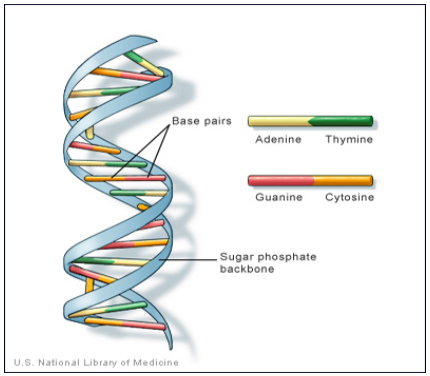
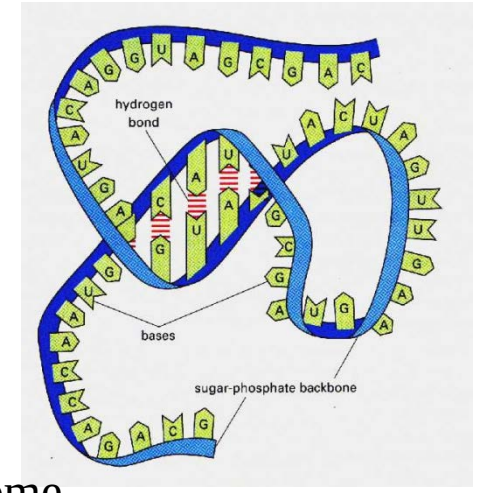
Visualization in bioinformatics and systems biology

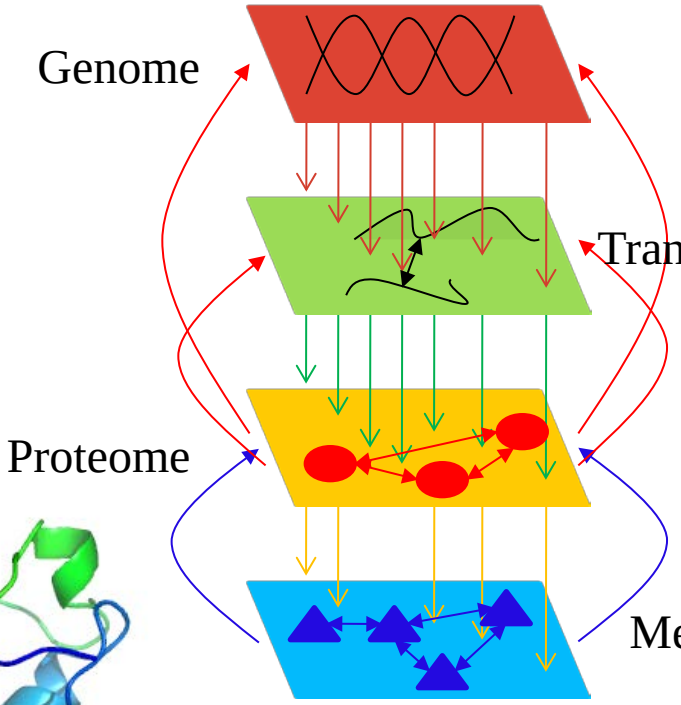
Torgeir R. Hvidsten

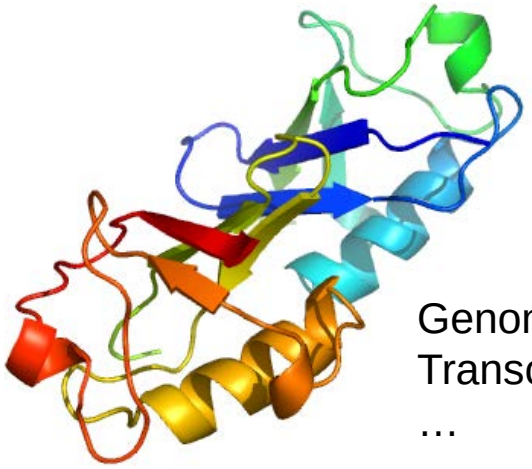
Slides: http://www.trhvidsten.com/Teaching.html

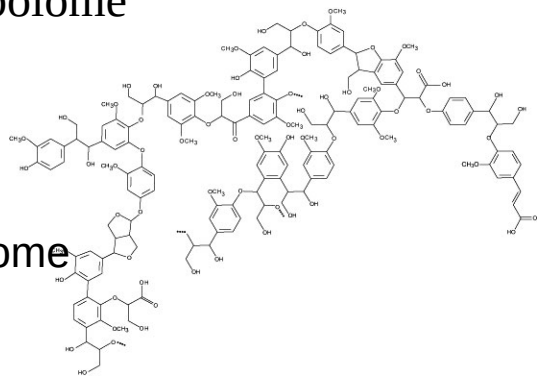
Omics data











Genomics – study the genome
Transcriptomics – study the transcriptome
…

omics data – observations from the *omes*

Omics data is multidimensional

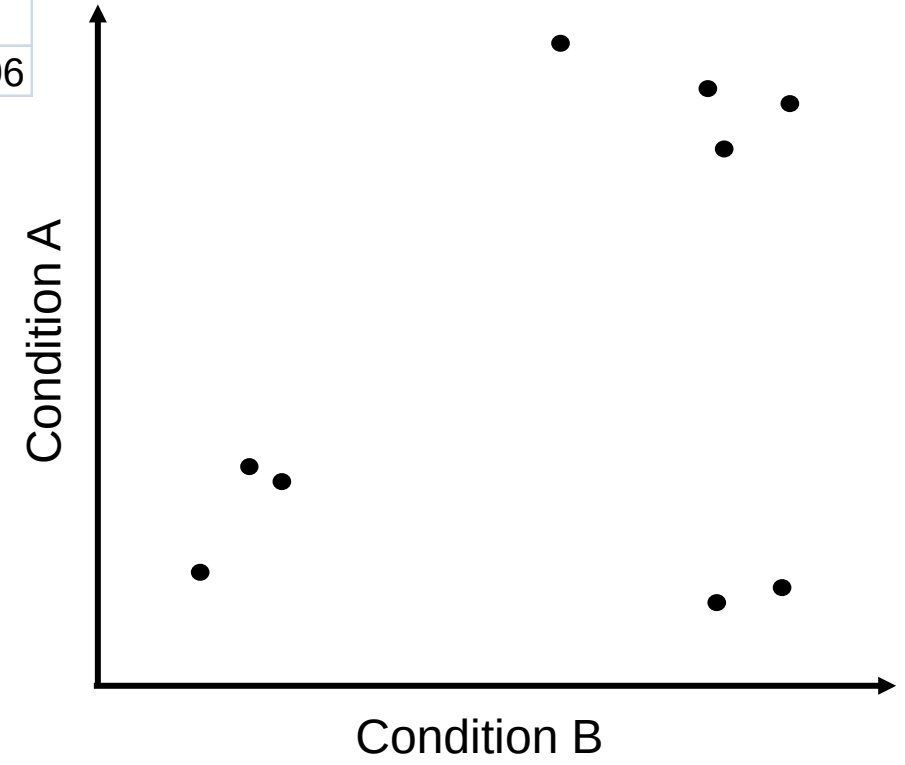
Rows: observations
Columns: features

Conditions/tissues/time (samples)

Genes/metabolites/proteins

0.54	0.53	0.16	0.14	0.20	-0.34	-0.38	-0.36
-0.47	-3.32	-0.81	0.11	-0.60	-1.36	-1.03	-1.84
0.66	0.07	0.20	0.29	-0.89	-0.45	-0.29	-0.29
0.14	-0.04	0.00	-0.15	-0.58	-0.30	-0.18	-0.38
-0.04	0.00	-0.23	-0.25	-0.47	-0.60	-0.56	-1.09
0.28	0.37	0.11	-0.17	-0.18	-0.60	-0.23	-0.58
0.54	0.53	0.16	0.14	0.20	-0.34	-0.38	-0.36
0.20	0.14	0.00	0.11	-0.34	-0.03	0.04	-0.76
0.40	0.43	0.18	0.00	-0.14	0.29	0.07	-0.79
0.01	0.46	0.28	-0.34	-0.23	-0.36	-0.45	-0.64
…	…	…	…	…	…	…	…
-0.23	0.04	0.00	-0.30	-0.29	-0.45	-0.97	-2.06

Time series versus
Feature space

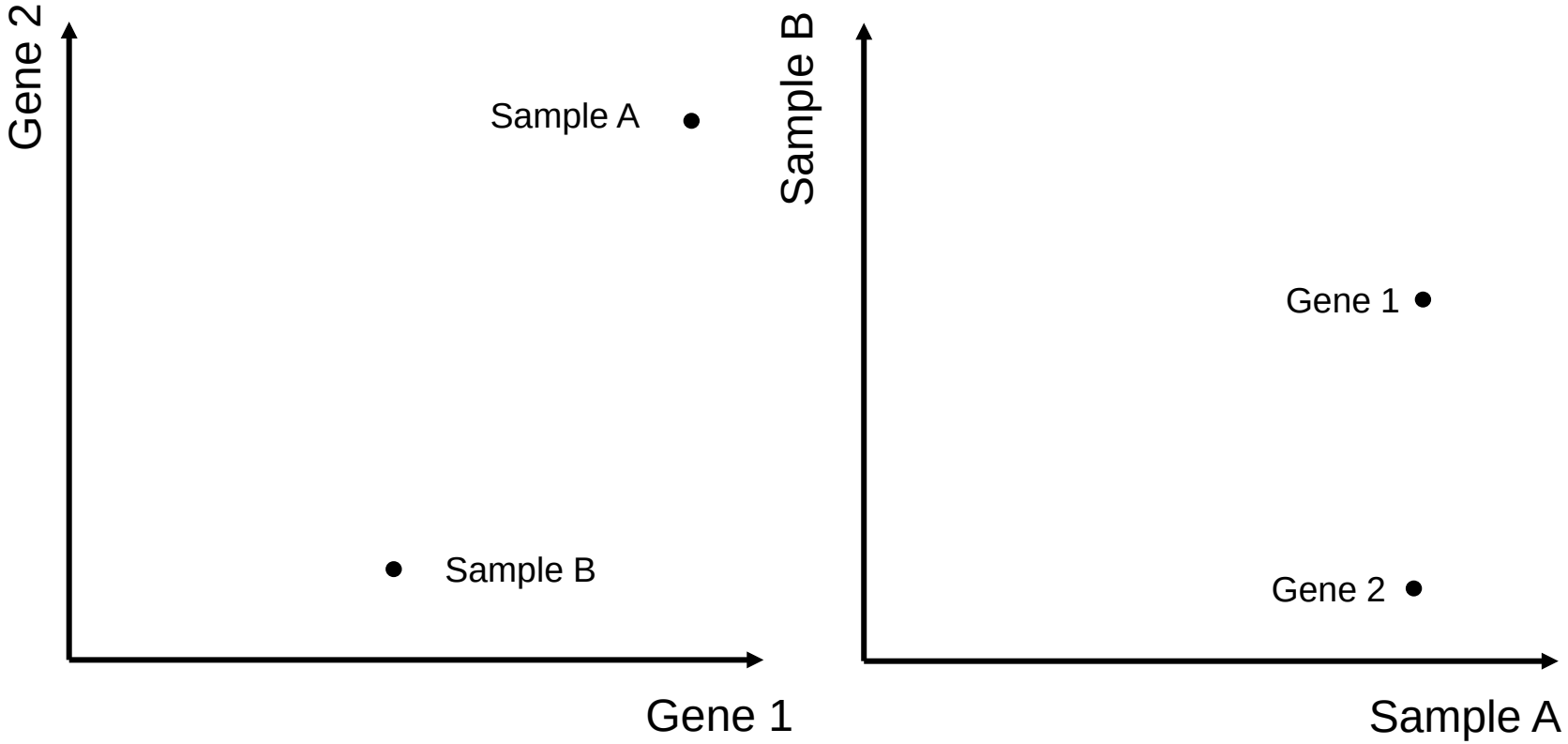


Genes or samples are features

Depends on the goal of the analysis

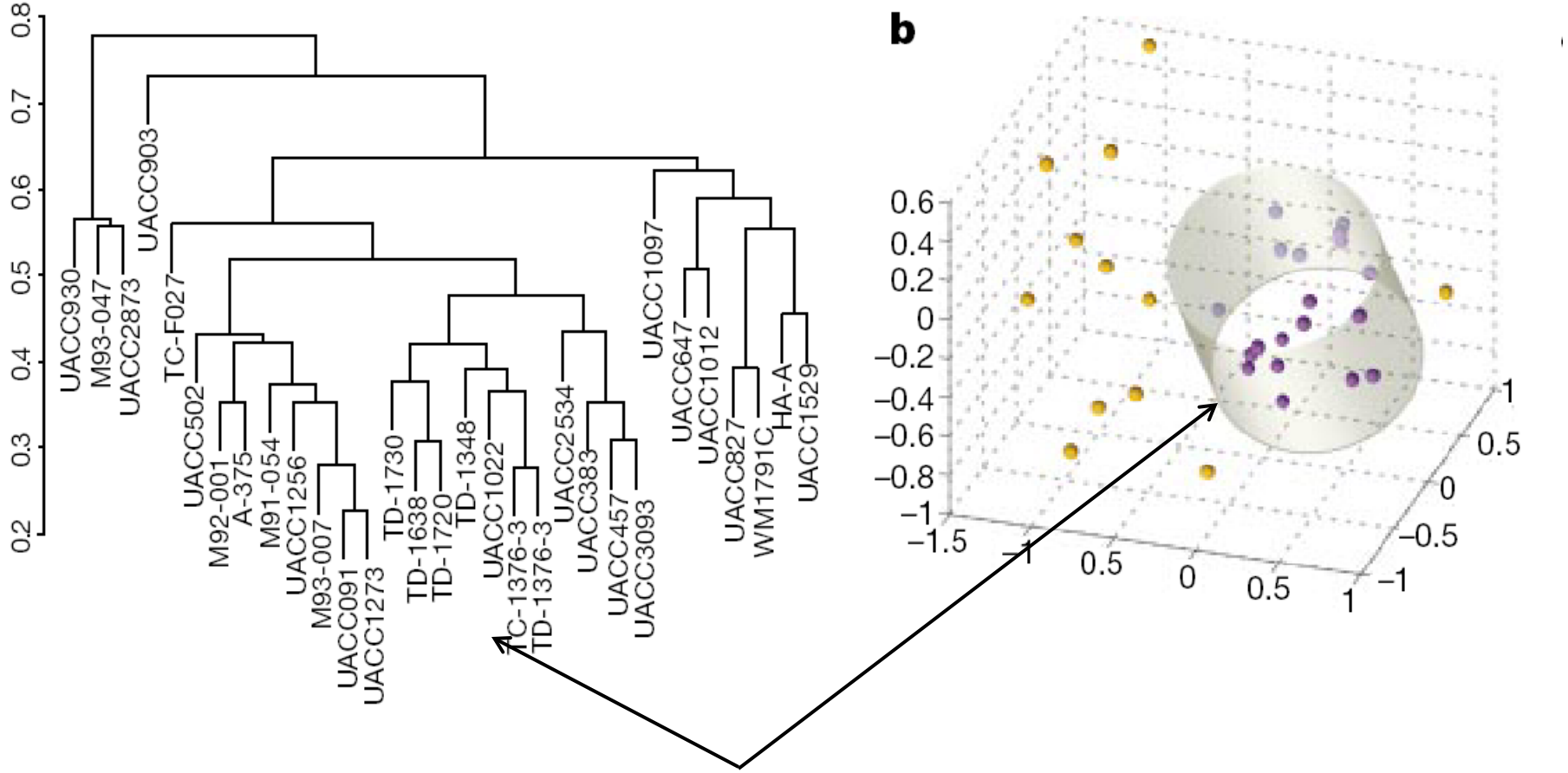
Samples are points in a space spanned by the genes

Genes are points in a space spanned by the samples



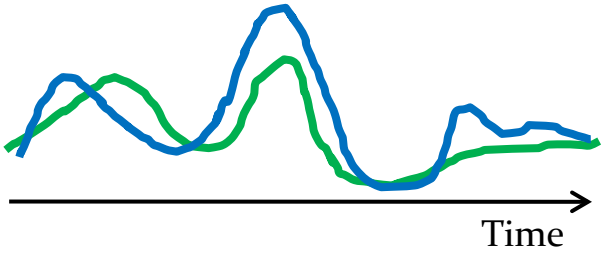
Looking into feature space with more than 3D:

