Bioinformatics 2 - Lecture 3

Guido Sanguinetti

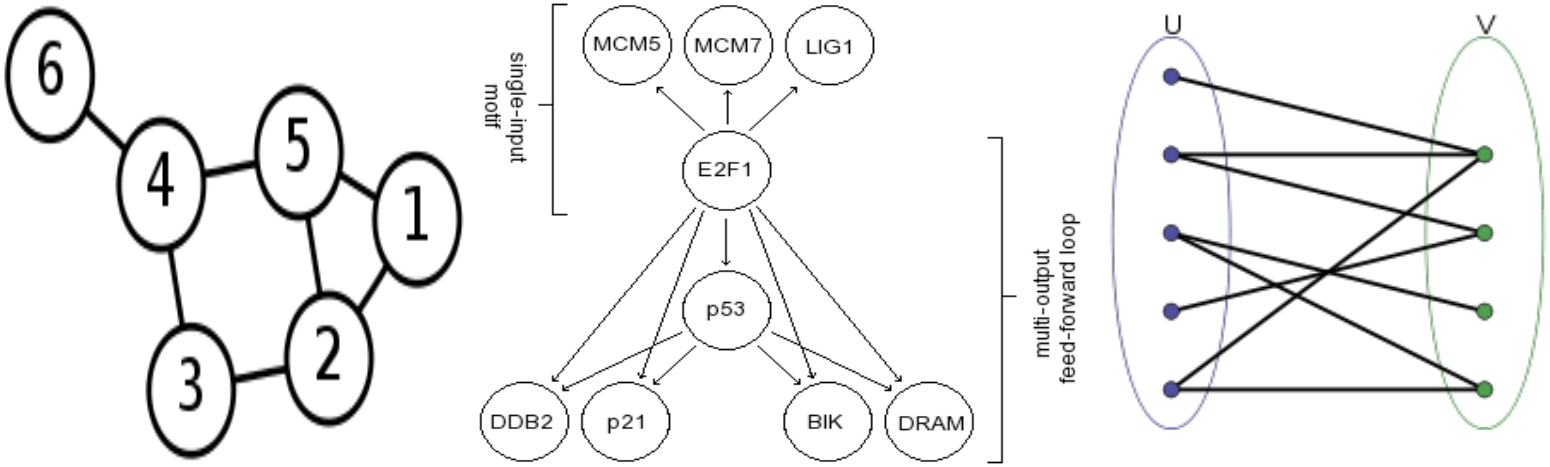
School of Informatics
University of Edinburgh

January 24, 2012

Networks

- A graph or network is a pair (V, E)
- V is a set of nodes or vertices, E is a set of edges or links connecting the nodes
- If the edges are associated with a direction, we have a *directed network*
- If the nodes can be partitioned in two sets A and B, and all edges are between A and B, we have a *bipartite network*

Example networks



Three networks: undirected (left), directed (centre), bipartite (right).

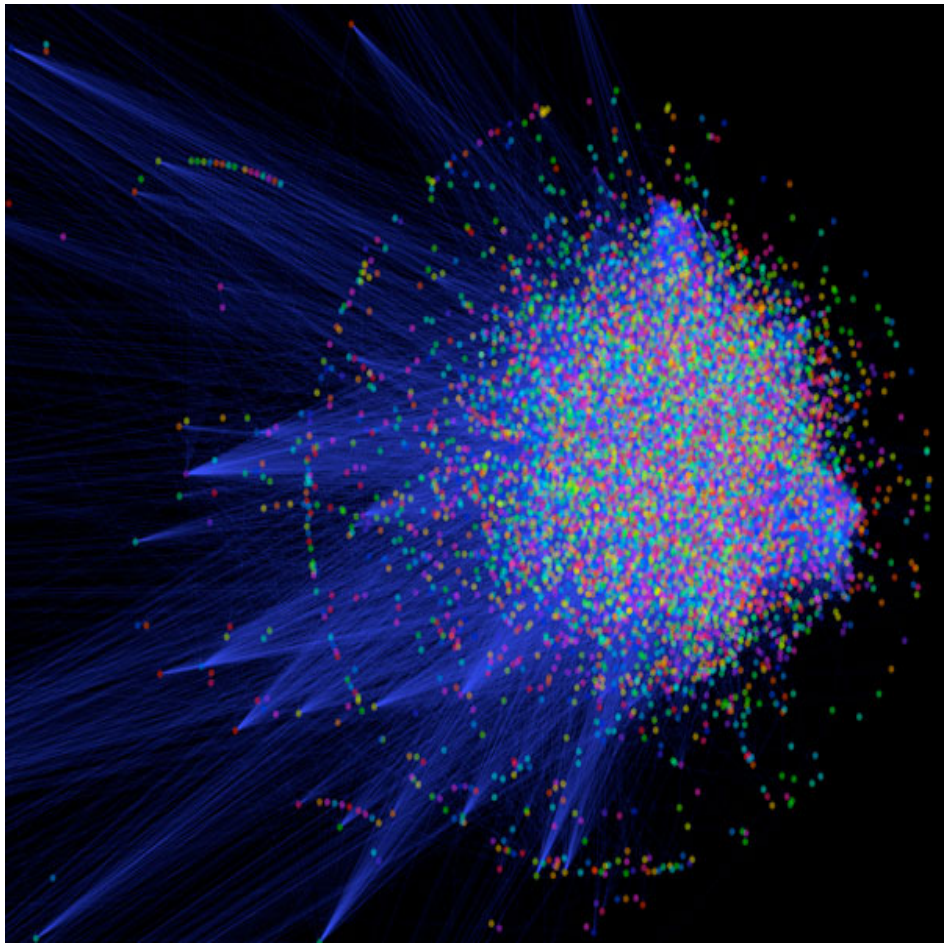
What are networks good for?

