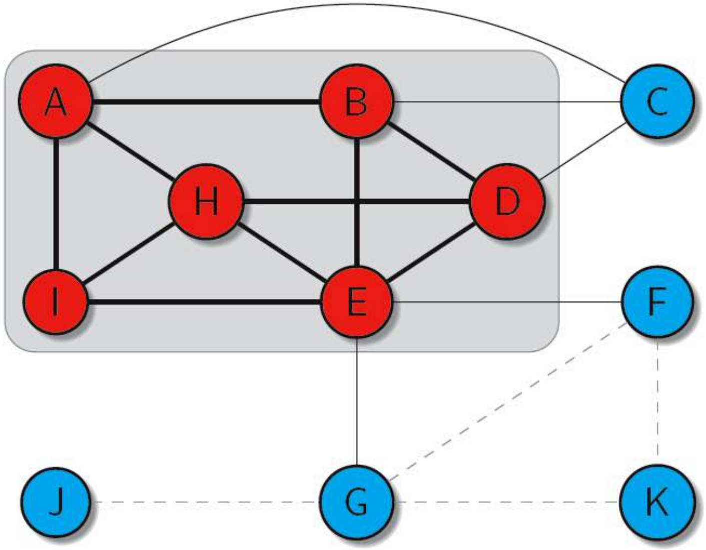
Step 1. Cluster Growth

A cluster should satisfy two structural properties:

a. **contain many reliable interactions** between its nodes
b. be **well-separated** from the rest of the network

Cohesiveness:

total weight of internal edges, divided by the total weight of internal or boundary edges.



A. Paccanaro, 2019

Cohesiveness measures how likely it is for a group of nodes to form a cluster

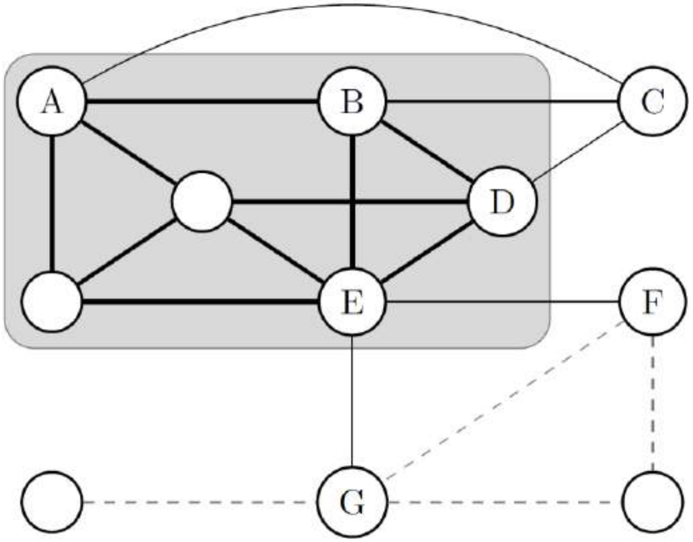
The cohesiveness function

$$f(V) = \frac{w^{in}(V)}{w^{in}(V) + w^{bound}(V) + p|V|}$$

- $w_{in}(V)$ the total weight of edges contained entirely by a cluster V
- $w_{bound}(V)$ the total weight of edges that connect the cluster with the rest of the network.
- $p|V|$ is a penalty term

A. Paccanaro, 2019

Details of cluster growth



v_0 = node with the highest degree among those that have not been included in any complex so far.

Greedy growth procedure :

1. Let $V_0 = \{v_0\}$. Set the step number $t = 0$.
2. Calculate the cohesiveness of V_t and let $V_{t+1} = V_t$.
3. For every <u>external</u> vertex v on a boundary edge, $V' = V_t \cup \{v\}$. If $f(V') > f(V_{t+1})$, then $V_{t+1} = V'$.
4. For every <u>internal</u> vertex v on a boundary edge, $V'' = V_t \setminus \{v\}$. If $f(V'') > f(V_{t+1})$, then $V_{t+1} = V''$.
5. If $V_t \neq V_{t+1}$, increase t and return to step 2.
   Otherwise, declare V_t a locally optimal cohesive group.

A. Paccanaro, 2019

Step 2. Cluster Merging

We merge pairs of putative clusters whose overlap score ω is greater than a given threshold.

The overlap score of two putative clusters A and B is defined as:

$$\omega(A,B) = \frac{|A \cap B|^2}{|A|\,|B|}$$

A. Paccanaro, 2019

Step 3. Cluster Postprocessing

Clusters are further analyzed and selected according to:

1. Size
2. Density
3. Other user parameters (e.g., in the case of protein clusters, functional enrichment)

In our implementation for detecting protein complexes, we discard complex candidates that:

a. contain less than 3 proteins

b. whose density $\delta = 2E_I/n(n\text{-}1) < \tau_2$*, where n is the number of proteins and* E_I *the total weight of internal edges.*

A. Paccanaro, 2019

Evaluation

Comparison with a gold standard is difficult:

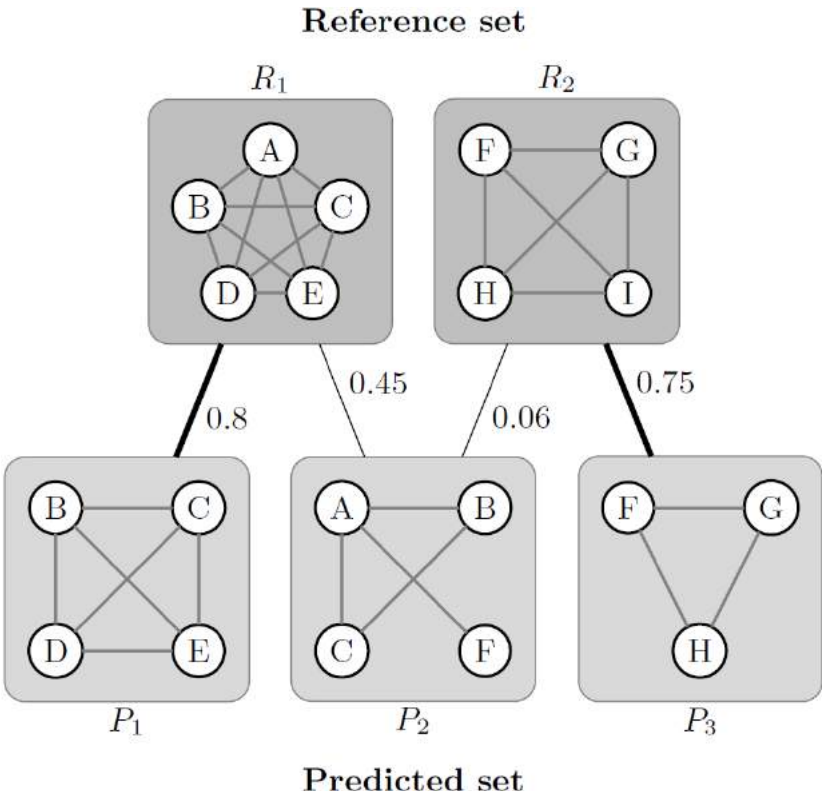
- ❑ matches are often only partial
- ❑ many-to-one and one-to-many matches
- ❑ gold standard is incomplete

Measures wrt gold standard:

1. the Maximum Matching Ratio (MMR)
2. clustering-wise sensitivity (Sn), positive predicted value (PPV) and geometric accuracy $Acc = \sqrt{Sn \times PPV}$ (Brohee, BMC Bioinf. 2006)
3. number of matched complexes with $\omega > 0.25$

A. Paccanaro, 2019

The Maximum Matching Ratio (MMR)



A. Paccanaro, 2019

1. bipartite graph (reference and predicted complexes sets)
2. select the maximum weighted (overlap score) bipartite matching
3.