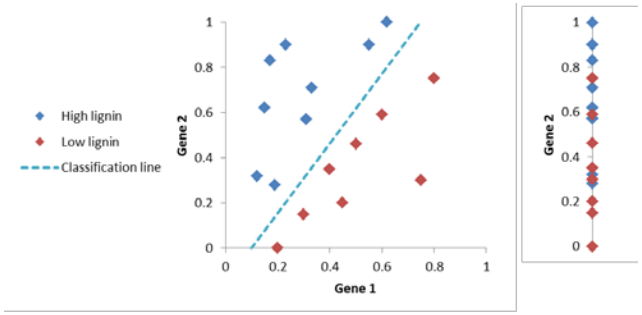
Hierarchical clustering and principle component analysis (PCA)

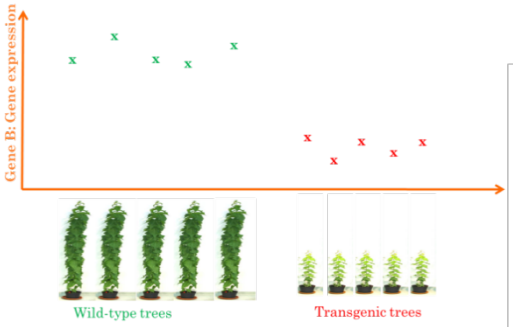


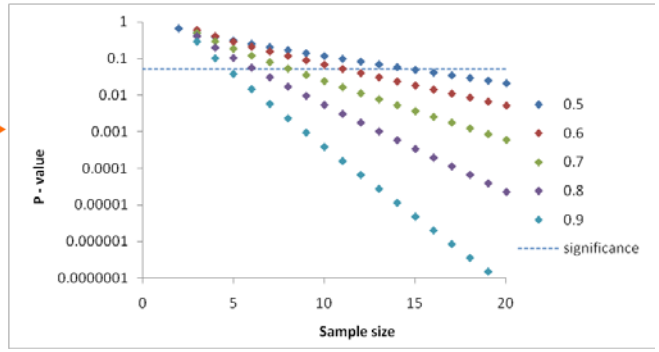
19 melanomas of all 31 cutaneous melanoma samples
(Bitter et al. *Nature.* 406: 536, 2000)

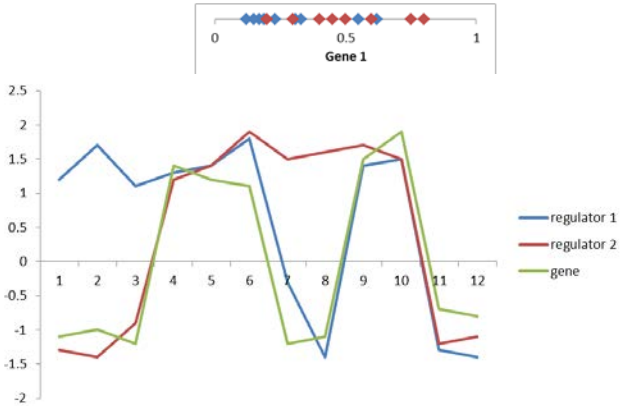
Methods











Clustering
Co-expression networks

Interactions
(Non-)linear models

Differential expression

Model complexity

Emergent properties

Expression similarity

Differential expression

Number of observations

Predicting "causality" from expression data: Analogous to establishing whether you are being followed by the car behind you

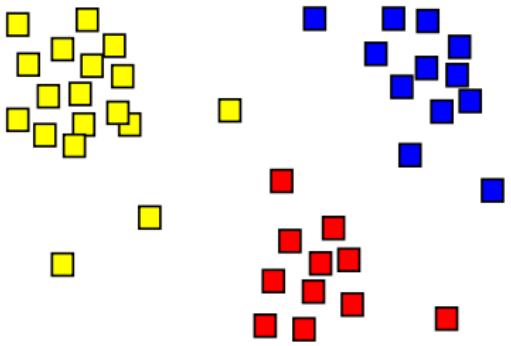
Using array data: with a fuggy rear-view mirror

Using RNA-Seq: with a clear rear-view mirror

Machine learning

- Supervised learning; used to learn a model from a set of examples given predefined classes (training examples)
  - Example = observation + class
  - Model: e.g. **IF** gene 1420 > 5.3 **THEN** mutant

- Unsupervised learning (clustering, class discovery); used to discover natural groups of observations



2D data: two features X and Y

Three groups discovered:

Yellow: low X, high Y
Red: high X, low Y
Blue: high X, high Y

Supervised vs unsupervised
=
With vs without classes

M < 100

Gene/Expr	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	…	EM
G1	-0.47	-3.32	-0.81	0.11	-0.60	-1.36	-1.03	-1.84	-1.00	-0.60	…	-0.94
G2	0.66	0.07	0.20	0.29	-0.89	-0.45	-0.29	-0.29	-0.15	-0.45	…	-0.42
G3	0.14	-0.04	0.00	-0.15	-0.58	-0.30	-0.18	-0.38	-0.49	-0.81	…	-1.12
G4	-0.04	0.00	-0.23	-0.25	-0.47	-0.60	-0.56	-1.09	-0.71	-0.76	…	-0.62
G5	0.28	0.37	0.11	-0.17	-0.18	-0.60	-0.23	-0.58	-0.79	-0.29	…	-0.74
G6	0.54	0.53	0.16	0.14	0.20	-0.34	-0.38	-0.36	-0.49	-0.58	…	-1.47
G7	0.20	0.14	0.00	0.11	-0.34	-0.03	0.04	-0.76	-0.81	-1.12	…	-1.36
G8	0.40	0.43	0.18	0.00	-0.14	0.29	0.07	-0.79	-0.81	-0.92	…	-1.22
G9	0.01	0.46	0.28	-0.34	-0.23	-0.36	-0.45	-0.64	-0.79	-1.22	…	-1.09
…	…	…	…	…	…	…	…	…	…	…	…	…
GN	-0.23	0.04	0.00	-0.30	-0.29	-0.45	-0.97	-2.06	-0.89	-1.22	…	-0.97

Cell growth

Transcripton

WT

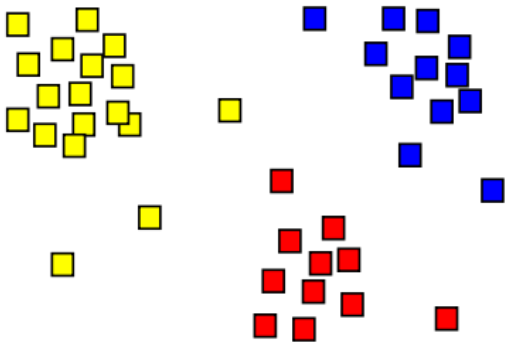
Mutant

N > 10000

Machine learning

- Supervised learning; used to learn a model from a set of examples given predefined classes (training examples)
  - Example = observation + class
  - Model: e.g. **IF** gene 1420 > 5.3 **THEN** mutant

- **Unsupervised learning** (clustering, class discovery); used to discover natural groups of observations

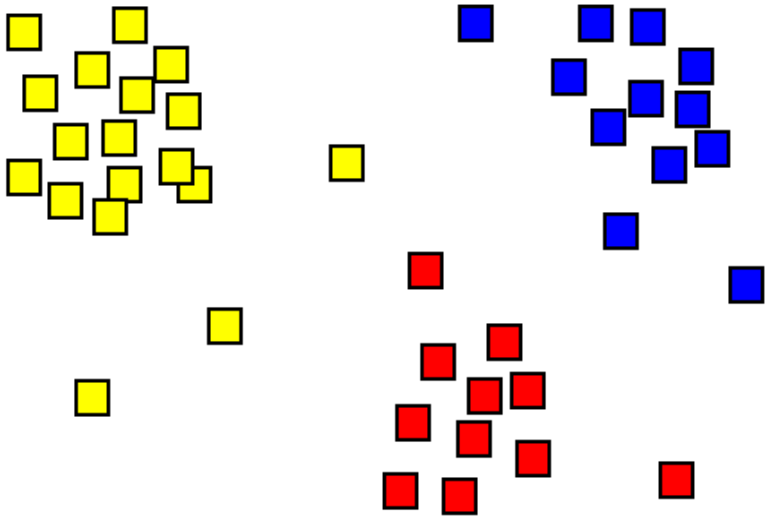


2D data: two features X and Y

Three groups discovered:

Yellow: low X, high Y
Red: high X, low Y
Blue: high X, high Y

Clustering



Clustering requires:

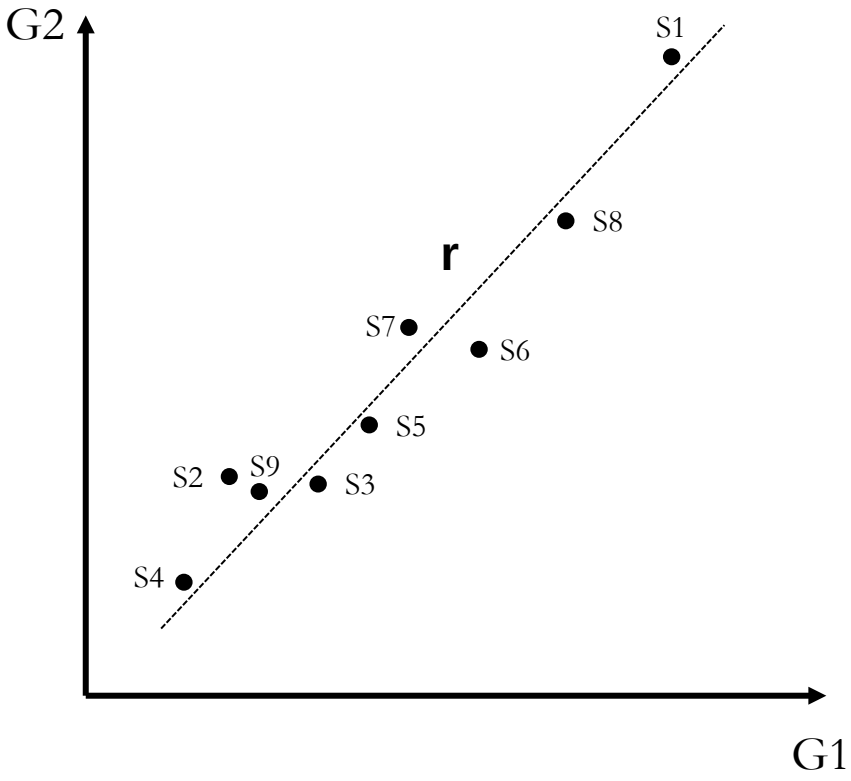
- A similarity measure
- An algorithm that finds similar observations in data

NP-hard: The number of ways to divide n items into k clusters: $k^n/k!$

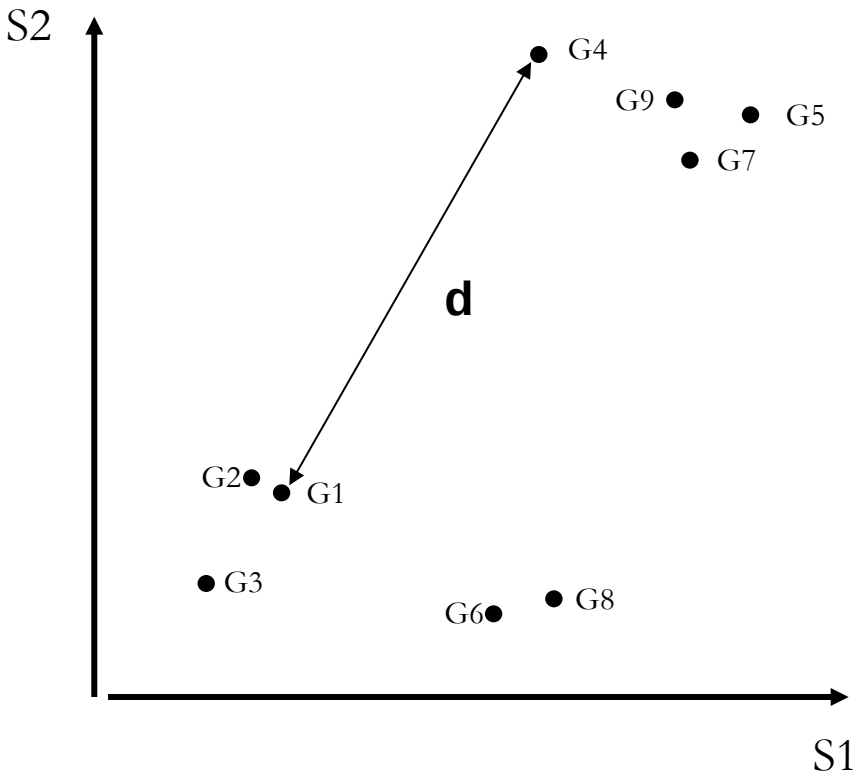
Example: $10^{500}/10! = 2.756 \times 10^{493}$

Similarity measures

Pearson correlation
(not feature space!)



Eucledian distance
(in feature space)

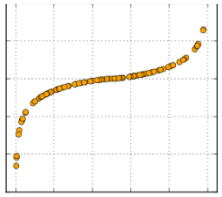


More similarity measures

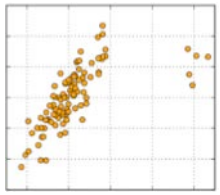
- **Similarity measures**: high values mean similar (e.g. Pearson correlation)
- **Dissimilarity measures/distances**: low values mean similar (e.g. Euclidian dist.)

- **Pearson correlation:** measure linear dependency

- **Spearman correlation**: measure monotonic trends, more robust to outliers

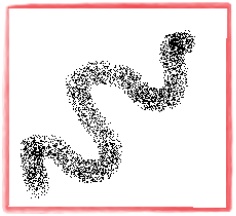


Spearman: 1.00
Pearson: 0.88



Spearman: 0.84
Pearson: 0.67

- **Mutual information (MI)**: measure non-monotonic and other more complex relationships



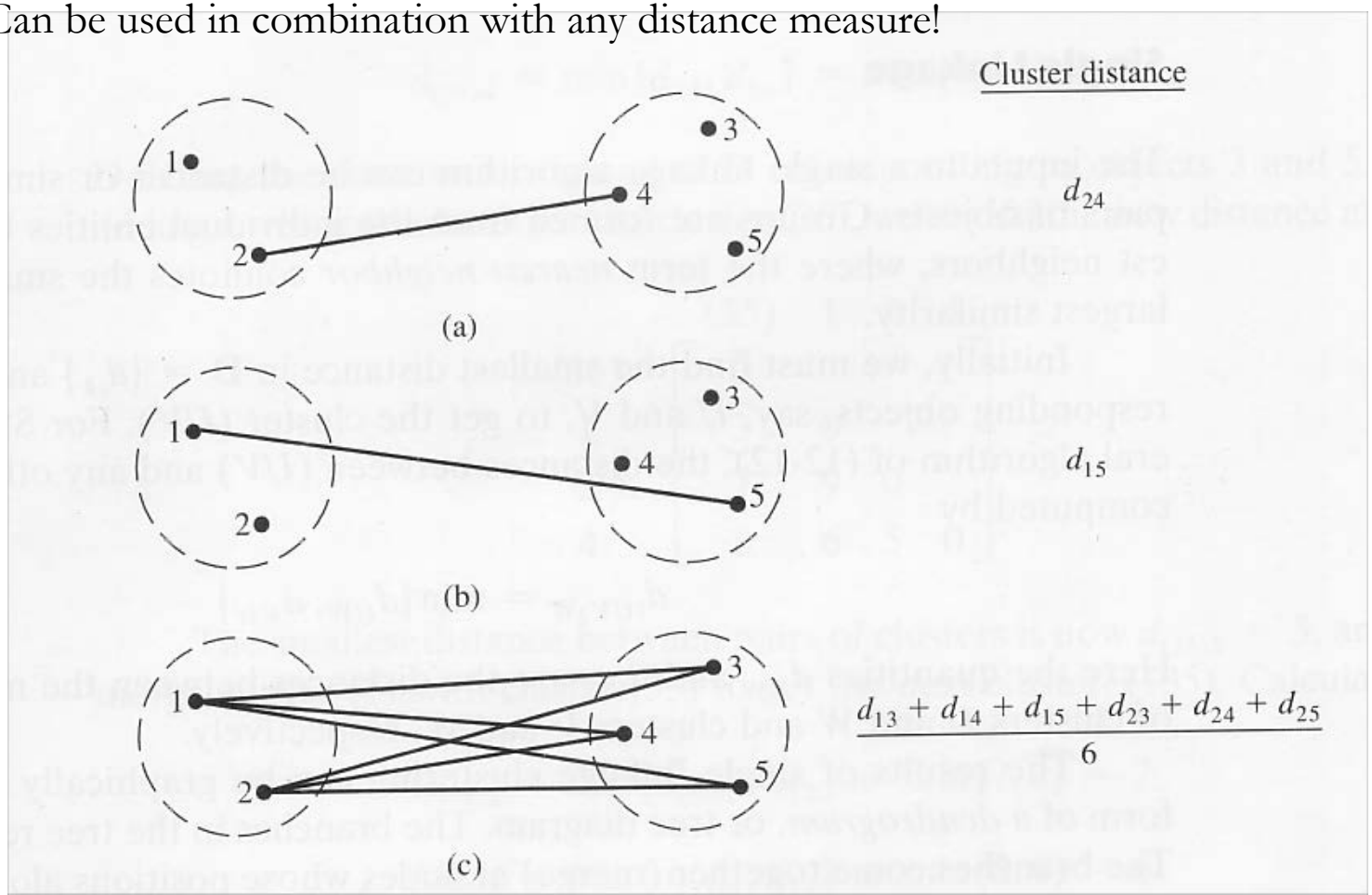
Hierarchical clustering

- Start with each observation as a separate cluster
- Compute the distance between all pairs of clusters – distance matrix
- Repeat until there is only one cluster:
  - Merge the two clusters with the shortest distance
  - Update the distance matrix with the new cluster

Hierarchical clustering
Linkage

Inter-cluster similarity measures: (a) single linkage, (b) complete linkage and (c) average linkage

Can be used in combination with any distance measure!