- Networks are a convenient model for collection of interacting objects
- Objects correspond to network nodes, and two objects are linked when they interact
- Network statistics may contain insights on fundamental properties of the system
- Network structure can help predict the behaviour of the system
- Networks make sense only when interactions are sparse

Protein-Protein Interaction (PPI) Networks

- Many cellular functions are carried out by complexes of proteins
- Two proteins are linked if they form a complex
- This construction yields a large undirected network (interactome)
- Important: the network is a model, it does not exist. Two proteins being linked does not necessarily mean they will always interact (why?)

Example PPI network

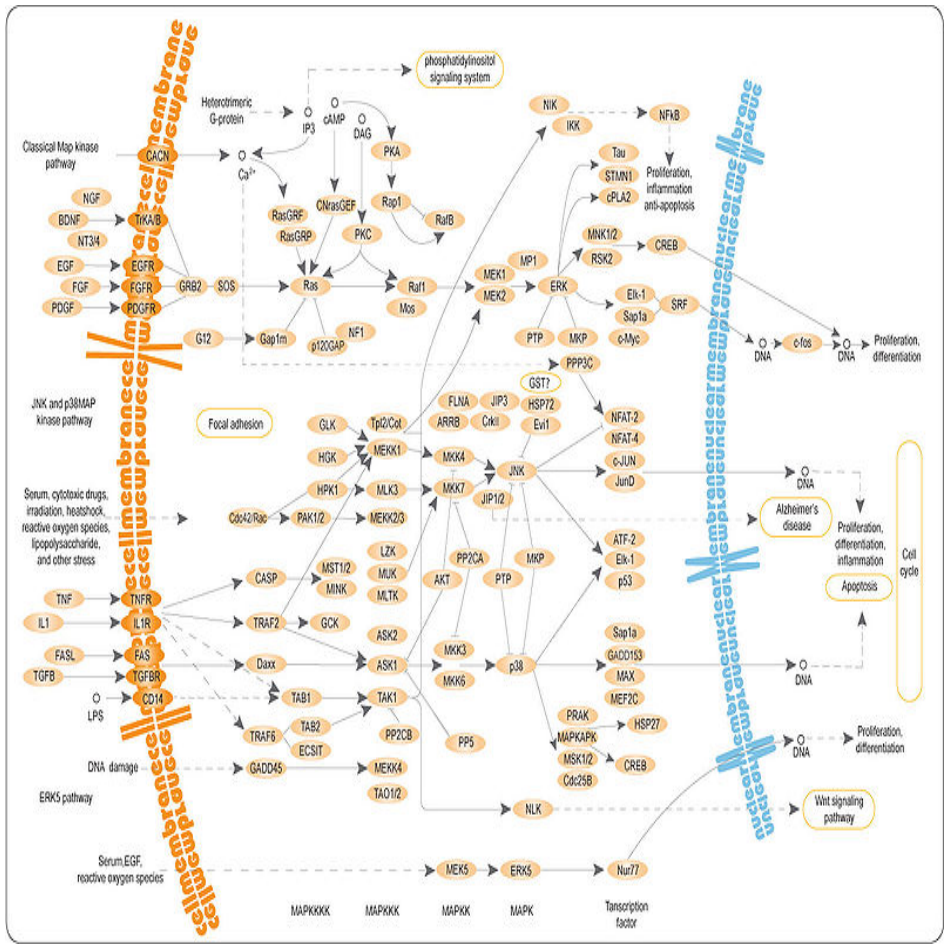


The human interactome.