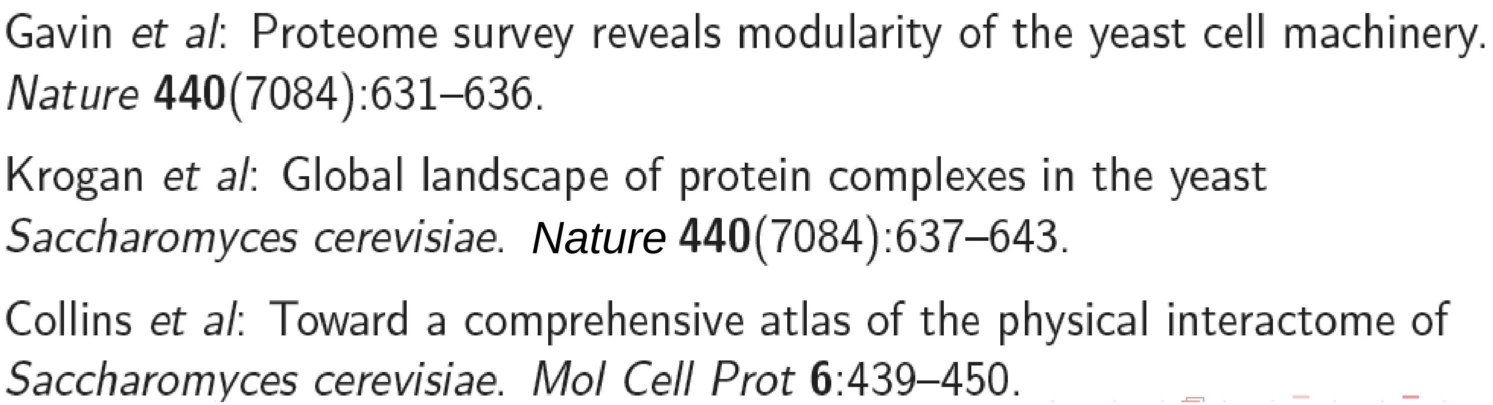
$$MMR = \frac{\text{total weight of selected edges}}{\text{number of reference complexes}}$$

Results using ClusterONE

PPI datasets for benchmarking

1. Gavin 1430 proteins, 6531 interactions. Large-scale AP-MS experimental data on yeast.
2. Krogan core 2708 proteins, 7123 interactions. Large-scale AP-MS experimental data on yeast.
3. Krogan extd 3672 proteins, 14137 interactions. Same as Krogan core, different threshold.
4. Collins 1622 proteins, 9074 interactions. Combined Gavin and Krogan.

Data sources:



Gavin *et al*: Proteome survey reveals modularity of the yeast cell machinery. *Nature* **440**(7084):631–636.

Krogan *et al*: Global landscape of protein complexes in the yeast *Saccharomyces cerevisiae*. *Nature* **440**(7084):637–643.

Collins *et al*: Toward a comprehensive atlas of the physical interactome of *Saccharomyces cerevisiae*. *Mol Cell Prot* **6**:439–450.

A. Paccanaro, 2019

Competing algorithms

Non-overlapping

- Affinity Propagation – Frey *et al*, Science (2007)
- MCL – Enright *et al*, NAR (2002)
- RNSC – King *et al*, Bioinformatics (2004)

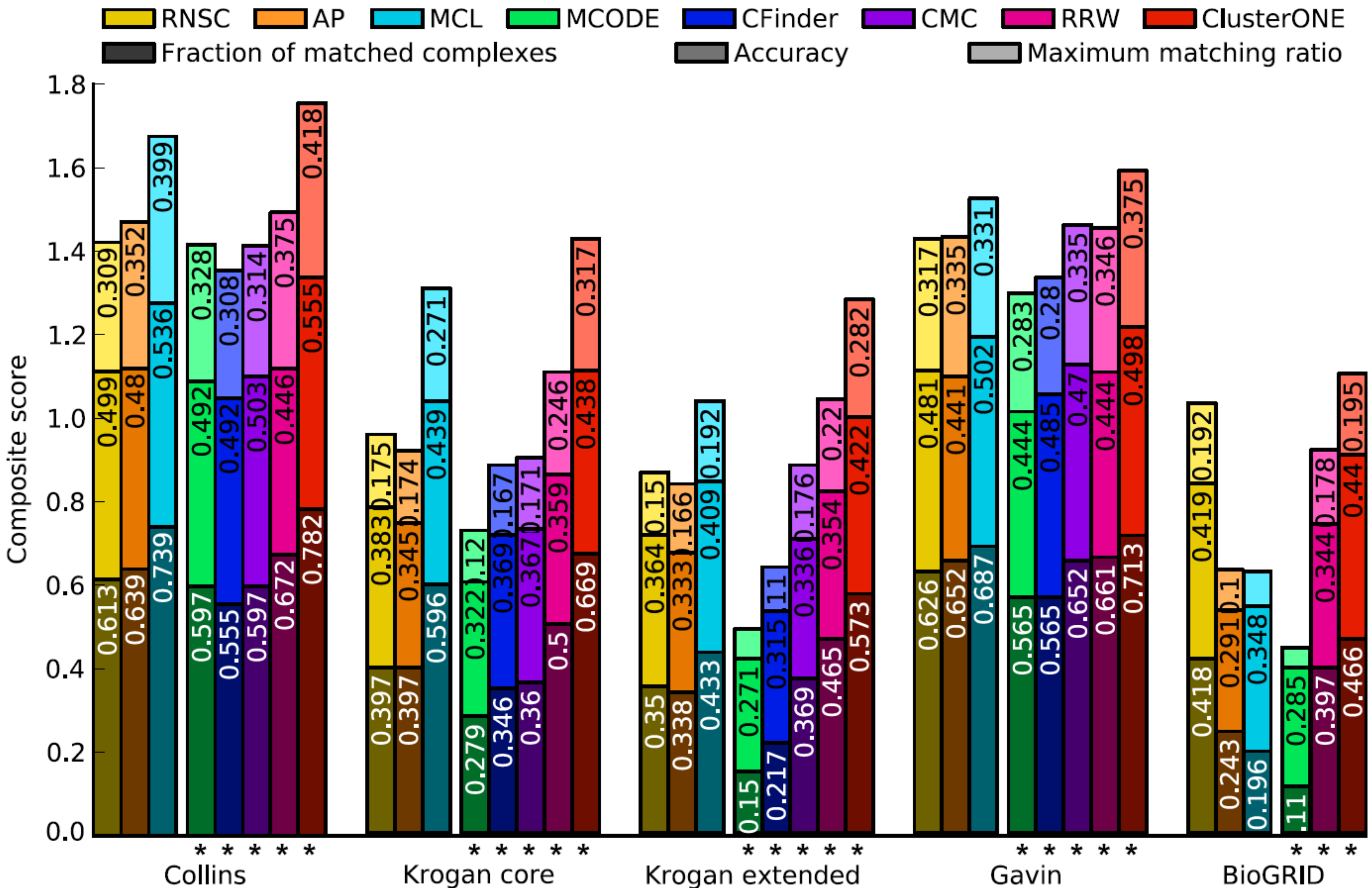
Overlapping

- CFinder – Palla *et al*, Nature (2005)
- CMC – Liu *et al*, Bioinformatics (2009)
- RRW – Macropol *et al*, Bioinformatics (2009)
- MCODE – Bader *et al*, Bioinformatics (2003)

The parameters for all the above algorithms were optimized
ClusterONE run was with the default parameters