Hierarchical clustering
Ward's method

- Minimize the total within-cluster variance:
  - $d = \sum_{k=1}^{n} \sum_{(i,j) \epsilon C_k} d(\boldsymbol{x}_i, \boldsymbol{x}_j)$
  - n is the number of clusters
  - C_k is the set of the indices of all observation-pairs in cluster k
  - $\boldsymbol{x}_i$ is observation i
- Merge the two clusters that results in the smallest total within-cluster variance of all cluster pairs

Example of hierarchical clustering: languages of Europe

TABLE 12.3 NUMERALS IN 11 LANGUAGES

English (E)	Norwegian (N)	Danish (Da)	Dutch (Du)	German (G)	French (Fr)	Spanish (Sp)	Italian (I)	Polish (P)	Hungarian (H)	Finnish (Fi)
one	en	en	een	eins	un	uno	uno	jeden	egy	yksi
two	to	to	twee	zwei	deux	dos	due	dwa	ketto	kaksi
three	tre	tre	drie	drei	trois	tres	tre	trzy	harom	kolme
four	fire	fire	vier	vier	quatre	cuatro	quattro	cztery	negy	neua
five	fem	fem	vijf	funf	cinq	cinco	cinque	piec	ot	viisi
six	seks	seks	zes	sechs	six	seis	sei	szesc	hat	kuusi
seven	sju	syv	zeven	sieben	sept	siete	sette	siedem	het	seitseman
eight	atte	otte	acht	acht	huit	ocho	otto	osiem	nyolc	kahdeksan
nine	ni	ni	negen	neun	neuf	nueve	nove	dziewiec	kilenc	yhdeksan
ten	ti	ti	tien	zehn	dix	diez	dieci	dziesiec	tiz	kymmenen

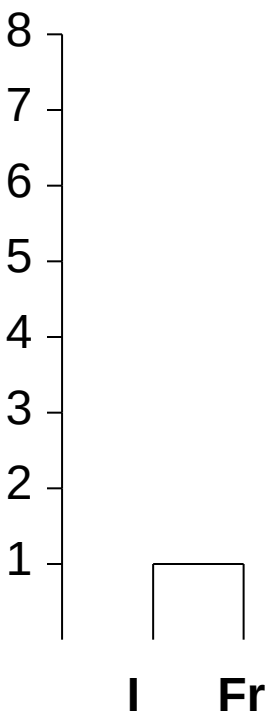
Distance: Frequency of numbers with different first letter e.g.

$d_{EN} = 2$ $d_{EDu} = 7$ $d_{SpI} = 1$

Inter-cluster strategy: SINGEL LINKAGE

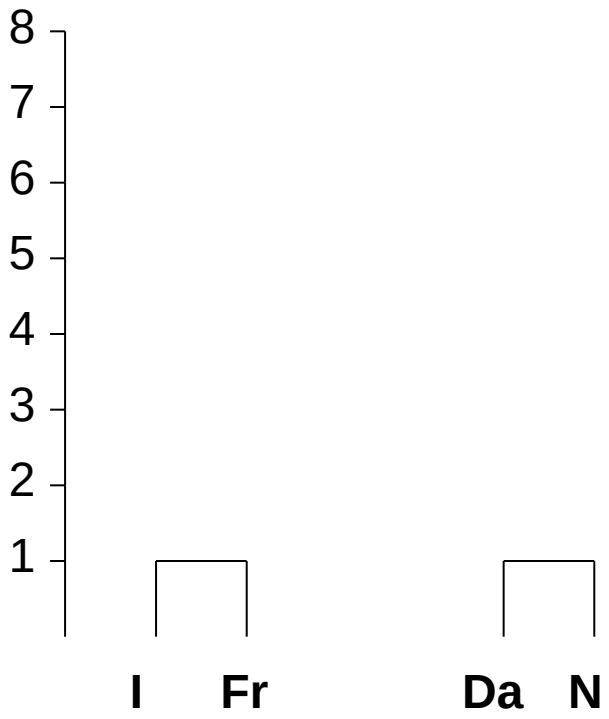
Iteration 1

	E	N	Da	Du	G	Fr	Sp	I	P	H	Fi
E	0										
N	2	0									
Da	2	1	0								
Du	7	5	6	0							
G	6	4	5	5	0						
Fr	6	6	6	9	7	0					
Sp	6	6	5	9	7	2	0				
I	6	6	5	9	7	1	1	0			
P	7	7	6	10	8	5	3	4	0		
H	9	8	8	8	9	10	10	10	10	0	
Fi	9	9	9	9	9	9	9	9	9	8	0



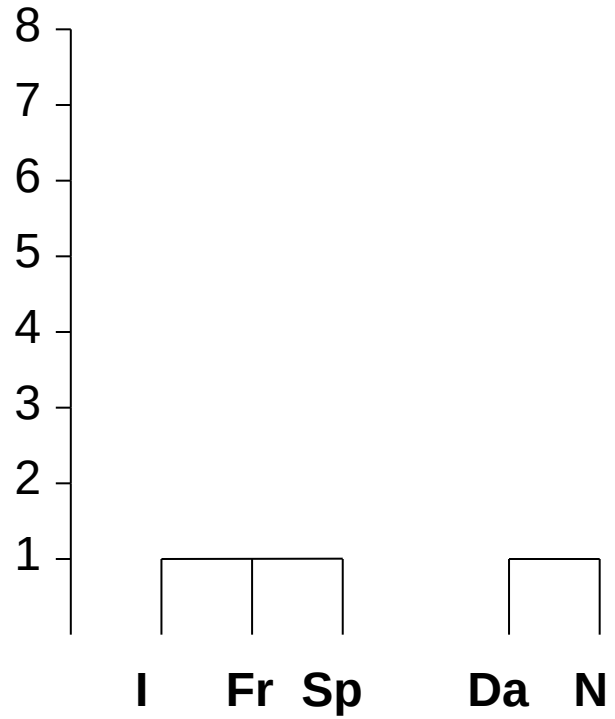
Iteration 2

	I Fr	E	N	Da	Du	G	Sp	P	H	Fi
I Fr	0									
E	6	0								
N	6	2	0							
Da	5	2	1	0						
Du	9	7	5	6	0					
G	7	6	4	5	5	0				
Sp	1	6	6	5	9	7	0			
P	4	7	7	6	10	8	3	0		
H	10	9	8	8	8	9	10	10	0	
Fi	9	9	9	9	9	9	9	9	8	0



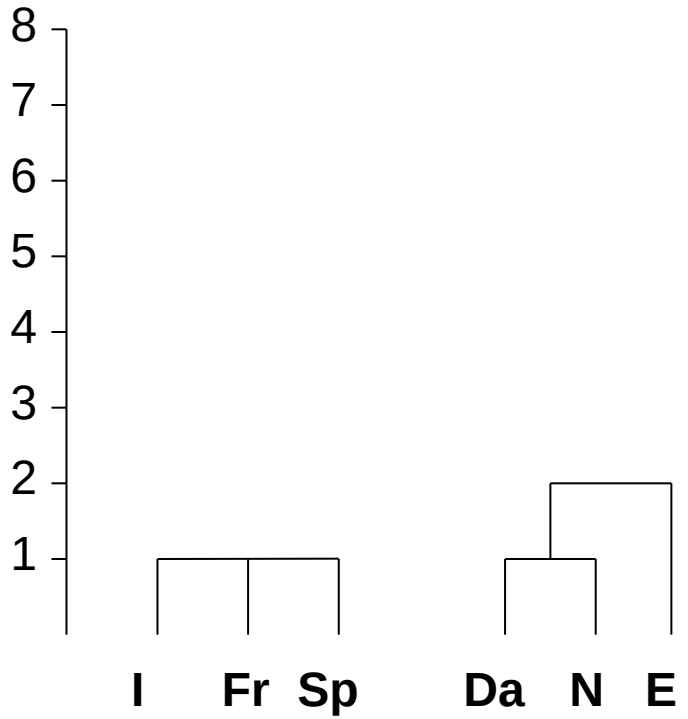
Iteration 3

	Da N	I Fr	E	Du	G	Sp	P	H	Fi
Da N	0								
I Fr	5	0							
E	2	6	0						
Du	5	9	7	0					
G	4	7	6	5	0				
Sp	5	1	6	9	7	0			
P	6	4	7	10	8	3	0		
H	8	10	9	8	9	10	10	0	
Fi	9	9	9	9	9	9	9	8	0



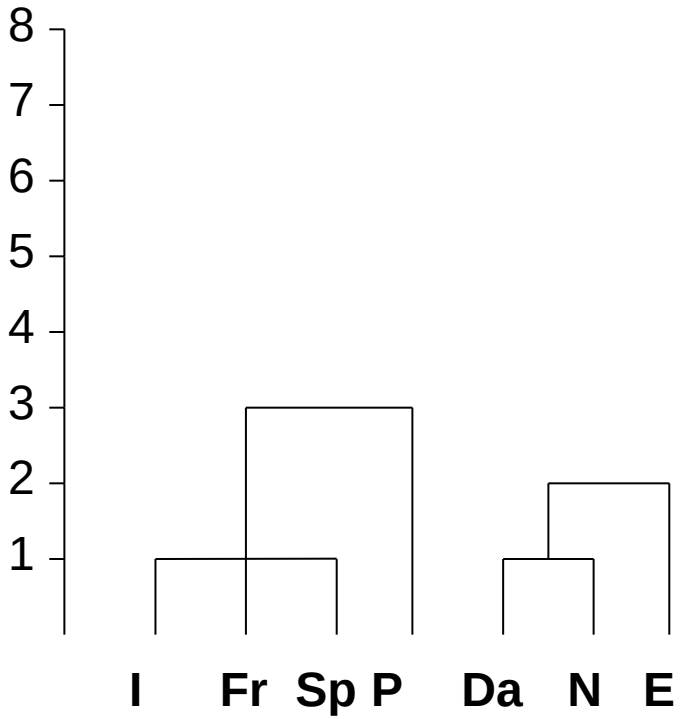
Iteration 4

	Sp I Fr	Da N	E	Du	G	P	H	Fi
Sp I Fr	0							
Da N	5	0						
E	6	2	0					
Du	9	5	7	0				
G	7	4	6	5	0			
P	3	6	7	10	8	0		
H	10	8	9	8	9	10	0	
Fi	9	9	9	9	9	9	8	0



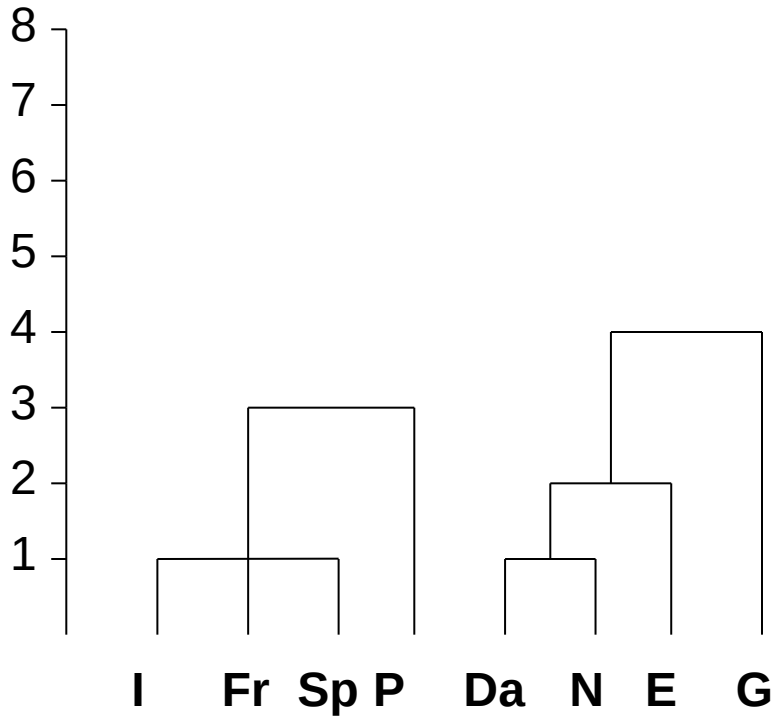
Iteration 5

	E Da N	Sp I Fr	Du	G	P	H	Fi
E Da N	0						
Sp I Fr	5	0					
Du	5	9	0				
G	4	7	5	0			
P	6	3	10	8	0		
H	8	10	8	9	10	0	
Fi	9	9	9	9	9	8	0



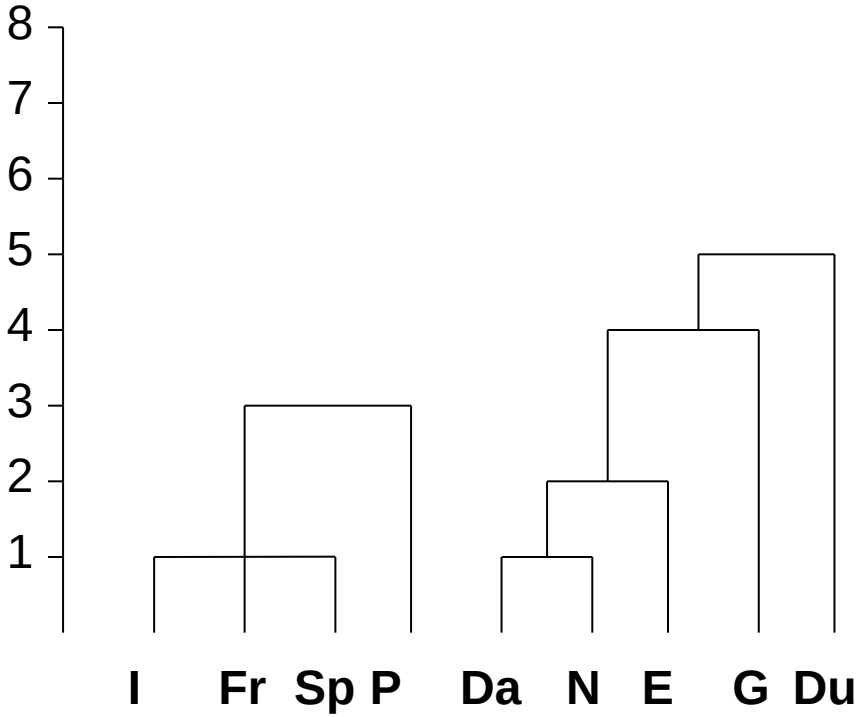
Iteration 6

	P Sp I Fr	E Da N	Du	G	H	Fi
P Sp I Fr	0					
E Da N	5	0				
Du	9	5	0			
G	7	4	5	0		
H	10	8	8	9	0	
Fi	9	9	9	9	8	0



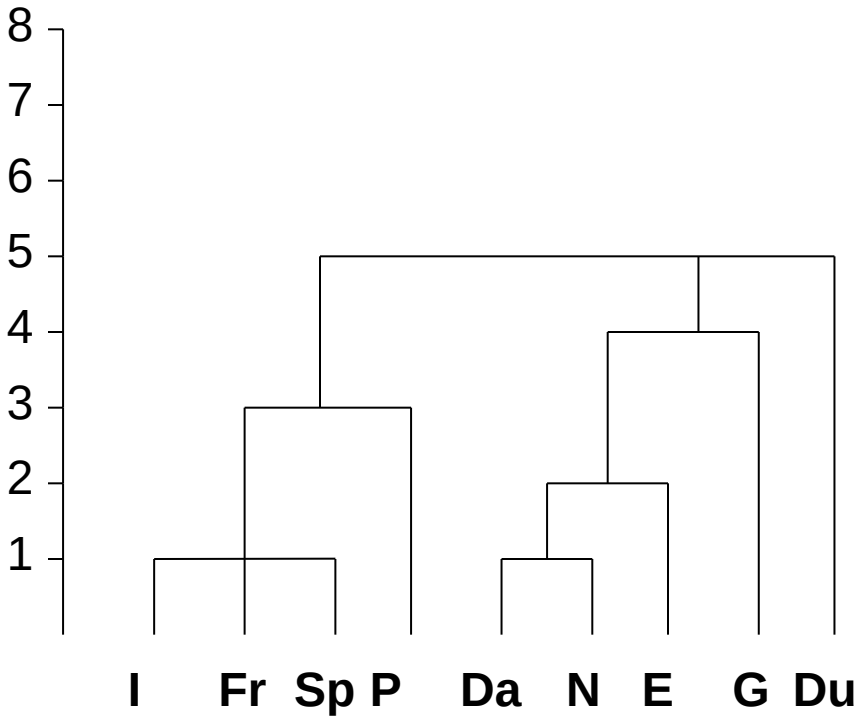
Iteration 7

	G E Da N	P Sp I Fr	Du	H	Fi
G E Da N	0				
P Sp I Fr	5	0			
Du	5	9	0		
H	8	10	8	0	
Fi	9	9	9	8	0