Signalling networks

- Many proteins exist in different states: e.g. dimer/ monomer, phosphorylated, nitrosylated...
- The change of state is often achieved through the action of another protein: e.g. kinases phosphorylate other proteins
- This produces a sequence of modification events that carry information from a receptor to the nucleus (signalling)
- If protein A modifies protein B we draw a directed edge from A to B
- This yields a signalling network, typically a small-ish directed network

Example Signalling network



The MAPK pathway.

Transcription networks

- mRNA transcription is modulated by transcription factor proteins (TFs)
- Each TF binds specifically to a number of target genes
- We build a directed network by linking each TF with its targets; this yields a bipartite network
- Some target genes do themselves code for TFs; this makes a transcription networt a normal directed network (not bipartite)

Metabolic networks

- Metabolism consists of a huge number of chemical reactions: virtually none of these happens in the absence of catalysts (enzymes)
- Metabolite-centric view (dominant): metabolites are nodes; A and B are linked together if A is a substrate and B is a product of a reaction
- Enzyme-centric view: enzymes are nodes, they are linked together if the products of the reaction catalyzed by A are the substrates for B
- Both constructions yield an undirected network (except for irreversible reactions)

Network jargon

- The number of nodes (edges) is denoted by $\mathrm{card}(V)$ ($\mathrm{card}(E)$)
- The (average) connectivity is the number of edges divided by number of nodes

$$\lambda = \frac{\mathrm{card}(E)}{\mathrm{card}(V)}$$

- The connectivity ranges from $\sim \mathrm{card}(V)$ (fully connected) to 0 (fully disconnected) (what's the exact maximum connectivity?)
- The degree $d(v)$ of a node v is the number of edges attached to it (in directed networks, you have in-degree and out-degree, sometimes called fan-in and fan-out)

Global properties

- The *degree distribution* of a network is the empirical distribution of degrees across nodes, i.e.

$$p(d) = \frac{1}{card(V)} \#\{v \in V : d(v) = d\}.$$

- It summarises global properties of the network
- If the degree distribution is a power law, $p(d) \propto d^{-\gamma}$ we have a *scale-free network* (also called small world)
- Scale free networks have many hubs and have the property that the path between any two nodes is short. They have been proposed as good models of metabolism (skepticism now).

Random networks and network motifs

- When faced with a large network like *E. coli*'s transcription regulatory network, we would like to see if there are some subnetworks which are particularly frequent
- These subnetworks are called *network motifs* and may have some specific biological function
- In order to assess whether something is over-represented, we need to have a *control* experiment
- The control is given by a *random network*
- We will see a simple algorithm for generating random networks

Erdös-Renyi random networks

- Generative algorithm for random networks with fixed average connectivity dating from the '50s
- For each ordered pair of nodes x_i, x_j, generate a random number $z \in [0, 1]$ from the uniform distribution
- Note, x_i and x_j can be the same node, *i.e.* we allow self-regulation
- If

$$z < \frac{\lambda}{\mathrm{card}(V)}$$

  place a directed edge between x_i and x_j
- Repeat through all nodes
- The result is a random network with (approximately) average connectivity λ

Comparing random and real networks

- We are interested in how many times a specific type of subnetwork we may find in a network of size $\mathrm{card}(V) = N$
- *Exercise*, how many times do we find autoregulation in a random network?
- Each x_i can regulate itself with probability

$$p = \frac{\lambda}{N}$$

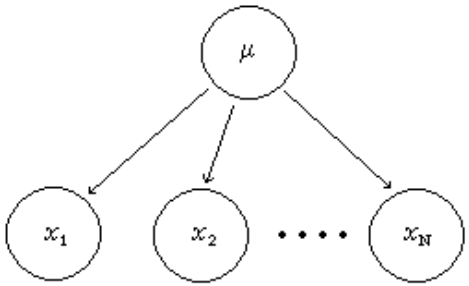
- On average, we would expect Np self-regulating nodes (assuming independence) with a standard deviation of $\sqrt{Np}$
- For *E.coli*, this is 1.1 and 1.2
- There are 40 self regulating nodes in *E.coli*