A. Paccanaro, 2019

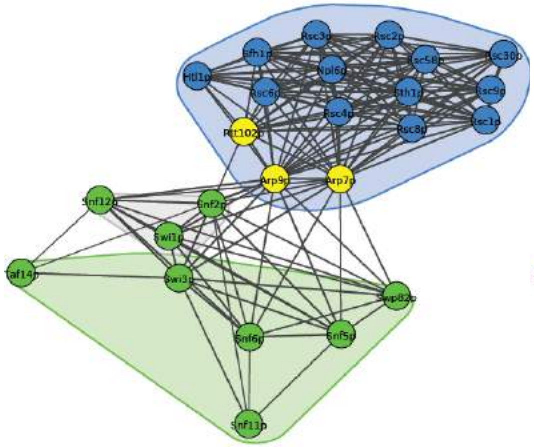
Results wrt the MIPS gold standard



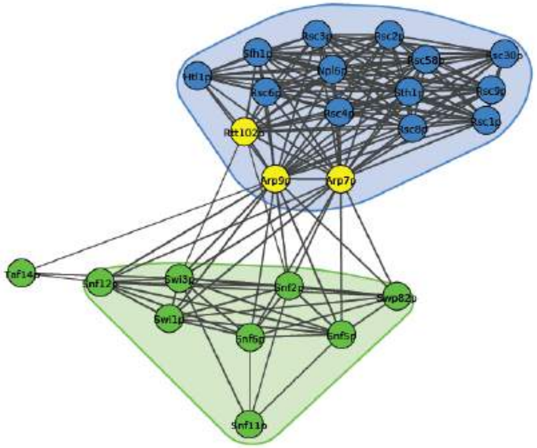
A. Paccanaro, 2019

The RSC and SW1/SNF chromatin remodelling complexes

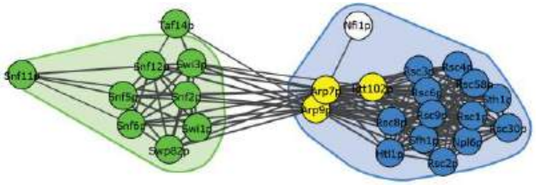
[Collins dataset]



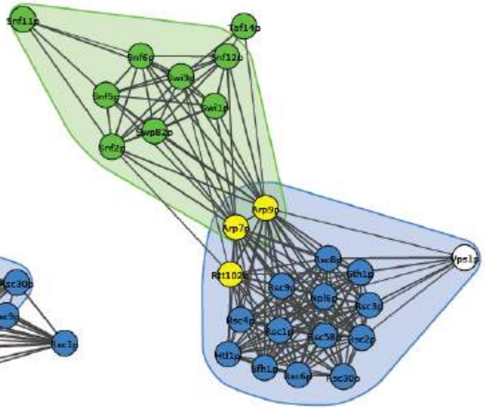
Affinity Propagation



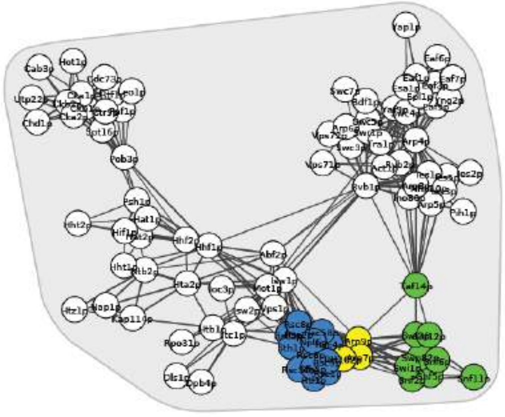
RNSC



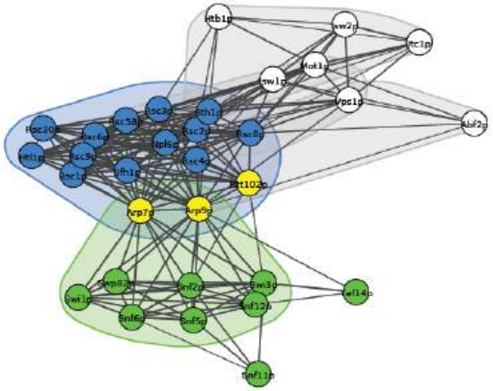
MCL



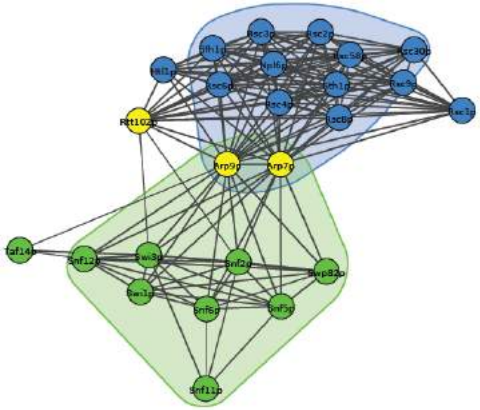
ClusterONE



CFinder



CMC



RRW

🔵 RSC only 🟢 SWI/SNF only 🟡 Both complexes ⚪ Not in complexes

Which algorithms we will look at

1. ~~K-means clustering~~

2. ~~Hierarchical clustering~~
   - ~~*Single linkage*~~
   - ~~*Complete linkage*~~
   - ~~*Average linkage*~~

3. ~~Connected Components Analysis~~

4. ~~ClusterONE~~

5. Spectral clustering

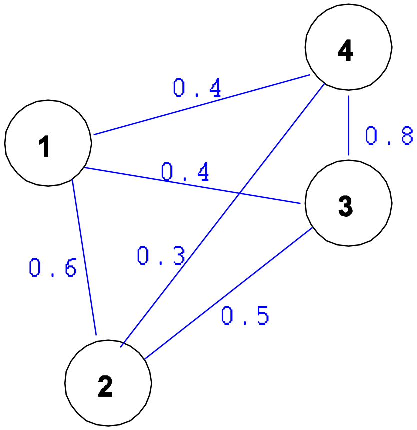
A. Paccanaro, 2019

Spectral Clustering

the basic idea

The material in the following slides is taken from:
A. Paccanaro, J. A. Casbon, and M. A. Saqi
Nucleic Acids Research, vol. 34, 2006

	1	2	3	4
1	1	0.6	0.4	0.4
2	0.6	1	0.5	0.3
3	0.4	0.5	1	0.8
4	0.4	0.3	0.8	1

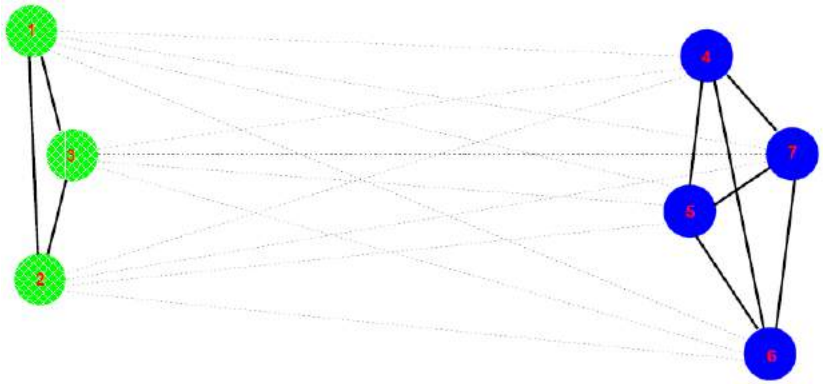


A. Paccanaro, 2019

Eigenvalues and eigenvectors of a matrix derived from affinities provide a basis for deciding on a particular segmentation

Spectral Clustering
the Markov chain perspective

$\boldsymbol{p}_0$ initial distribution of a particle.