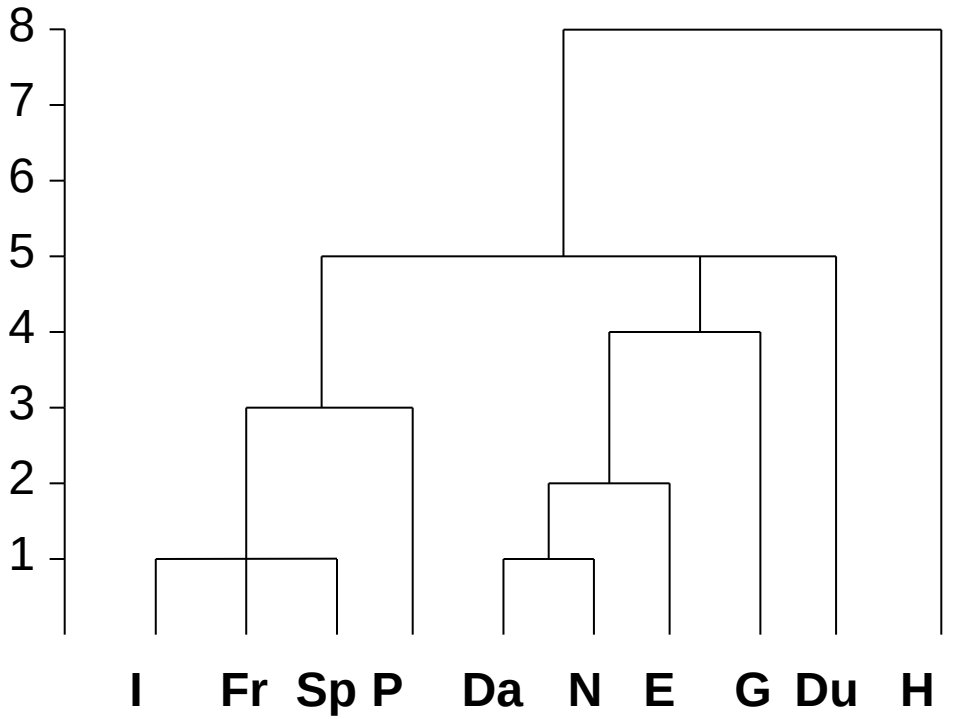
Iteration 8

	Du G E Da N	P Sp I Fr	H	Fi
Du G E Da N	0			
P Sp I Fr	5	0		
H	8	10	0	
Fi	9	9	8	0



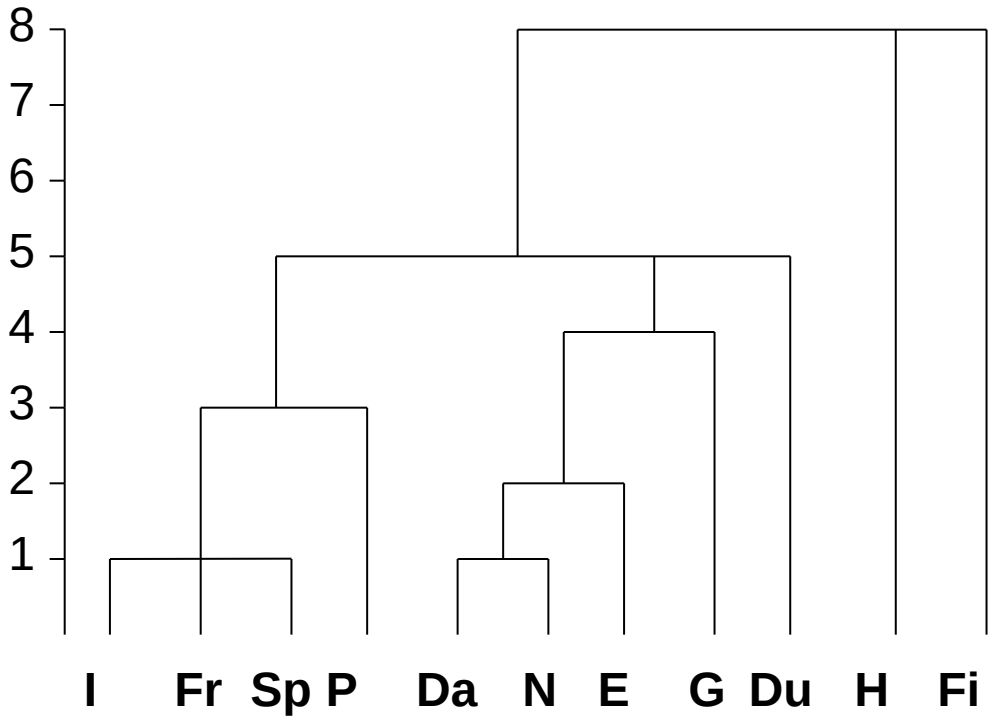
Iteration 9

	P Sp I Fr Du G E Da N	H	Fi
P Sp I Fr Du G E Da N	0		
H	8	0	
Fi	9	8	0



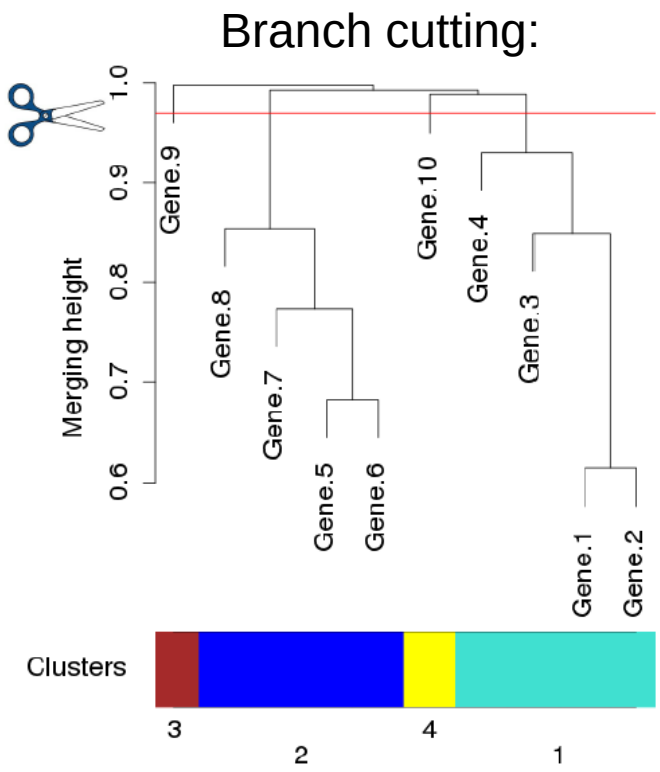
Iteration 10

	Fi H	P Sp I Fr Du G E Da N
Fi H	0	
P Sp I Fr Du G E Da N	8	0



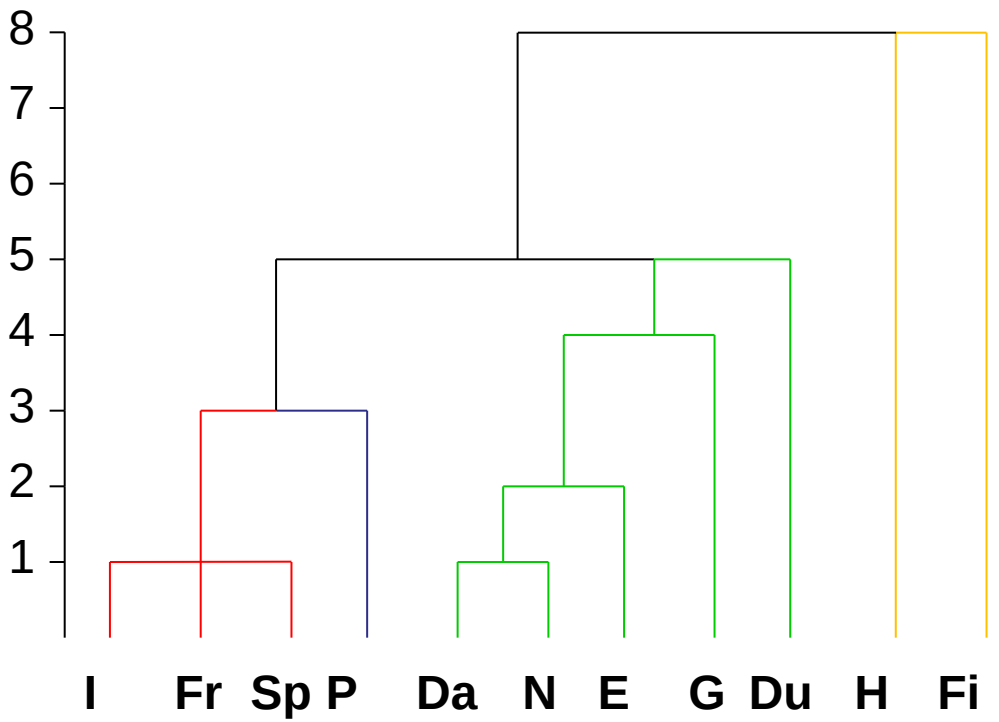
Hierarchical clustering: properties

- A greedy algorithm
- Huge memory requirements: stores the $n \times n$ matrix
- Running time: $O(n^3)$
- Deterministic: produces the same clustering every time
- Nice visualization: dendrogram
- Number of clusters can be selected using the dendrogram; so called branch cutting methods



Evaluation of clusters

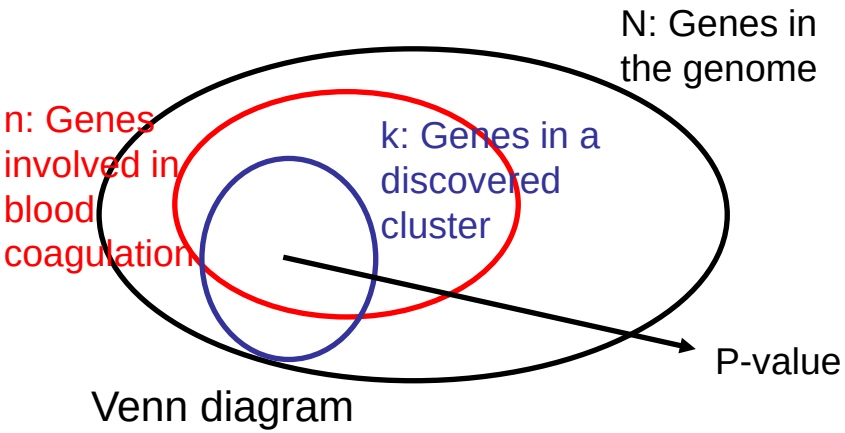
Clusters may be evaluated according to how well they describe existing knowledge



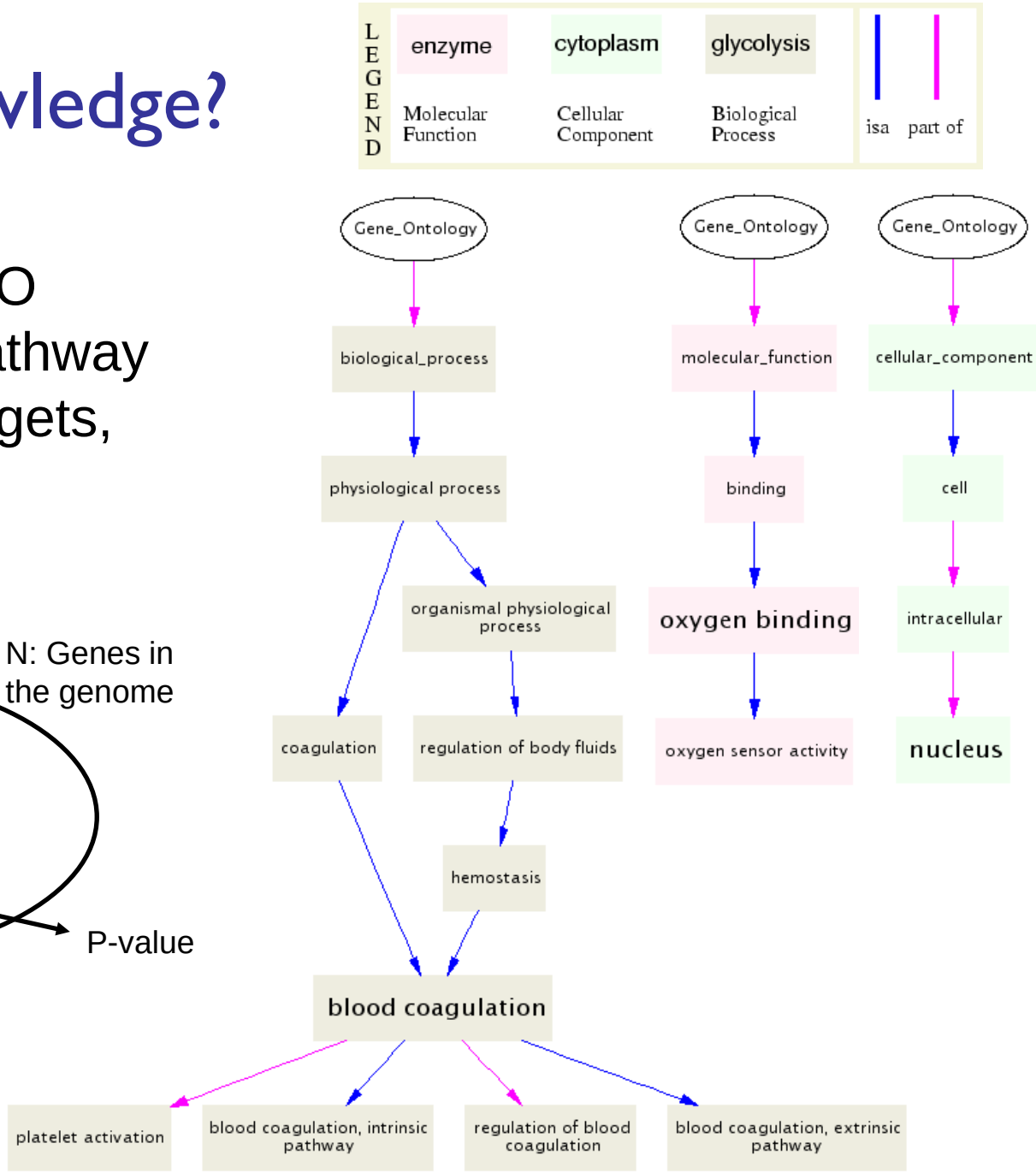
Roman
Slavic
Germanic
Ugro-Finnish

Existing knowledge?

Are the clusters overrepresented for GO annotations, KEGG pathway annotation, miRNA targets, regulatory motifs, … ?