Significant motifs

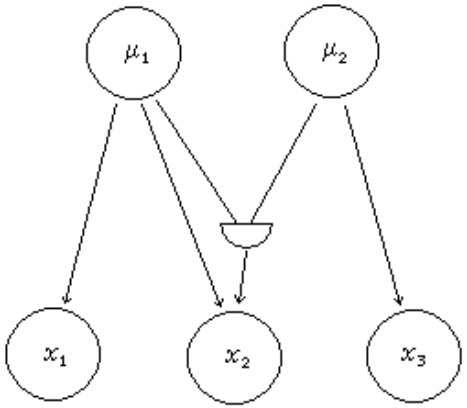
- We have shown that autoregulation is a significant network motif
- Other important motifs are made of three nodes, feed-forward loops
- These have the property of acting as filters for transient signals
- Also important are single-input motifs (one TF regulating many genes) and dense overlapping regulons (group of genes sharing a small number of regulators

Network motifs

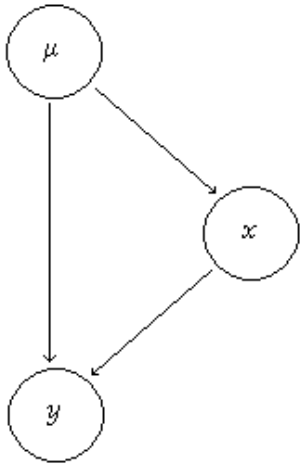
SIM



DOR



FFL



Nodes measurements

- We frequently can measure the quantities associated with the nodes, e.g. expression levels of genes
- Network structure encodes interactions
- Statistically, interactions are modelled as *dependencies* among variables
- Networks of random variables are usually called *graphical models*
- We will first introduce a cheap and cheerful way of constructing networks
- We will explore the case of *Gaussian graphical models* (GGMs), networks of Gaussian variables

Correlation networks

- We are faced with a high-dimensional data set with few observations (e.g. microarrays) with no specific hypotheses or background knowledge
- In these cases, one only needs a fast, scalable algorithm and can compromise on accuracy
- Efficiency is a pre-requisite; in practice for transcriptomics this means $O(n^2)$ where n is the number of genes
- Intuitively, we will connect every pair of nodes that pass a certain similarity threshold
- The results can then be used for data visualisation and exploration/ hypothesis generation

Pearson correlation

- A frequently used measure of similarity is *Pearsons correlation*
- The Pearson correlation coefficient between two vectors **x** and **y** is given by

$$r = \frac{(\mathbf{x} - \mu_x) \cdot (\mathbf{y} - \mu_y)}{\|\mathbf{x} - \mu_x\| \|\mathbf{y} - \mu_x\|}$$

  where μ_x is the mean of the vector **x**
- Exercise: in which sense is Pearson correlation a correlation? (hint: think of the vectors as realisations of a zero mean random variable)

Conditionals of multivariate Gaussians

- Let $x_1, \ldots, x_N$ be jointly zero-mean Gaussian, and denote $C = \Sigma^{-}1$ the inverse of the covariance matrix
- The *conditional distribution* for x_i given *all the other variables* is again Gaussian with variance C_{ii}^{-1} and mean

$$\mu = -C_{ii}^{-1} \sum_{j \neq i} C_{ij} x_j.$$

- Proof done on the board or as exercise (write $x_i = \mathcal{N}(\mu, \sigma^2)$ and compare this pdf with the pdf of the joint)

Conditional independence

- Two variables x_i and x_j are *conditionally independent* given x_k if
$$P\left(x_i|x_j, x_k\right) = P\left(x_i|x_k\right).$$
- If $x_1, \ldots, x_N$ are jointly Gaussian, then x_i and x_j are conditionally independent given all the others if and only if $C_{ij} = 0$
- *Important remark*: in a multivariate Gaussian (and only in a Gaussian), the covariance matrix encodes the dependence (correlation) between the variables, while the *inverse covariance* encodes the conditional dependencies

Networks and Inverse covariances

- We can encode a network by exploiting conditional independence
- We declare that two variables x_i and x_j are conditionally independent given all the others if and only if they are not directly linked in the network
- The network structure is therefore encoded in the pattern of zeros of the inverse covariance matrix
- Inferring the network in the Gaussian case is the same as learning the (inverse) covariance
- Exercise: what is the pattern of zeros in the inverse covariance of the undirected graph in slide 3?

Cytoscape: next week

- At a more basic level, one may want to visualise on the network the expression patterns found in an experiment
- The dominant tool is Cytoscape `http://www.cytoscape.org/`, a widely used open-source tool
- Does not model the data, only offers visualisation
- Large number of plug-ins which allow for basic data mining