The probability distribution at the next time step is:

$$\mathbf{p}_1 = M \cdot \mathbf{p}_0$$

where: $M = SD^{-1}$

is the Markov transition probability matrix

The probability distribution after β iterations is:

$$\mathbf{p}_\beta = M \cdot \mathbf{p}_{(\beta-1)} = M \cdot \left(M \cdot \mathbf{p}_{(\beta-2)}\right) = \ldots = M^\beta \cdot \mathbf{p}_0$$

Therefore, to see what happens to the particle during the random walk, **we need to analyze $\mathbf{M}^\beta$**

A. Paccanaro, 2019

- For analysis, consider the *similar* matrix L:

$$\begin{aligned} L &\overset{\text{def}}{\equiv} D^{-1/2} M D^{1/2} \\ &= D^{-1/2} S D^{-1} D^{1/2} \\ &= D^{-1/2} S D^{-1/2} \end{aligned}$$

$\Rightarrow L$ symmetric $\Rightarrow L = U \Lambda U^{-1} = U \Lambda U^T$
where $U = [\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_n]$ eigenvectors, Λ diagonal matrix of eigenvalues

- We can write M as:

$$M = D^{1/2} L D^{-1/2} = D^{1/2} U \, \Lambda \, U^T D^{-1/2}$$

$$M^\beta = D^{1/2} U \, \Lambda^\beta \, U^T D^{-1/2}$$

$$\boxed{= \sum_{i=1}^{n} D^{1/2} \mathbf{u}_i \lambda_i^\beta \mathbf{u}_i^T D^{-1/2}}$$

A. Paccanaro, 2019

(1) What happens after an infinite number of iterations?

$$M^{\beta} = D^{1/2} U \ \Lambda^{\beta} \ U^T D^{-1/2} = \sum_{i=1}^{n} D^{1/2} \mathbf{u}_i \lambda_i^{\beta} \mathbf{u}_i^T D^{-1/2}$$

$$M^{\infty} = D^{1/2} \mathbf{u}_1 \ \mathbf{u}_1^T D^{-1/2}$$

The leading eigenvector of *L* is: $\mathbf{u}_1 = \frac{\sqrt{\mathbf{d}}}{\sqrt{\sum_i d_i}}$

Therefore:
$$\begin{aligned} M^{\infty} &= D^{1/2} \frac{\sqrt{\mathbf{d}}}{\sqrt{\sum_i d_i}} \frac{\sqrt{\mathbf{d}^T}}{\sqrt{\sum_i d_i}} D^{-1/2} \\ &= \frac{\mathbf{d}}{\sum_i d_i} \cdot \mathbf{1}^T \\ &= [\boldsymbol{\pi} \ \boldsymbol{\pi} \ \ldots \ \boldsymbol{\pi}] \end{aligned}$$

$$\boldsymbol{\pi} = \frac{\mathbf{d}}{\sum_i d_i}$$

is the leading eigenvector of M

A. Paccanaro, 2019

Therefore, for any initial distribution $\boldsymbol{p}_0$ we always reach the same stationary distribution $\boldsymbol{\pi}$

$$\boldsymbol{p}_\infty = M^\infty \boldsymbol{p}_0 = \boldsymbol{\pi}$$

$$\begin{aligned}
\mathbf{u}_1 &= D^{-1/2} S D^{-1/2} \mathbf{u}_1, \\
&= D^{-1/2} S D^{-1/2} \sqrt{\frac{\mathbf{d}}{\sum d_i}}, \\
&= D^{-1/2} S \frac{\mathbf{1}}{\sqrt{\sum d_i}}, \qquad \left(\because \qquad D^{-1/2}\sqrt{\mathbf{d}} = \mathbf{1}\right) \\
&= D^{-1/2} \frac{\mathbf{d}}{\sqrt{\sum d_i}}, \qquad (\because \qquad S\mathbf{1} = \mathbf{d}\ ) \\
&= \sqrt{\frac{\mathbf{d}}{\sum d_i}},
\end{aligned}$$

$$\begin{aligned}
\boldsymbol{\pi} \quad &= \quad M \cdot \boldsymbol{\pi} \\
&= \quad S \cdot D^{-1} \frac{\mathbf{d}}{\sum_i d_i} \\
&= \quad S \cdot \mathbf{1} \cdot \frac{1}{\sum_i d_i} \\
&= \quad \frac{\mathbf{d}}{\sum_i d_i} \\
&= \quad \boldsymbol{\pi}
\end{aligned}$$

A. Paccanaro, 2019

(2) What happens to $\mathbf{p}_0$ after β iterations ?

$$M^{\beta} = D^{1/2}\mathbf{u}_1\ \mathbf{u}_1^{\mathrm{T}} D^{-1/2} + \sum_{i=2}^{n} D^{1/2}\mathbf{u}_i \lambda_i^{\beta} \mathbf{u}_i^{\mathrm{T}} D^{-1/2}$$

$$= M^{\infty} + \sum_{i=2}^{n} D^{1/2}\mathbf{u}_i \lambda_i^{\beta} \mathbf{u}_i^{\mathrm{T}} D^{-1/2}$$

$$\boldsymbol{p}^{\beta} = D^{1/2}\mathrm{u}_1\ \mathrm{u}_1^{T} D^{-1/2}\mathrm{p}_0 + \sum_{i=2}^{n} D^{1/2}\lambda_i^{\beta}\mathrm{u}_i\mathrm{u}_i^{T} D^{-1/2}\mathrm{p}_0$$

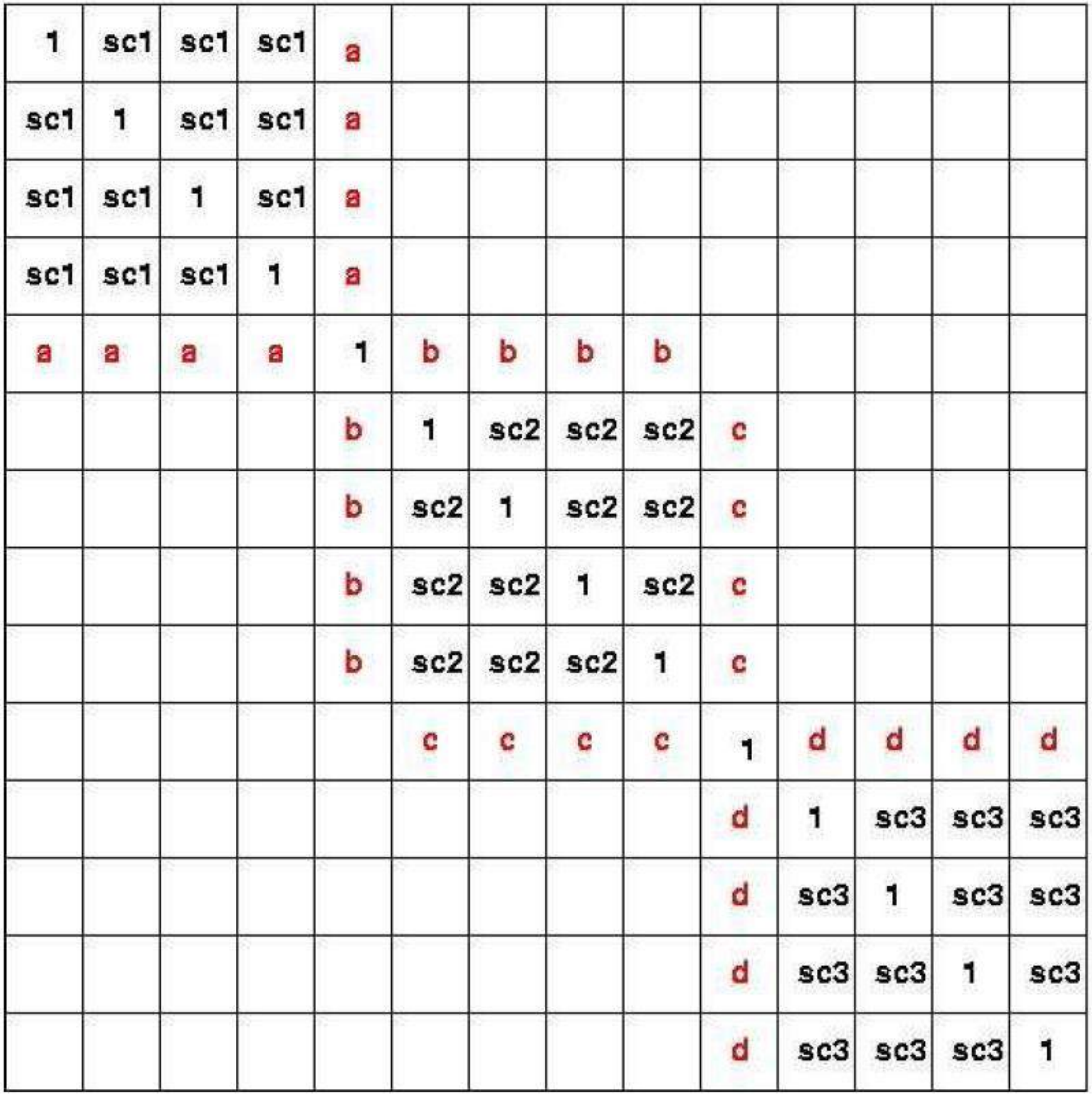
$$= \pi + \sum_{i=2}^{n} D^{1/2}\lambda_i^{\beta}\mathrm{u}_i\mathrm{u}_i^{T} D^{-1/2}\mathrm{p}_0$$