Venn diagram

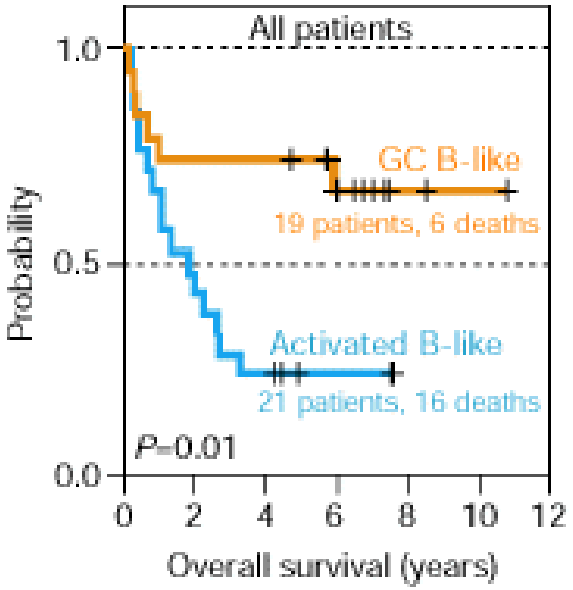


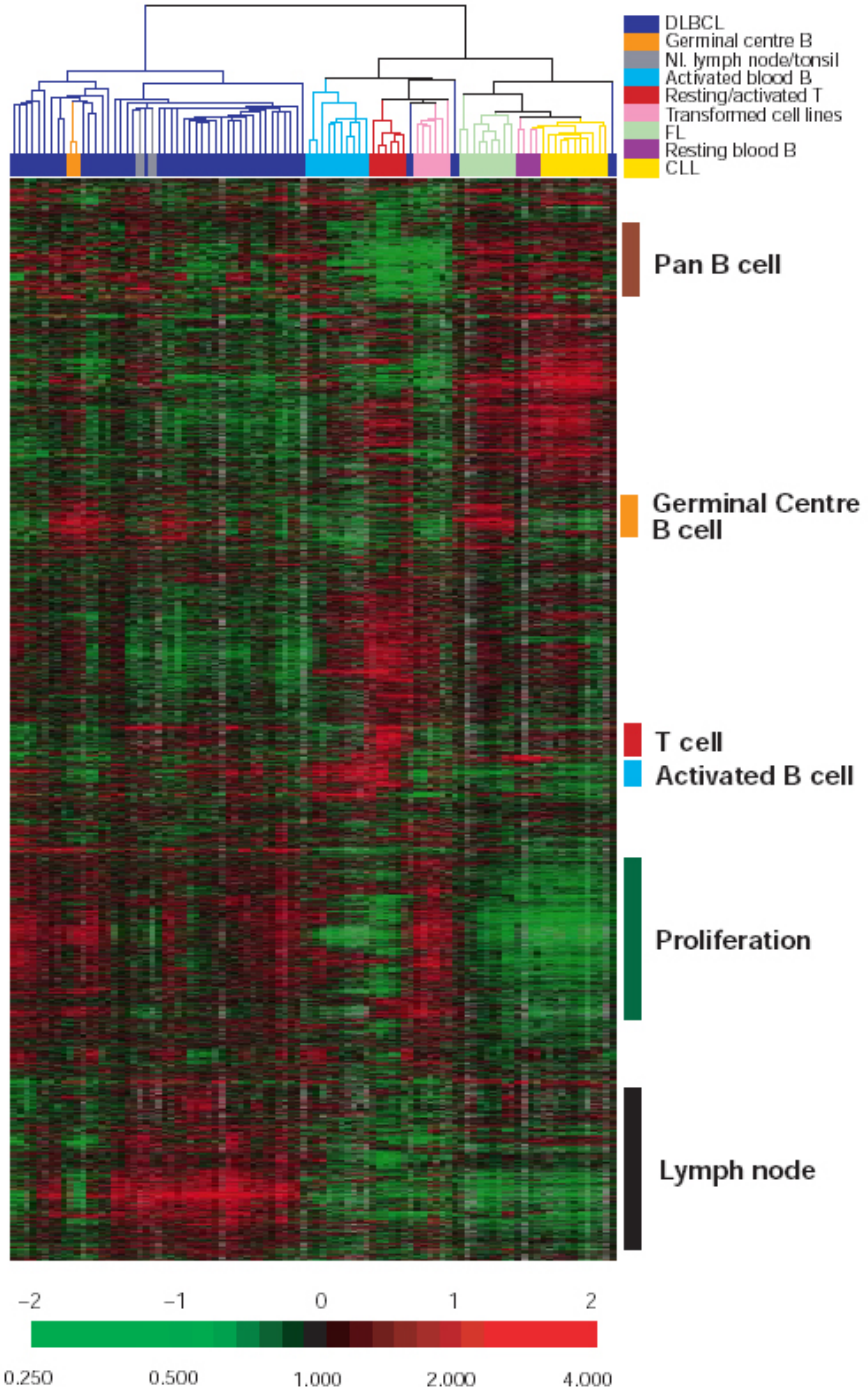
Example: Hierarchical clustering

96 normal and malignant lymphocyte samples

Almost 20 000 cDNA clones

Two sub-clusters of DLBCL were shown to include patients with significantly different expected survival time!





Alizadeh et al., Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling, *Nature*, 403:503-511, 2000.

LETTER

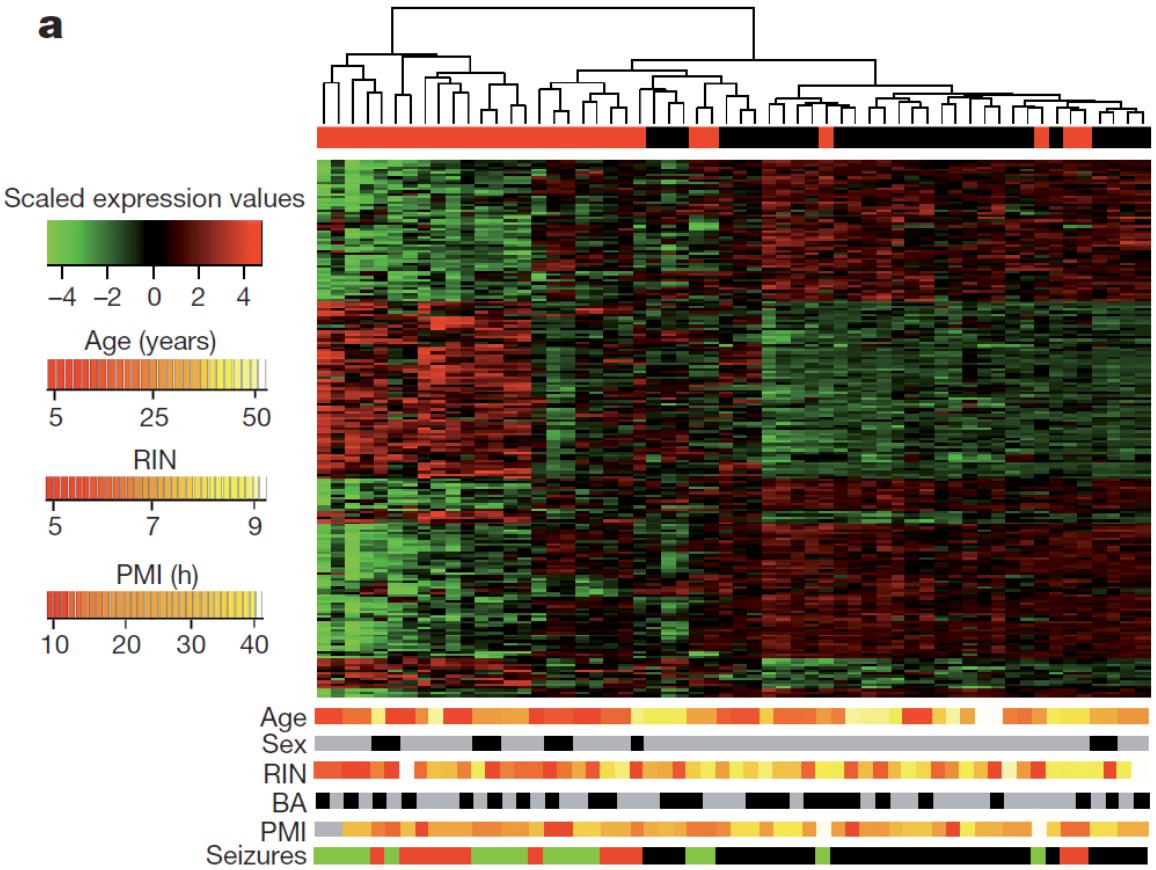
doi:10.1038/nature10110

Transcriptomic analysis of autistic brain reveals convergent molecular pathology

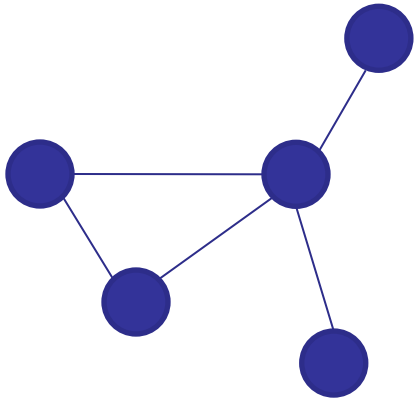
Irina Voineagu[1], Xinchen Wang[2], Patrick Johnston[3], Jennifer K. Lowe[1], Yuan Tian[1], Steve Horvath[4], Jonathan Mill[3], Rita M. Cantor[4], Benjamin J. Blencowe[2] & Daniel H. Geschwind[1,4]

- Microarray-data:
  - 58 samples from cortex: 29 autism, 29 control
  - 21 samples from cerebellum: 11 autism, 10 control
- 200 most differentially expressed genes between autism and control in the cortex
  - This makes the analysis only semi-unsupervised since class information was used to filter genes before clustering!

Red: autism
Black: control

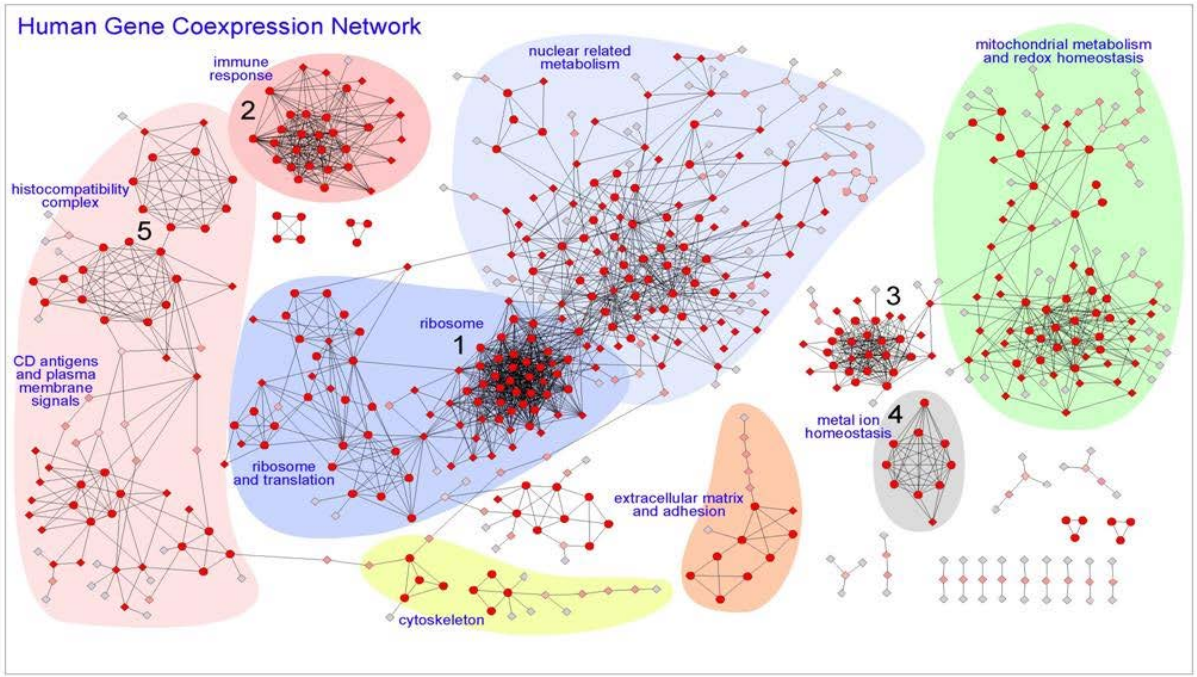


Co-expression networks



- Network: nodes connected by links
- Nodes can be genes, proteins, metabolites, …
- Links represent associations such as similarity

- Co-expression network: two genes are linked if their expression similarity is higher than some threshold ($r > 0.8$, $d < 2$, …)
- Network cluster: sub-networks/modules of nodes with many links between themselves but few links to other nodes in the network

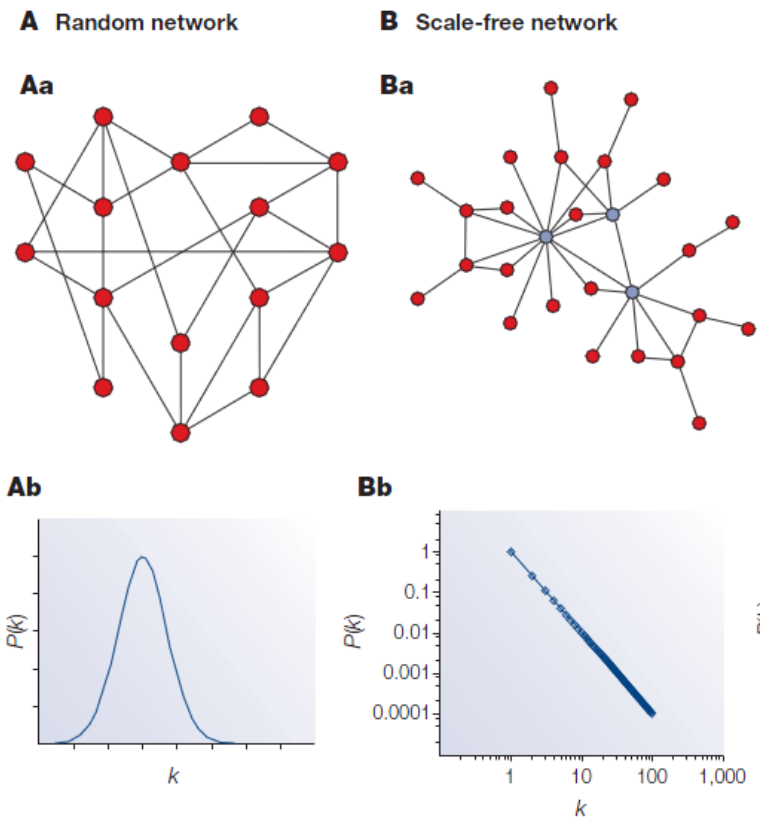


Biological networks

- Random networks: pairs of genes are linked with the same probability *p*
- Biological networks are typically different from random networks:
  - *k* – node degree: the number of links a node has to other nodes (i.e. number of neighbors)
  - *P(k)* – the degree distribution
  - Scale-free network:

$$P(k) \sim k^{-\gamma} \text{ where } 2 < \gamma < 3$$

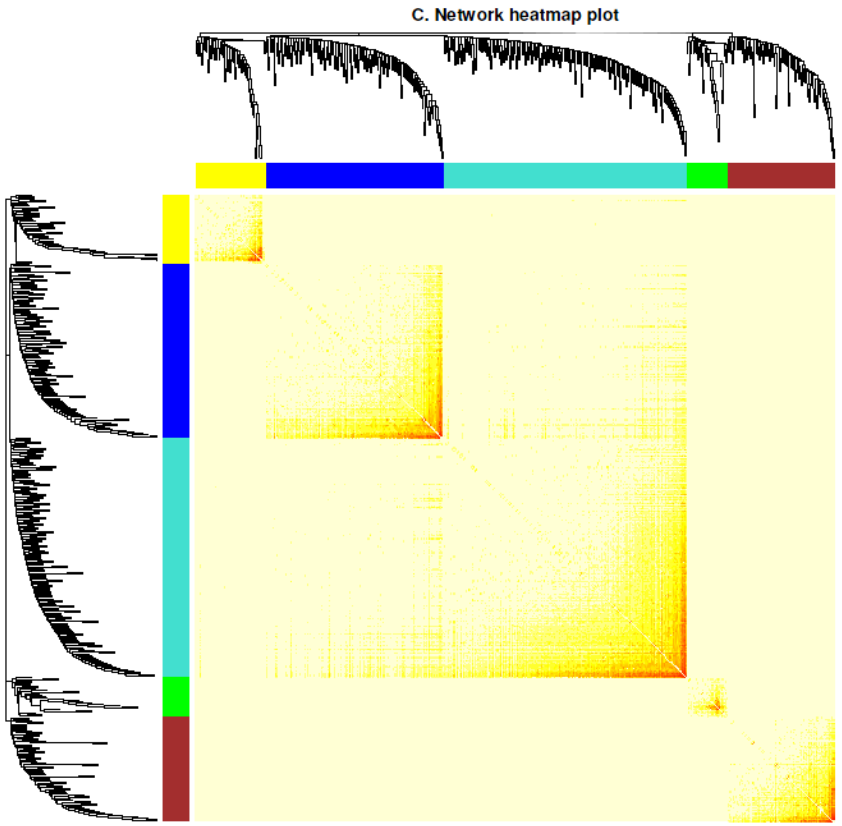
- Gene centrality can be used select important genes:
  - Degree: genes with high degree are hubs
  - Average nearest neighbor (avnn): genes with high avnn degree have neighbors with high degree
  - Betweenness: genes with high betweenness are often in the shortest path between arbitrary pairs of genes in the network



Barabási and Oltvai. *Nature reviews* 5: 101-113, 2004.

Example: WGCNA Network clustering

- Weighted correlation network analysis (WGCNA)
  - Weighted correlation network: no threshold, all pairs of genes are linked but with different weight proportional to their correlation
- Topological overlap: the number of “common neighbors" of two genes in a network
- Modules are found by doing hierarchical clustering on the topological overlap matrix
- Branch cutting methods are used to select the number of modules



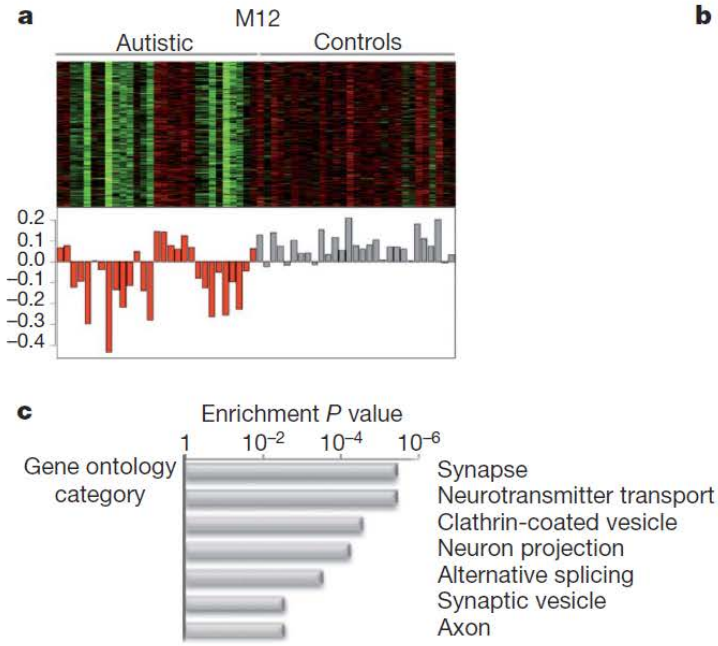
LETTER

doi:10.1038/nature10110

Transcriptomic analysis of autistic brain reveals convergent molecular pathology

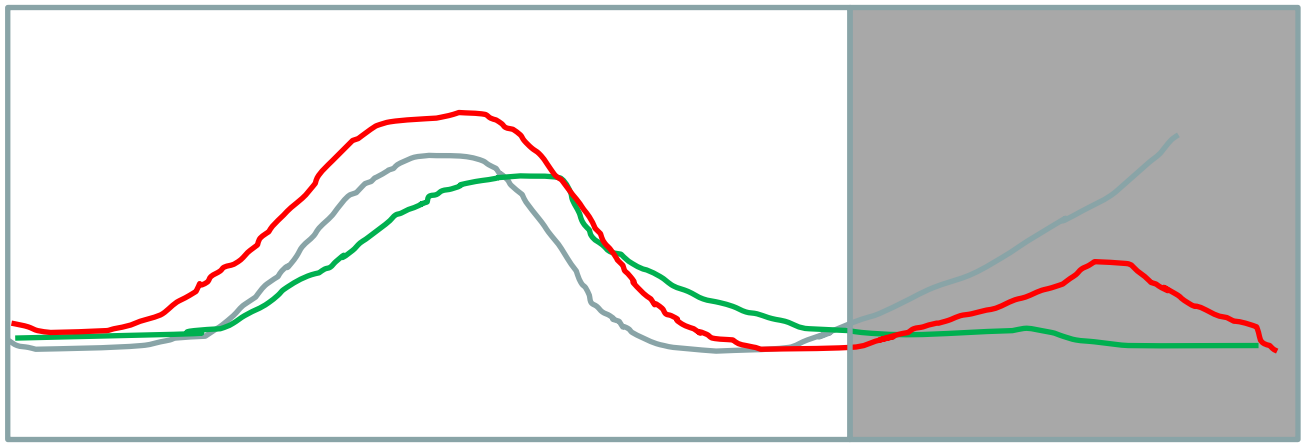
Irina Voineagu[1], Xinchen Wang[2], Patrick Johnston[3], Jennifer K. Lowe[1], Yuan Tian[1], Steve Horvath[4], Jonathan Mill[3], Rita M. Cantor[4], Benjamin J. Blencowe[2] & Daniel H. Geschwind[1,4]

- WGCNA was used to find network modules
- Module M12 is overrepresented for gens known to be «autism-genes» (differentially expressed genes was not!)
- Several hubs in M12 are «autism-genes» (A2BP1, APBA2, etc.)



Bi-clustering

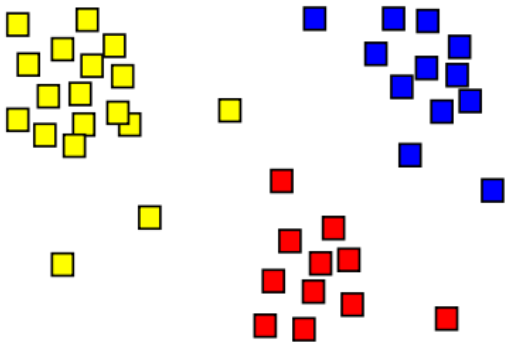
- We have seen that hierarchical clustering can be used to independently cluster both samples and genes
- Bi-clustering clusters samples and genes simultaneously
  - E.g. finds cluster of genes that are co-expressed in a subset of samples
- Example methods: Iterative Signature Algorithm (ISA) and the Ping-Pong Algorithm (PPA)



Machine learning

- **Supervised learning**; used to learn a model from a set of examples given predefined classes (training examples)
  - Example = observation + class
  - Model: e.g. **IF** gene 1420 > 5.3 **THEN** mutant

- Unsupervised learning (clustering, class discovery); used to discover natural groups of observations



2D data: two features X and Y

Three groups discovered:

Yellow: low X, high Y
Red: high X, low Y
Blue: high X, high Y

Example of supervised learning: Decision tree learning

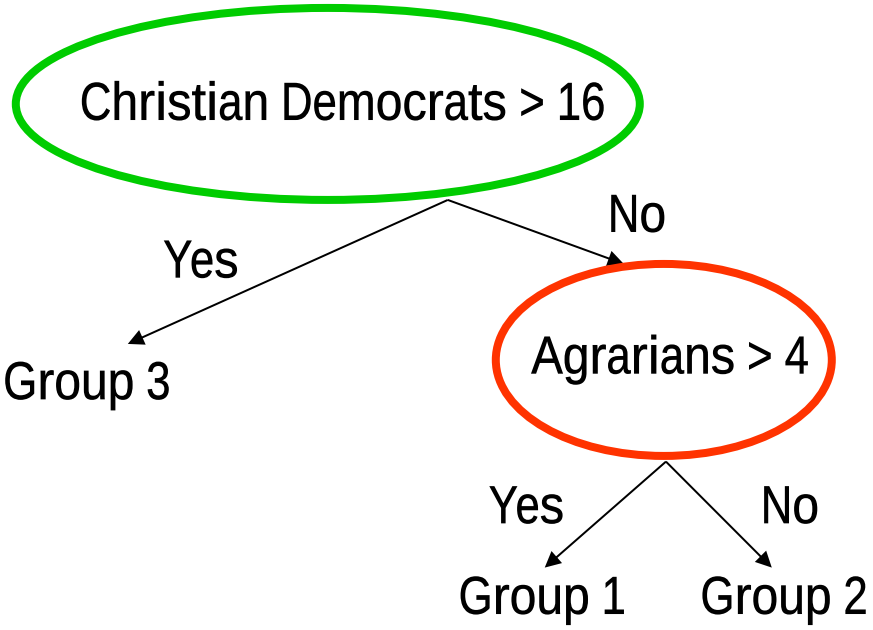
Country	Communists	Socialists	Greens	Social Democrats	Liberals	Agrarians	Subnational, regional and ethnic parties	Christian Democrats	Conservatives	Extreme Right
Norway	0	7	0	38	4	8	0	9	24	6
Sweden	6	0	2	43	10	17	0	2	18	1
Denmark	4	9	0	33	13	14	0	3	15	9
Finland	15	0	2	24	3	25	5	3	21	0
Iceland	0	18	3	16	4	22	0	0	36	0
UK	0	0	9	39	15	0	4	0	42	0
Netherlands	2	5	0	30	23	0	0	37	0	0
Belgium	2	0	4	27	19	0	14	31	0	2
Luxembourg	6	1	3	31	21	0	0	34	0	1
Switzerland	2	2	7	22	23	11	0	22	3	5
Austria	1	0	2	48	0	0	0	41	0	8
Germany	1	0	3	40	9	0	0	46	0	1
France	15	2	2	28	20	0	0	0	25	5
Italy	29	0	3	15	4	0	3	35	2	6
Greece	10	0	0	39	6	0	0	0	44	0
Spain	8	0	0	39	16	0	10	0	21	0
Portugal	15	0	1	31	38	0	0	1	11	0

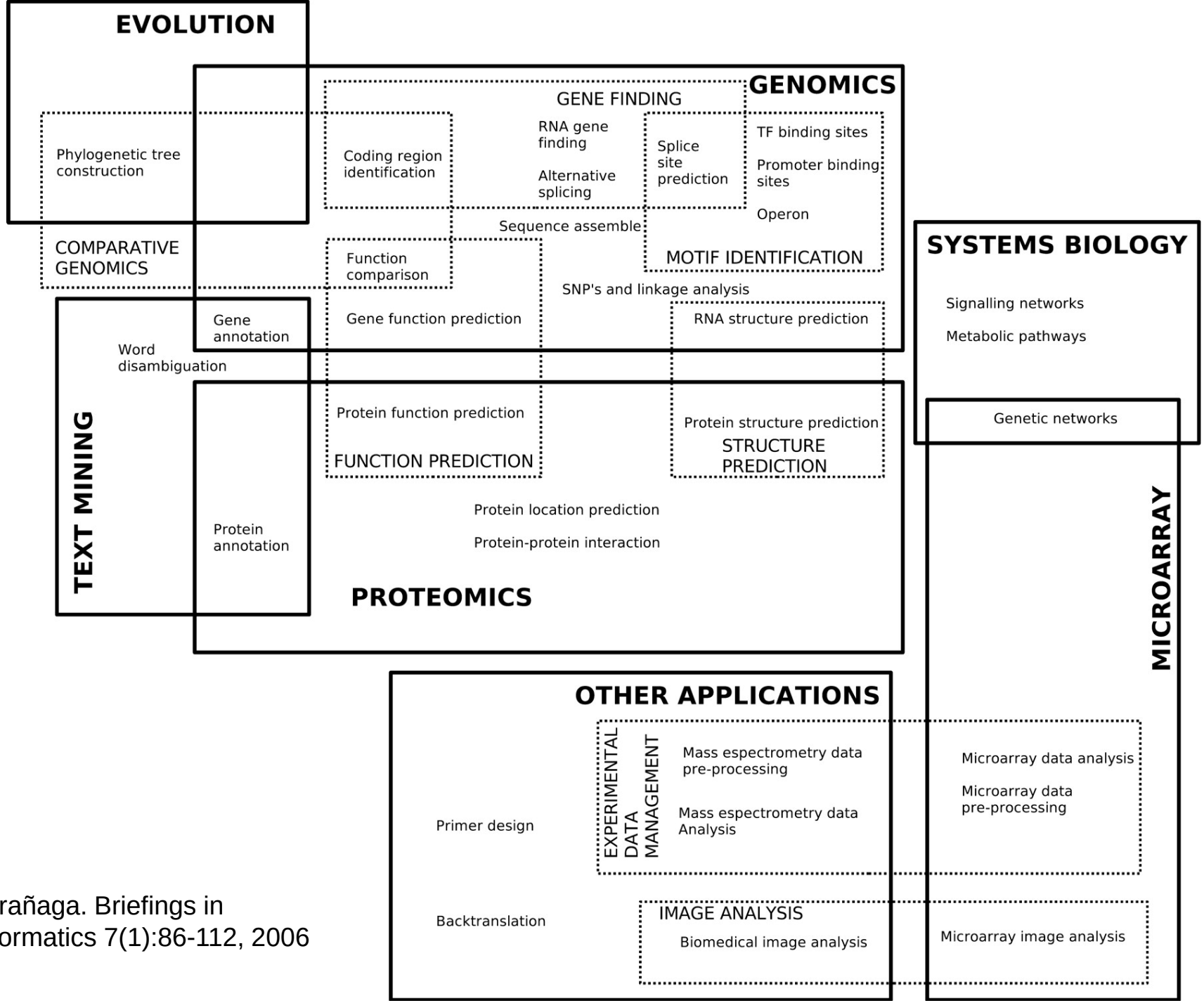
Class knowledge:

Group 1: Nordic countries

Group 2: UK, France, Greece, Spain, Portugal

Group 3: Benelux countries, Switzerland, Austria, Italy, Germany





P. Larrañaga. Briefings in Bioinformatics 7(1):86-112, 2006

Bayes decision rule

likelihood

prior

$$P(\omega_j \mid x) = \frac{p(x \mid \omega_j)P(\omega_j)}{p(x)}$$

posterior

evidence

ω_j: class j (e.g. apples and peaches)

x: observation (e.g. red or orange)

Given a training set (examples) we can estimate:

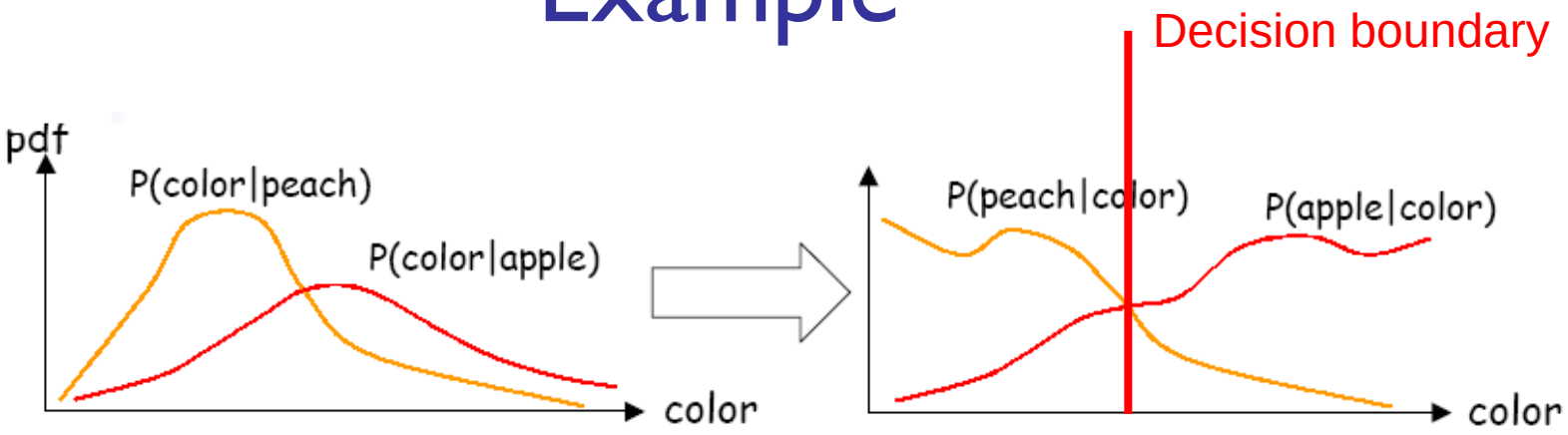
$P(x|\omega_j)$: e.g. the fraction of apples that are red

$P(\omega_j)$: e.g. the fraction of examples that are apples

Bayes decision rule:

If $P(\omega_1|x) > P(\omega_2|x)$> then choose w_1, else choose w_2.

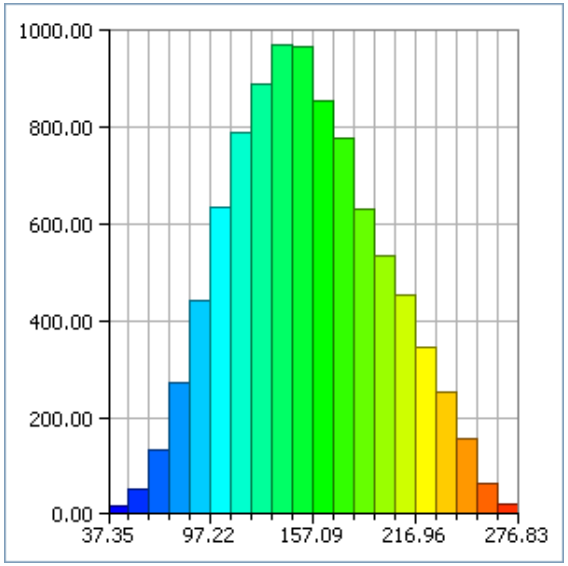
Example



- Assume that we measure the color on a scale from orange to red
- Bayes decision rule
  - If P(apple | color) > P(peach | color) then choose apple
- Gives a decision boundary; a threshold in one dimension, line in two dimensions and (hyper)plane in more than two dimensions
- The evidence p(color) is only necessary for normalization purposes; it does not affect the decision rule

Curse of dimensionality

- We use the examples (training data) to estimate the probability distributions:
  - P(w_j): Easy. Fraction of apples/peaches
  - p(x|w_j): Histogram for apples/peaches!