A. Paccanaro, 2019

1. **Markovian relaxation process as perturbations to the stationary distribution!**
2. **condition of piecewise constancy on the form of the leading eigenvectors**

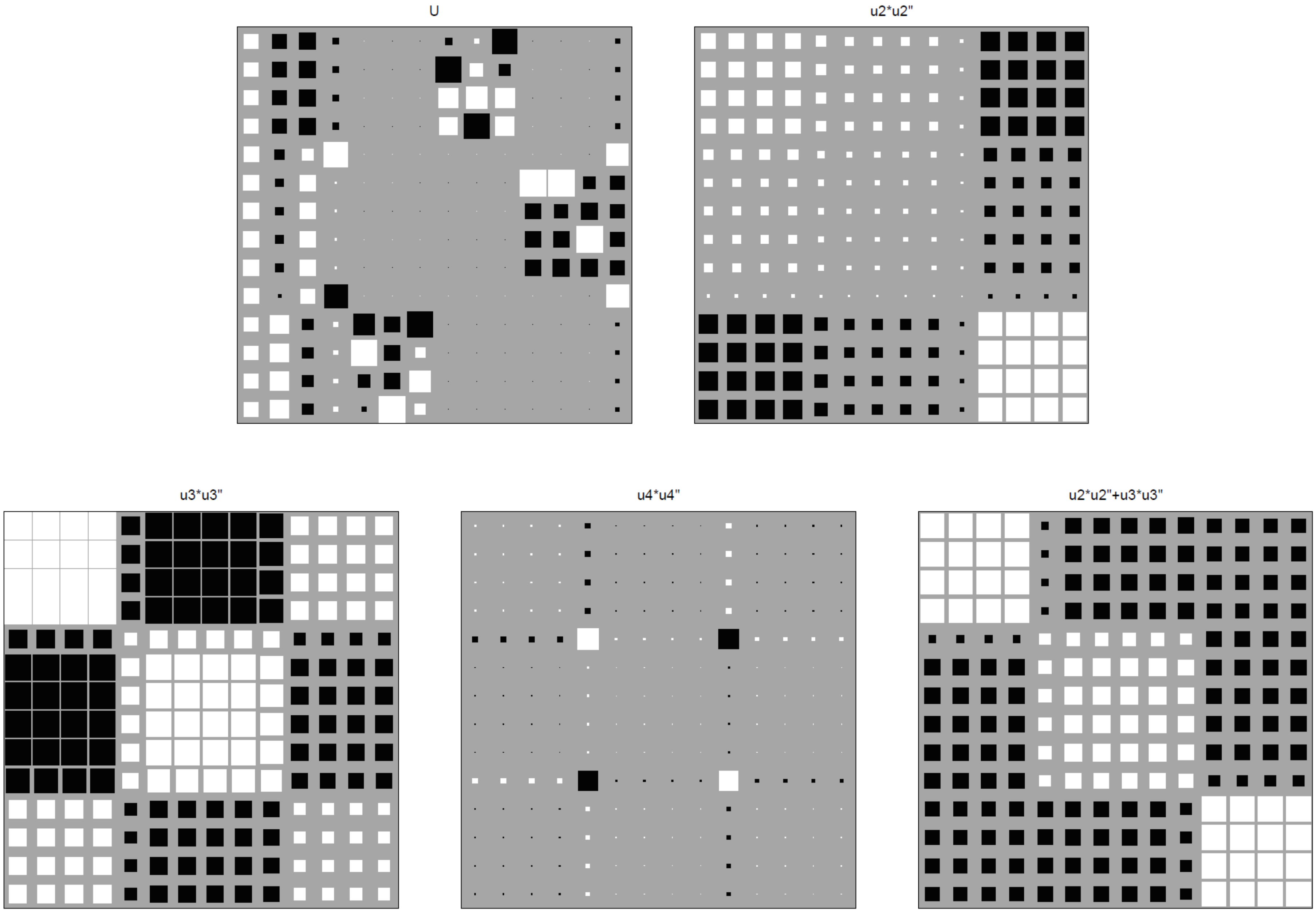
sc1=0.9
sc2=1
sc3=0.8

a=0.2
b=0.7
c=0.8
d=0.1



1	sc1	sc1	sc1	a									
sc1	1	sc1	sc1	a									
sc1	sc1	1	sc1	a									
sc1	sc1	sc1	1	a									
a	a	a	a	1	b	b	b	b					
				b	1	sc2	sc2	sc2	c				
				b	sc2	1	sc2	sc2	c				
				b	sc2	sc2	1	sc2	c				
				b	sc2	sc2	sc2	1	c				
					c	c	c	c	1	d	d	d	d
									d	1	sc3	sc3	sc3
									d	sc3	1	sc3	sc3
									d	sc3	sc3	1	sc3
									d	sc3	sc3	sc3	1

A. Paccanaro, 2019



U

u2*u2"

u3*u3"

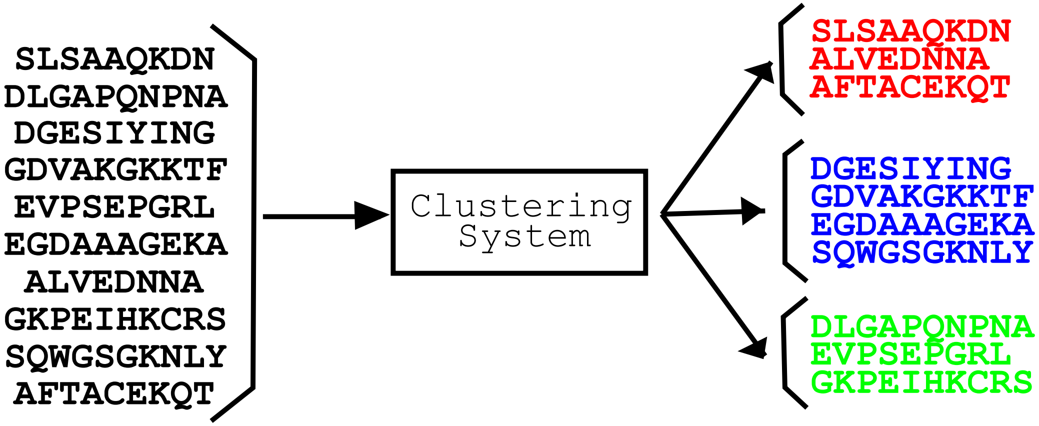
u4*u4"

u2*u2"+u3*u3"

A. Paccanaro, 2019

The problem

- Given a set of protein sequences, automatically group them based on their functional similarity



SLSAAQKDN
DLGAPQNPNA
DGESIYING
GDVAKGKKTF
EVPSEPGRL
EGDAAAGEKA
ALVEDNNA
GKPEIHKCRS
SQWGSGKNLY
AFTACEKQT

Clustering System

SLSAAQKDN
ALVEDNNA
AFTACEKQT

DGESIYING
GDVAKGKKTF
EGDAAAGEKA
SQWGSGKNLY

DLGAPQNPNA
EVPSEPGRL
GKPEIHKCRS

A. Paccanaro, 2019

The core of most methods was based on simply thresholding a measure related to the distance between the sequences

The algorithm

Assign proteins to clusters based on the value of the elements of $\mathbf{u}_i$

1. we use the eigengap to guess the number of clusters k (ratio of successive eigenvalues)

2. we use the first k eigenvectors to map the proteins onto points in R^k; normalize these points to unit length; then cluster using K-Means (Ng et al, NIPS, 2000)

A. Paccanaro, J. A. Casbon, and M. A. Saqi
Spectral clustering of protein sequences
Nucleic Acids Research, vol. 34, iss. 5, pp. 1571-1580, 2006

A. Paccanaro, 2019

Learning to discriminate e-values

(= adding a bit of background knowledge)

- build a dataset of distances:

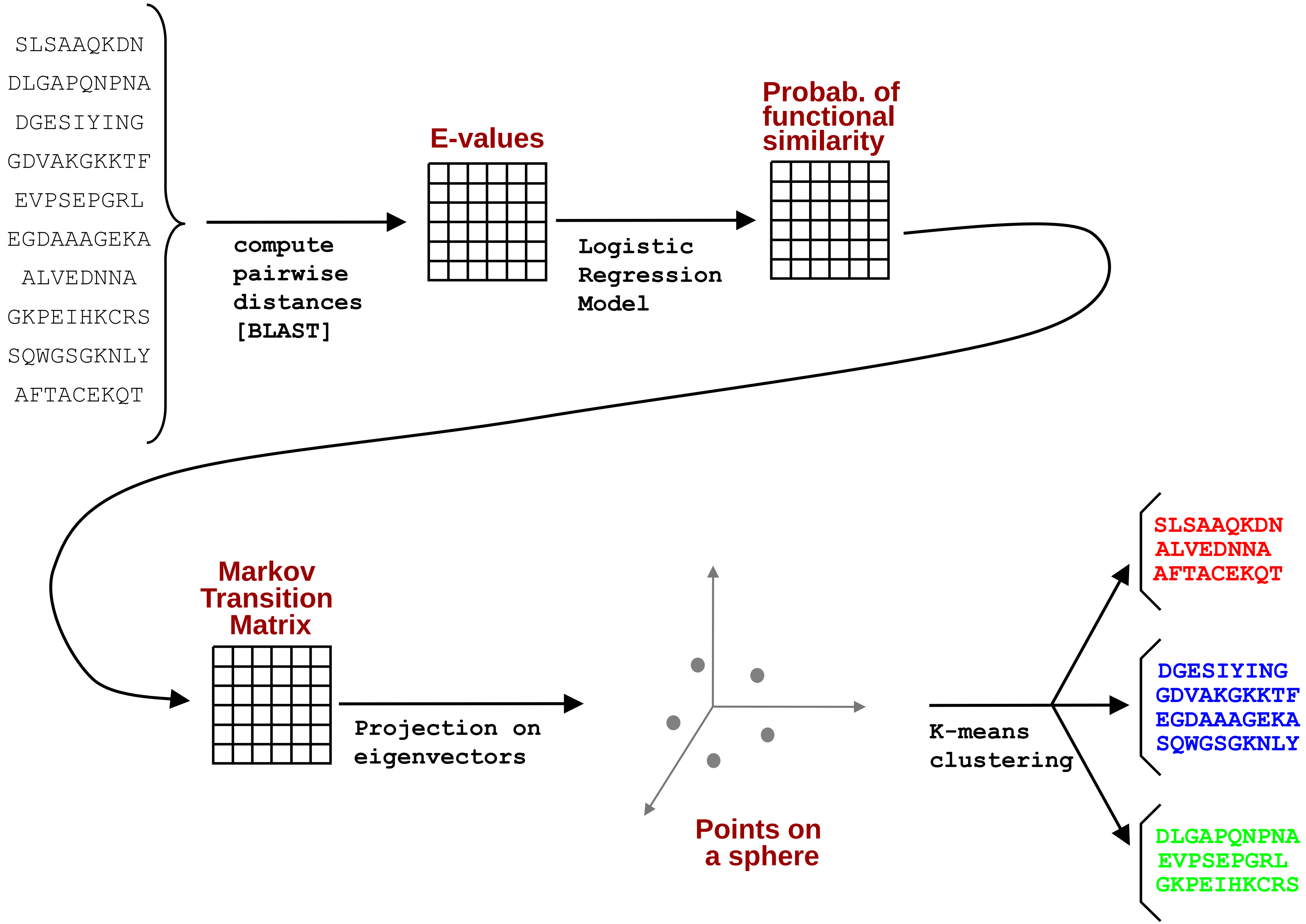
  Class 1: distances between proteins of the same super-family

  Class 2: distances between proteins in two different super-families

- learn a logistic regression model to discriminate between 2 classes

➔ the posterior probabilities returned by the model can be seen as probabilities of functional relatedness

Outline of the method



SLSAAQKDN
DLGAPQNPNA
DGESIYING
GDVAKGKKTF
EVPSEPGRL
EGDAAAGEKA
ALVEDNNA
GKPEIHKCRS
SQWGSGKNLY
AFTACEKQT

compute pairwise distances [BLAST]

E-values

Logistic Regression Model

Probab. of functional similarity

Markov Transition Matrix

Projection on eigenvectors

Points on a sphere

K-means clustering

SLSAAQKDN
ALVEDNNA
AFTACEKQT

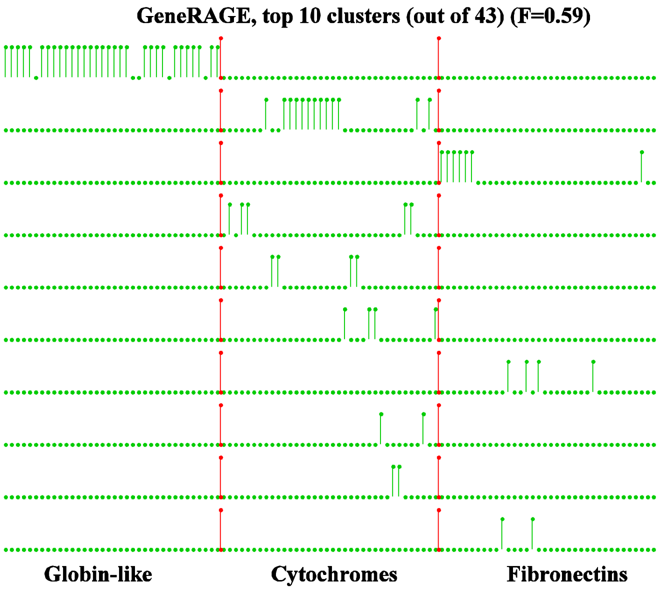
DGESIYING
GDVAKGKKTF
EGDAAAGEKA
SQWGSGKNLY