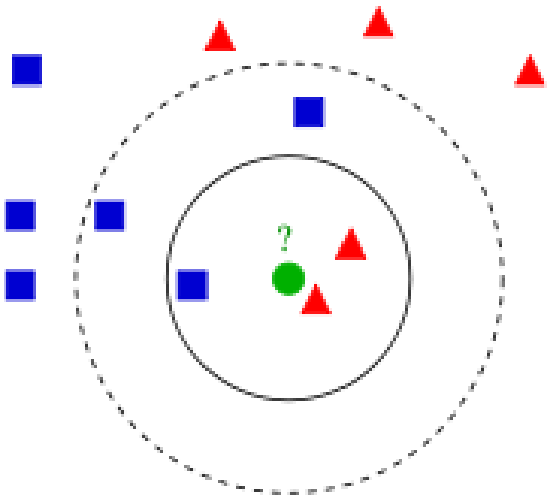
- One feature: bins are rectangles, Two features: cubes, n-features: hyper-cubes
- More dimensions/features require more training data: Curse of dimensionality!
  - If we need 10 observations when we have one feature (to get a good histogram), then we need 10^n observations when we have n-features!
- If the true probability distribution p(w_j | x) is known, then Bayes decision rule is optimal (minimizes error rate)

Feature selection

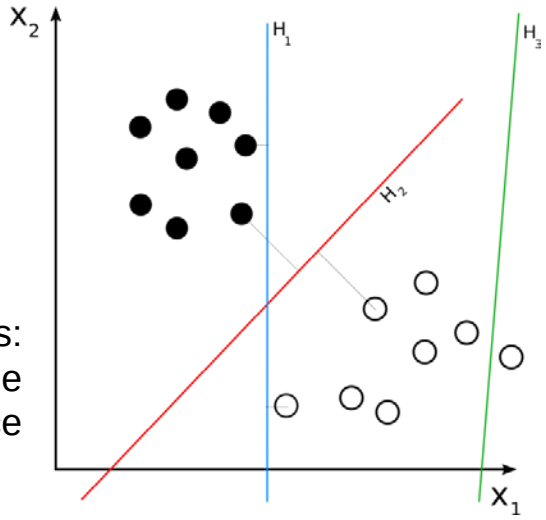
Feature selection is used to deal with the curse of dimensionality; reduce the number of features before model induction

- Filtering methods: compute a statistic for each of the feature's discriminatory capability, rank the features and select the most discriminating ones (e.g. find the differentially expressed genes using the modified t-test)
- Wrapper methods: select a subset of features, induce and validate the resulting model and repeat. Computationally expensive: NP-hard
- Embedded methods: Reduce the number of features as part of the machine learning method (method specific)
- Dimensionality reduction (feature extraction): map your features into a smaller features space (e.g. PCA)

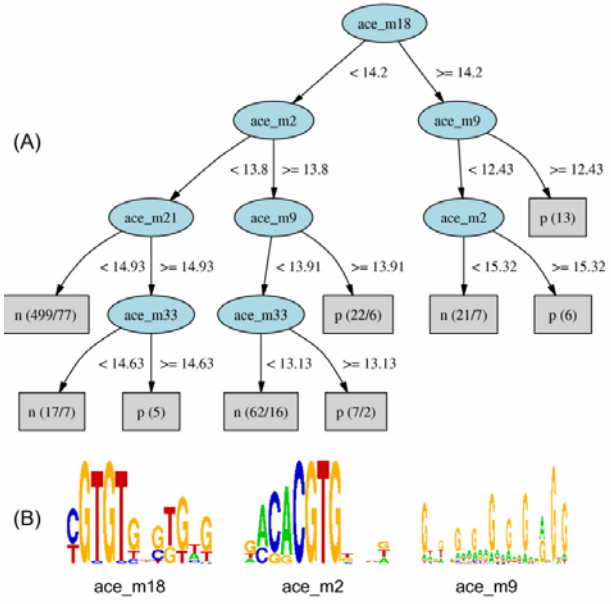
Classes of machine learning methods



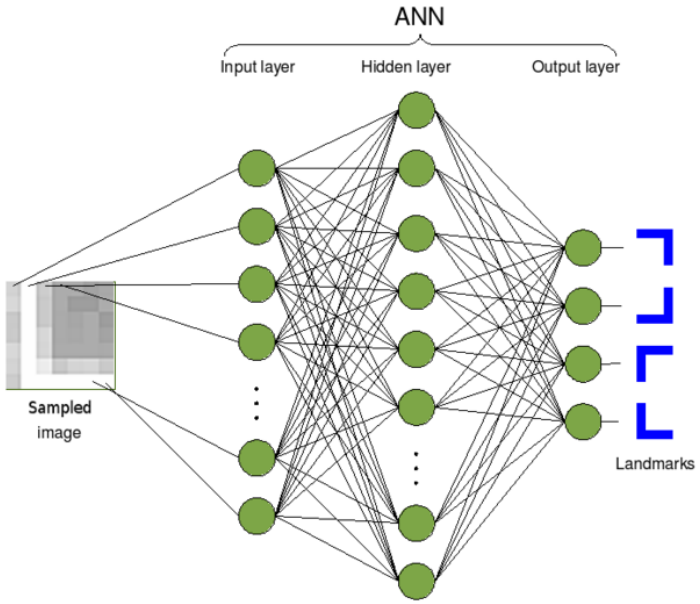
K-nearest neighbor: assign a new observation to the prevalent class among the *k* nearest neighbors in feature space



Separation-based methods: Find a plane that separates the classes in feature space



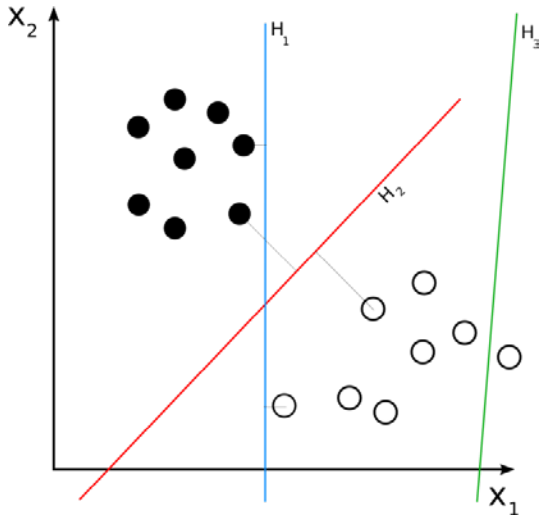
Decision trees/rule-based methods: Find combinations of thresholded features that separate the classes in feature space



Artificial neural networks: Train a network of neurons to transform observations into class predictions

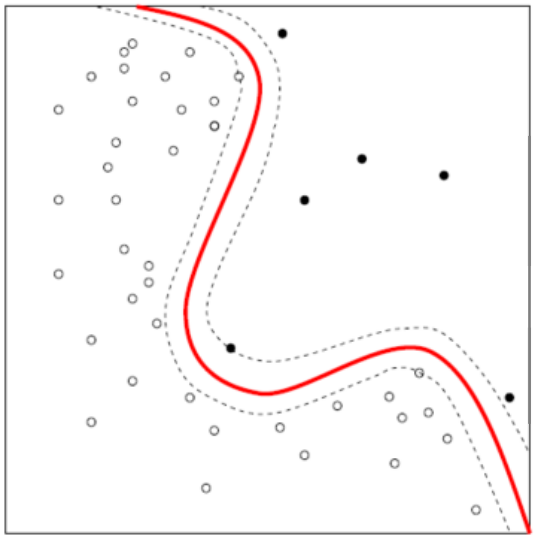
Linear versus non-linear classifiers

- Linear: Finds a hyperplane that separates the classes (decision boundary)
  - In two dimensions: $w_0 + w_1 \cdot x_1 + w_2 \cdot x_2$
  - Use the examples $\boldsymbol{x}$ to estimate $\boldsymbol{w}$
  - w_i is the importance of feature i



- Non-linear: Finds curved decision boundaries
  - Example in two dimensions:

    $w_0 + w_1 \cdot x_1 + w_2 \cdot x_2 + w_3 \cdot x_1 \cdot x_2$
  - Support vector machines use the kernel trick:
    - The kernel maps the observations into a higher dimensional space where the problem is linearly separable
  - Artificial neural networks
    - Only input/output layer (perceptron): linear
    - Hidden layers: non-linear



The machine learning strategy

- Induce the model from a training set of examples
  - Assumes that the training set is representative for the examples you will encounter in the future
- Test the model on an *independent* test set of examples
  - A useful model will generalize to unseen examples i.e. correctly predict the class of these examples
  - If the model can predict examples in the training set, but fails to generalize to the test set, it is overfitted
  - To avoid overfitting, use Occam's razor: "the simplest model that explains the data should be used"
    - Reduce the curse of dimensionality by feature selection
    - Use linear over non-linear methods if applicable
    - …

Model evaluation

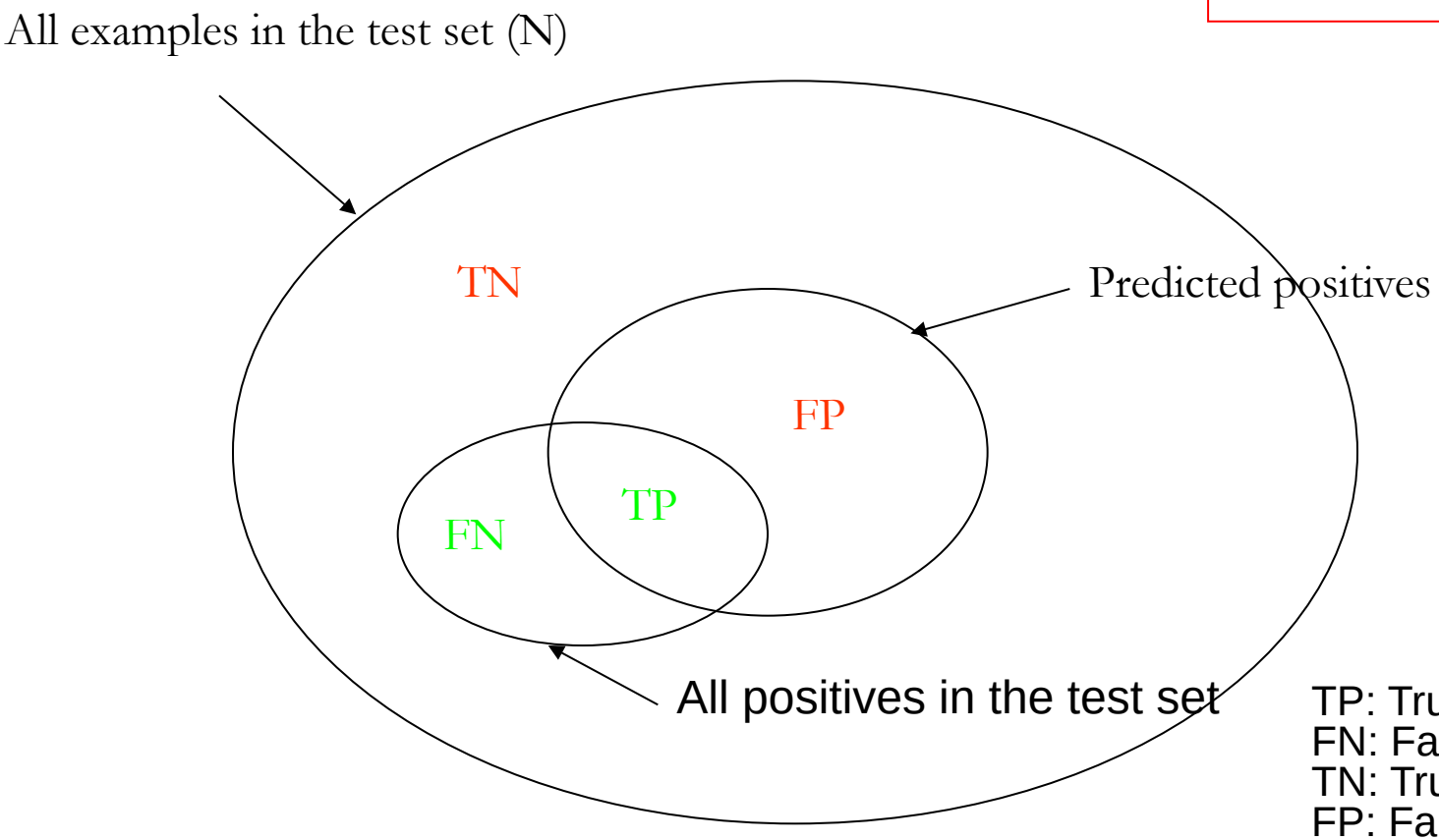
Use a model to predict the classes of a number of examples (test set)

Two classes: Positives and Negatives

Sensitivity: TP/(TP+FN)

Specificity: TN/(TN+FP)