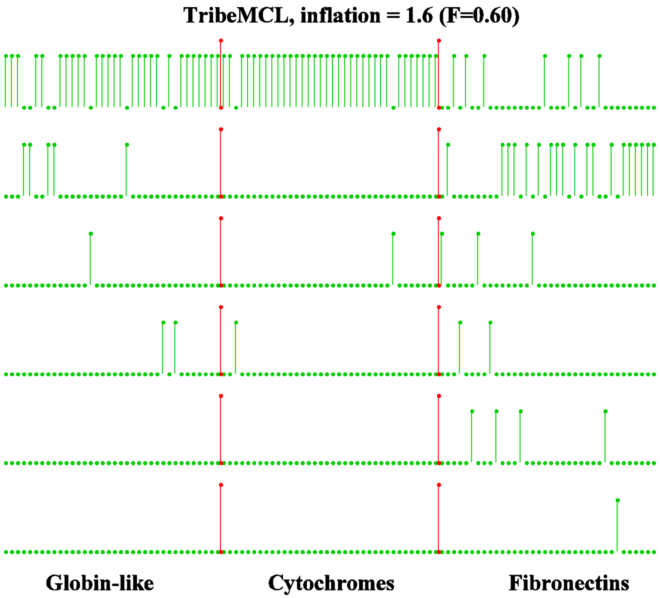
DLGAPQNPNA
EVPSEPGRL
GKPEIHKCRS

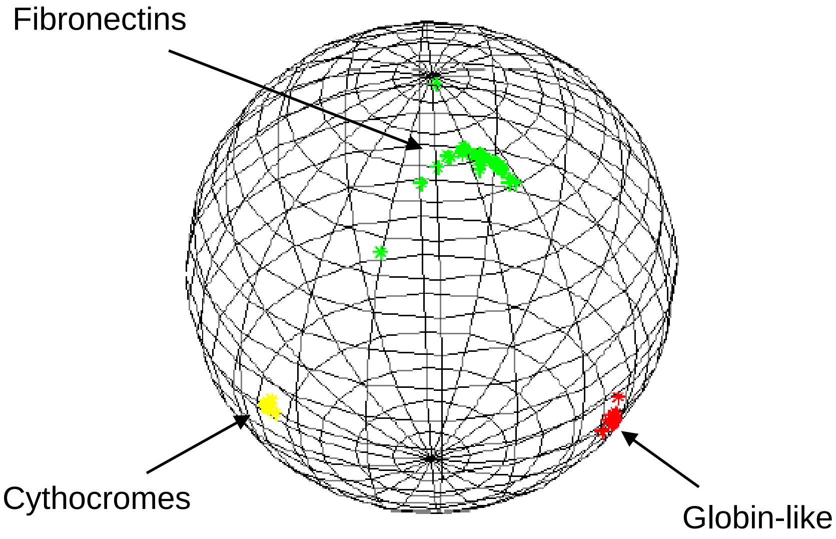
A. Paccanaro, 2019

[Paccanaro et al, Nucleic Acids Research, 2006]

Results: 108 proteins, 3 super-families, Astral 40

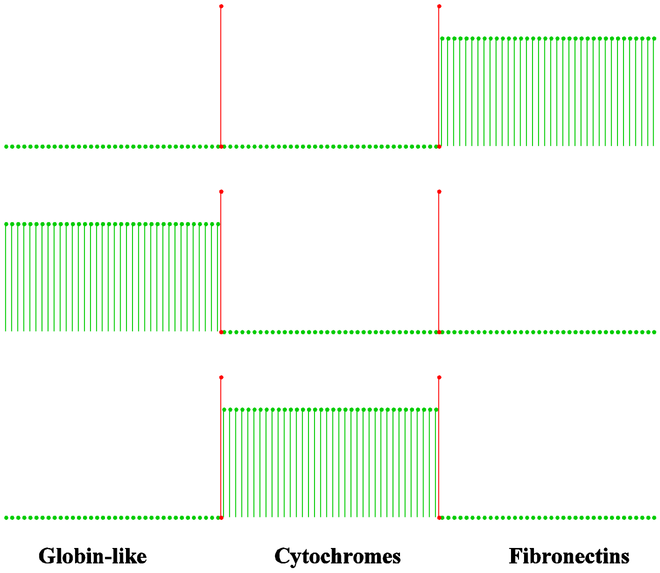

GeneRAGE, top 10 clusters (out of 43) (F=0.59)


TribeMCL, inflation = 1.6 (F=0.60)

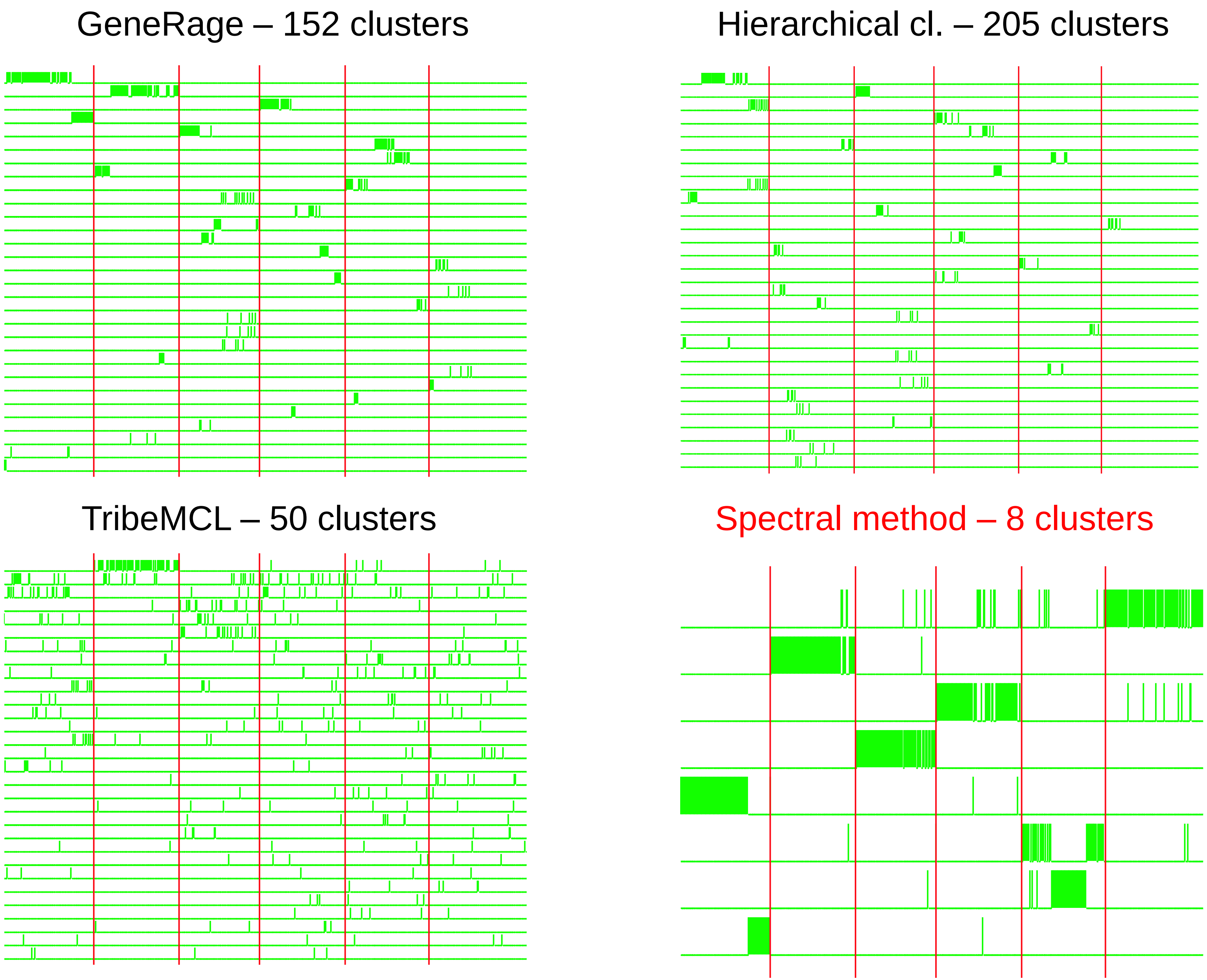

Fibronectins

Cythocromes

Globin-like



A. Paccanaro, 2019



GeneRage – 152 clusters

Hierarchical cl. – 205 clusters

TribeMCL – 50 clusters

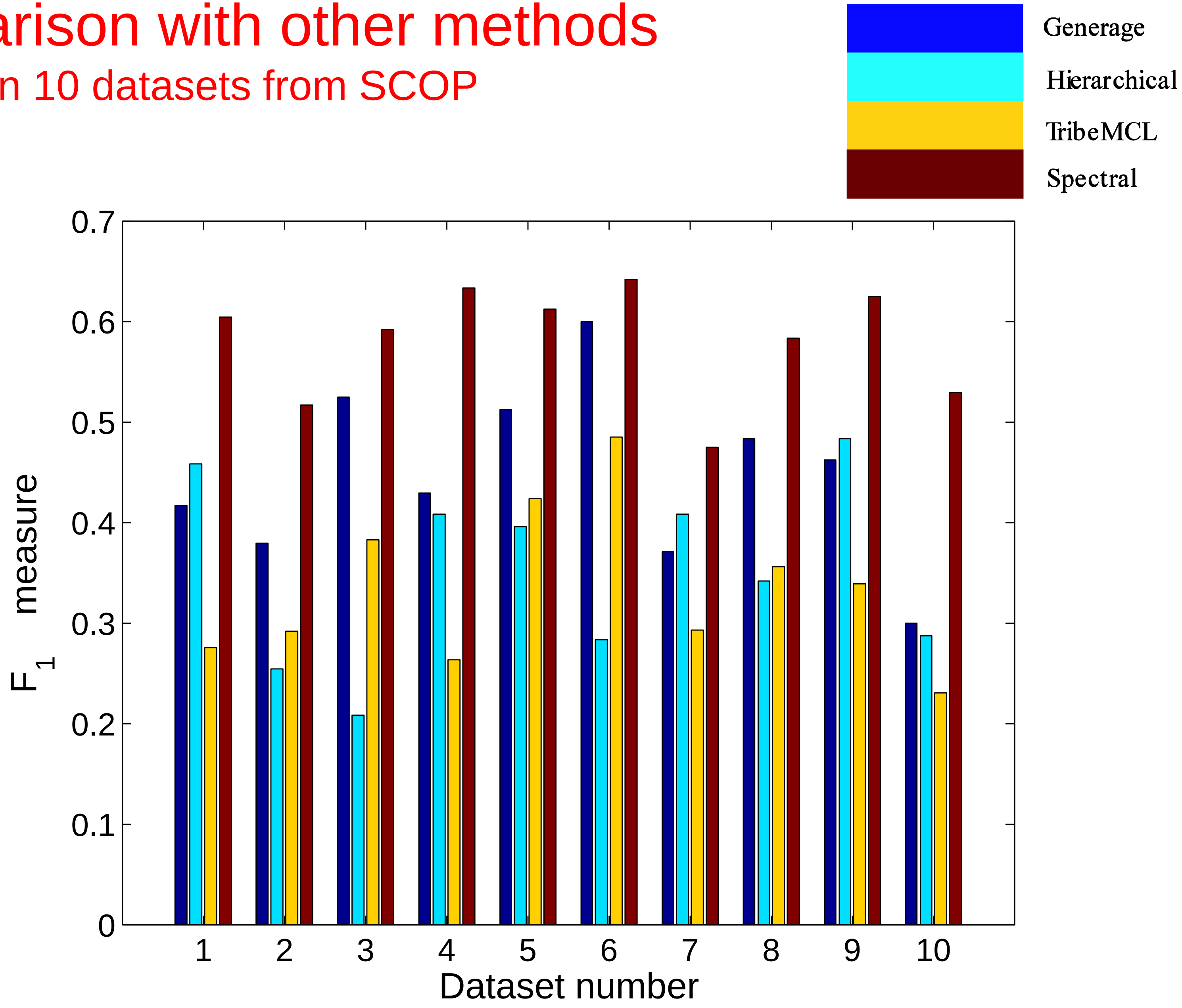
Spectral method – 8 clusters

Globin-like(88), EF-hand(83), Cupredoxins(78), (Trans)glycosidases(83), Thioredoxin-like(81), Membrane all-alpha(94)

A. Paccanaro, 2019

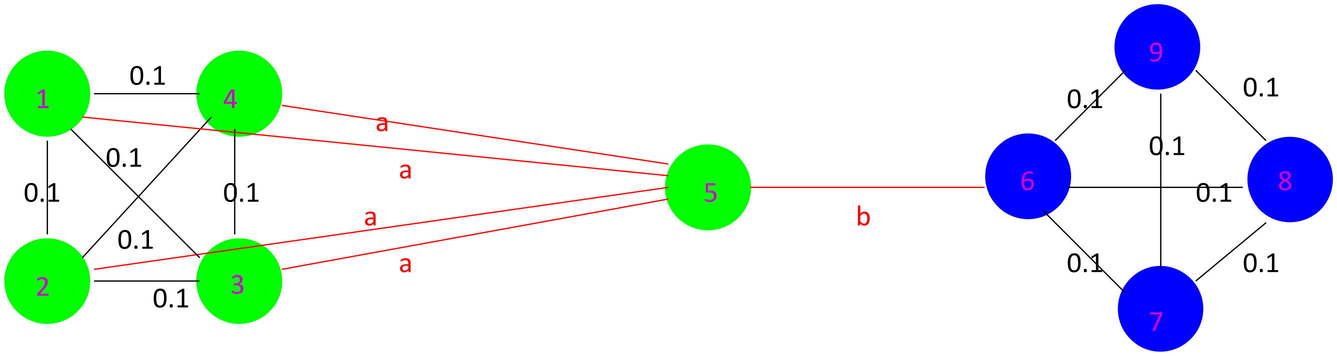
Comparison with other methods

Results on 10 datasets from SCOP



A. Paccanaro, 2019

Conclusion – why does spectral clustering works so much better for clustering protein sequences?



a=0.5, b=0.3

the spectral clustering is still correct

- Spectral clustering looks at **global** properties in the affinity matrix, and this makes it more robust to noise

- Local methods, that decide the grouping based on the value of one (or a few) sequence similarities, are very sensitive to this noise

A. Paccanaro, 2019

SCPS: a fast implementation of a spectral method for detecting protein families on a genome-wide scale *[Nepusz et al. BMC Bioinformatics 2010]*

spectral
clustering of
protein
sequences

- Simple, clean and user-friendly **graphical user interface** (requires no background knowledge in the details of spectral clustering)

- SCPS is also able to perform
  - **connected component analysis**
  - **hierarchical clustering**
  - **TribeMCL**
  - **provides different cluster quality scores**

- SCPS Interfaces with:
  - **BLAST**
  - **Cytoscape**

- **Extremely efficient** and its speed scales well with the size of the dataset
- Produces **publication-quality graphical representations of the clusters**
- included a **sophisticated command line interface** (for automated batch jobs)

- **SCPS was written in C++ and is distributed as an open-source package. Precompiled executables are available for the three major operating systems (Windows, Linux and Mac OS X) at http://www.paccanarolab.org/software/scps**

A. Paccanaro, 2019

Material

(from which I took some of the figures in these slides)

- Tamás Nepusz, Haiyuan Yu, Alberto Paccanaro
  Detecting overlapping protein complexes in protein-protein interaction networks
  Nature Methods (2012) -- doi:10.1038/nmeth.1938
  Code available from the lab website at: http://www.paccanarolab.org/cluster-one/

- A. Paccanaro, J. A. Casbon, and M. A. Saqi
  Spectral clustering of protein sequences
  Nucleic Acids Research, vol. 34, iss. 5, pp. 1571-1580, 2006

- T. Nepusz, R. Sasidharan, and A. Paccanaro
  SCPS: a fast implementation of a spectral method for detecting protein families on a genome-wide scale
  BMC Bioinformatics, vol. 11, iss. 1, p. 120, 2010.
  Code available from the lab website at: http://www.paccanarolab.org/software/scps

A. Paccanaro, 2019