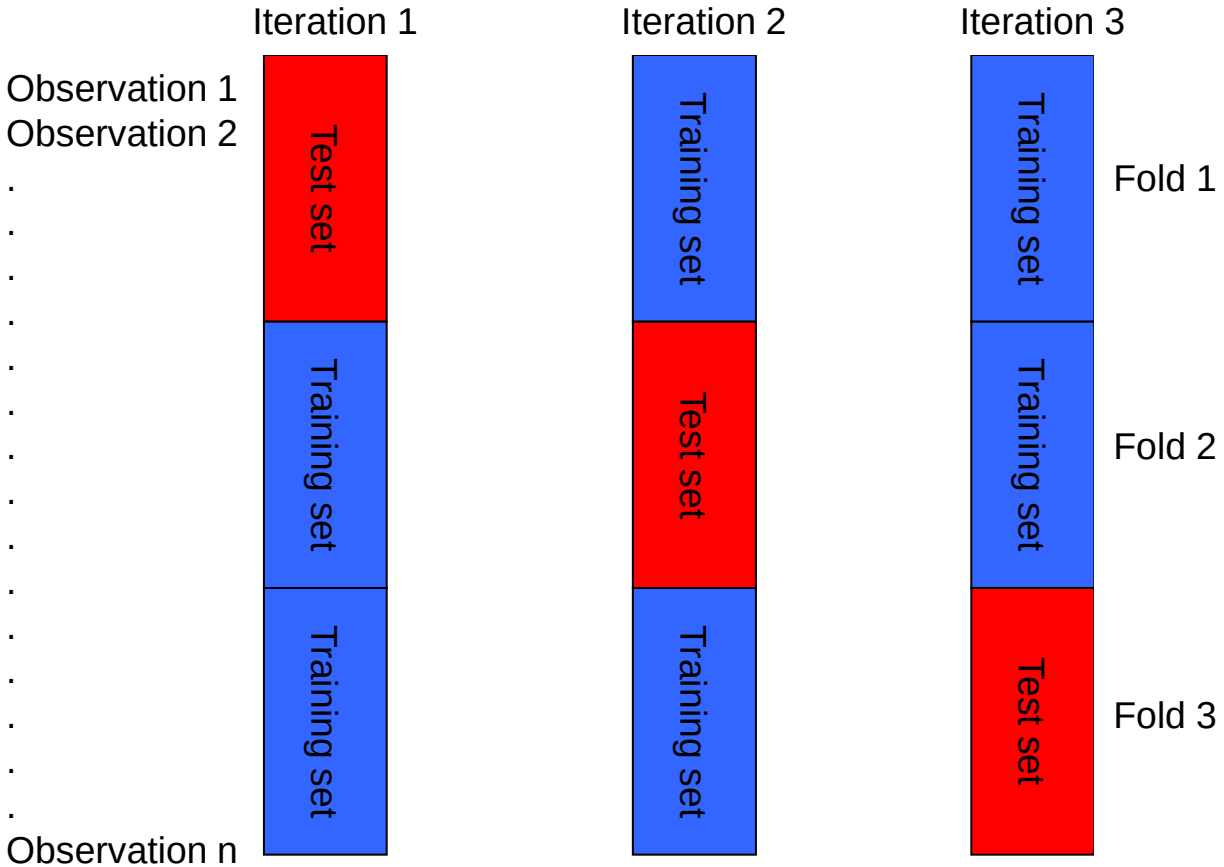
Accuracy = (TP+TN)/N



TP: True positives
FN: False negatives
TN: True negatives
FP: False positives

Cross validation

Approach to divide data into training and test sets



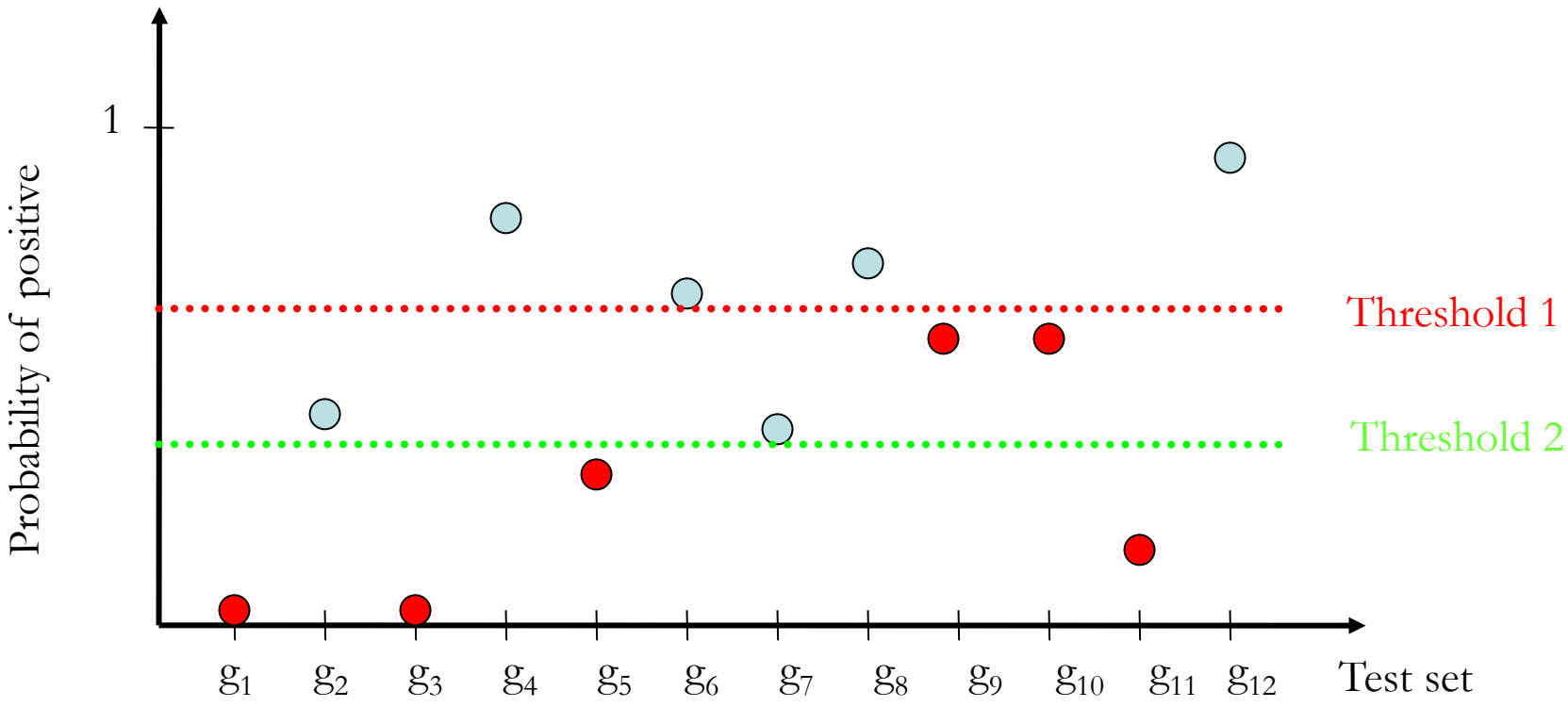
- *k*-fold cross validation: *k* iterations
- Leave-one out cross validation: *n* iterations

Threshold selection

Positives

Negatives

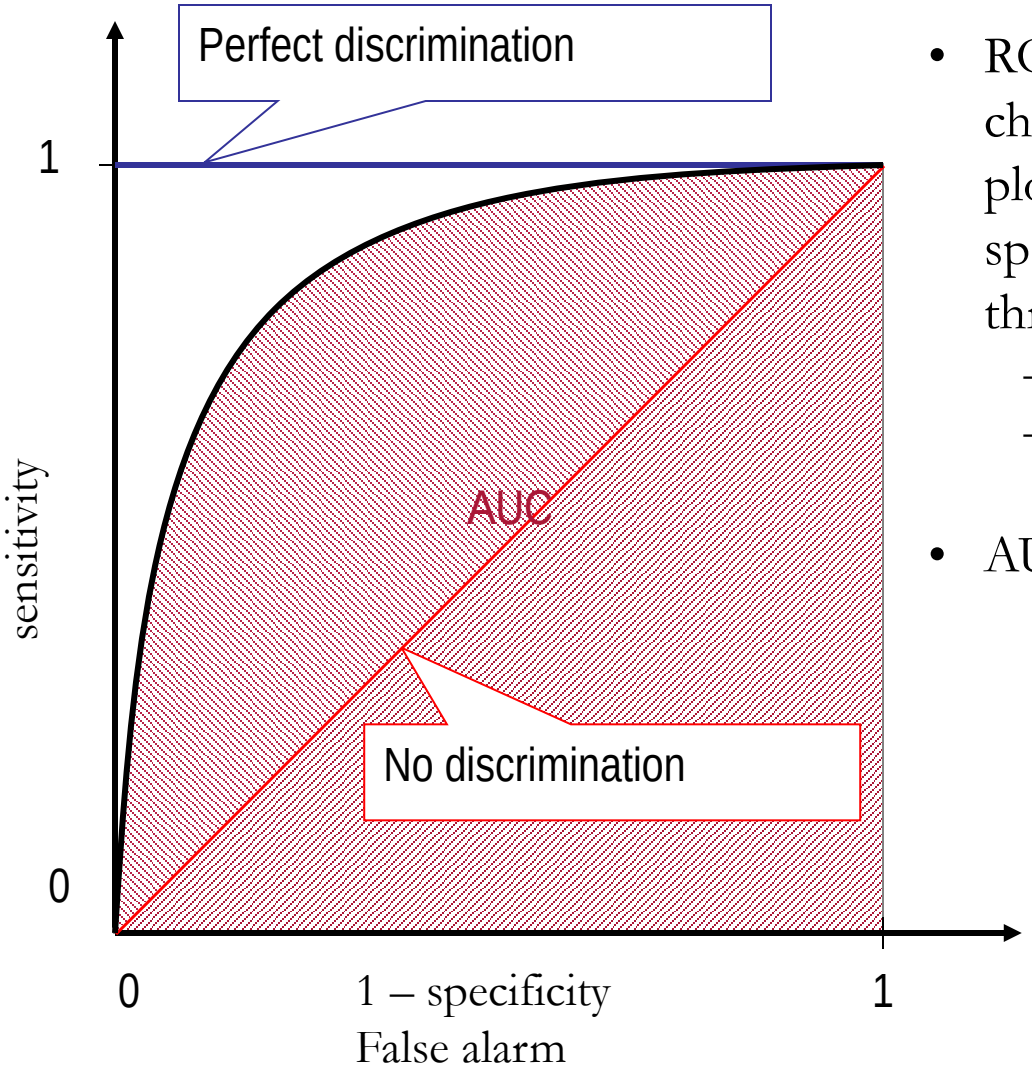
sensitivity: TP/(TP+FN)
specificity: TN/(TN+FP)



Sensitivity = 2/3, Specificity=1

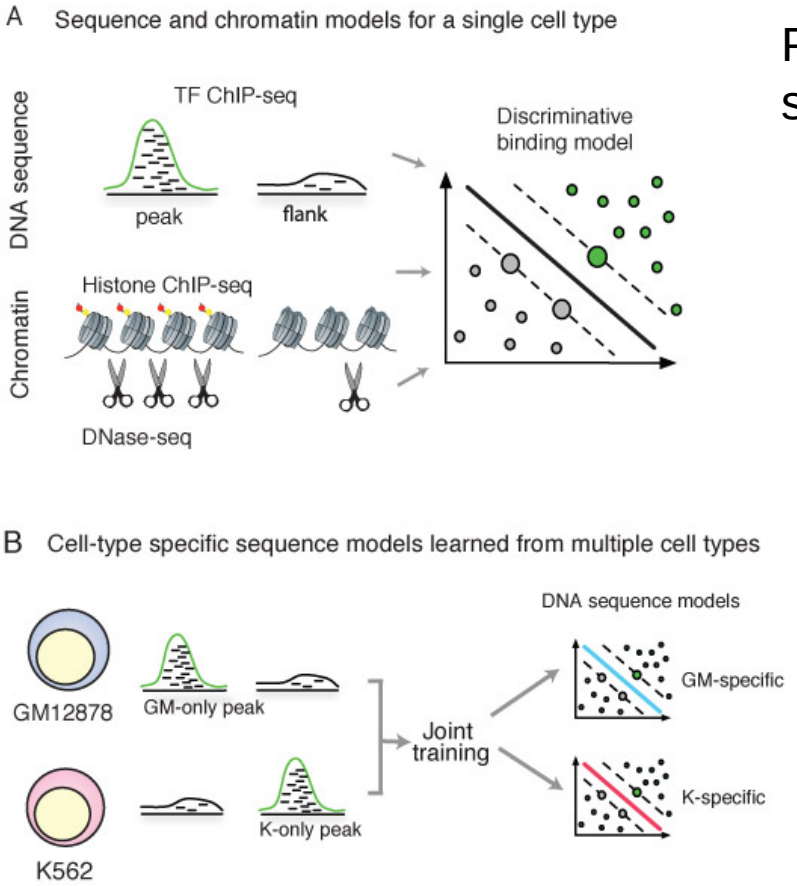
Sensitivity = 1, Specificity=2/3

ROC analysis

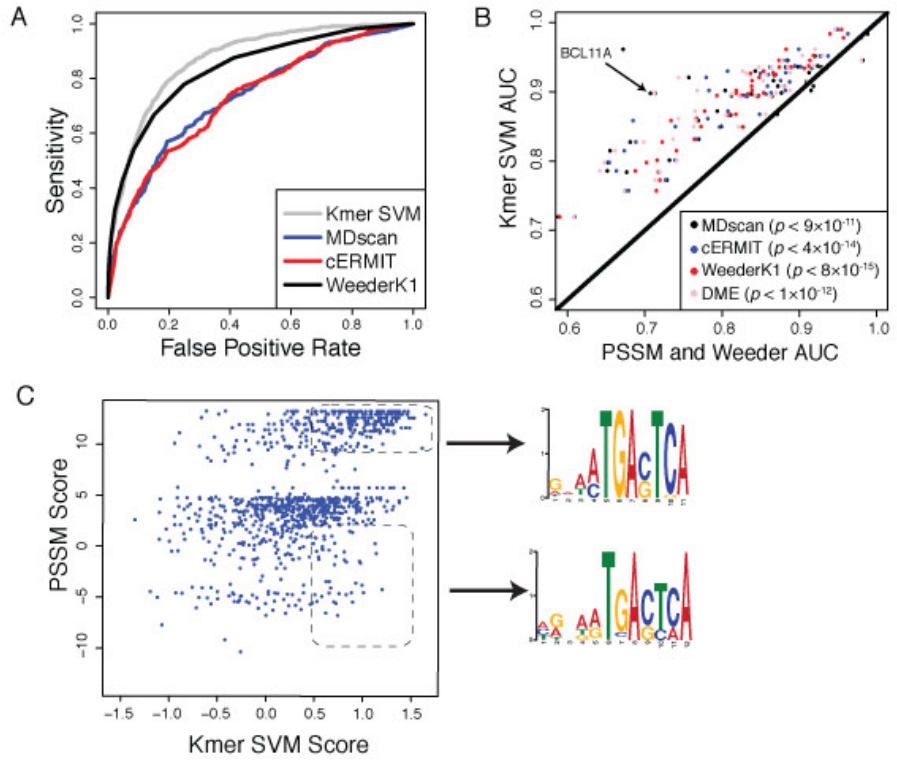


- ROC: Receiver operating characteristics curve results from plotting sensitivity against specificity for all possible thresholds
  - sensitivity: TP/(TP+FN)
  - specificity: TN/(TN+FP)
- AUC: Area under the ROC curve

… nearly every section of the ENCODE integrative paper … was driven by machine learning approaches



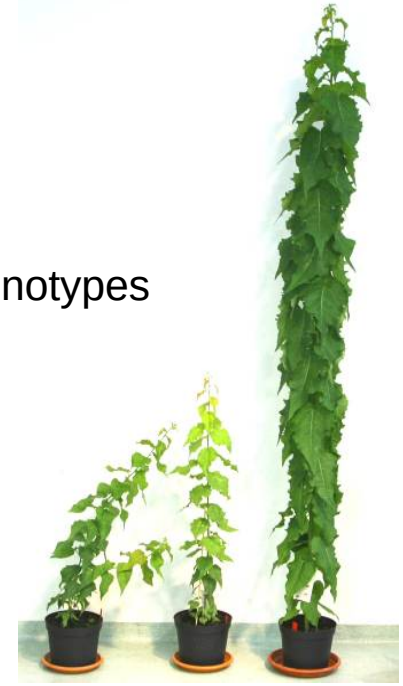
Predict transcription factor occupancy from sequence and chromatin



Machine learning approaches to genomics
http://www.nature.com/encode/threads/machine-learning-approaches-to-genomics

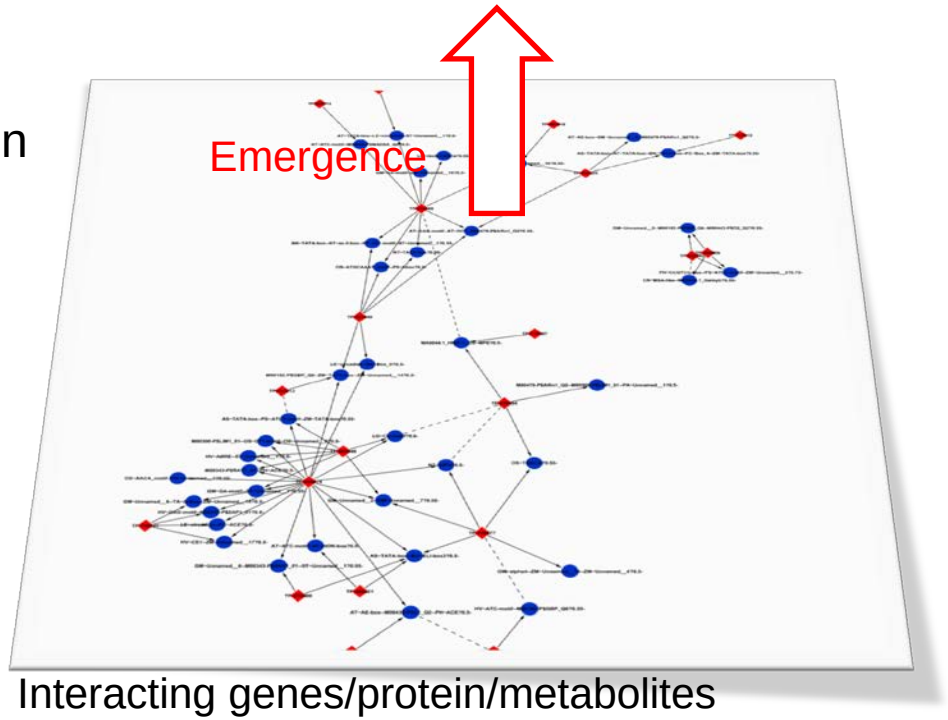
Emergent properties?

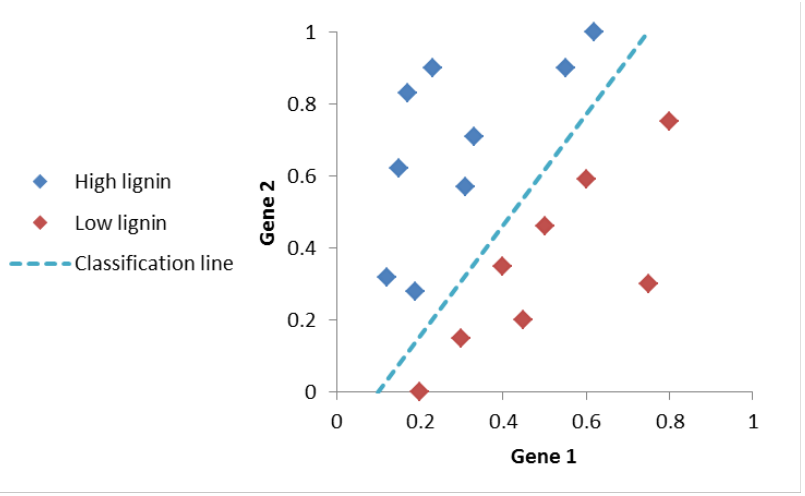
Phenotypes

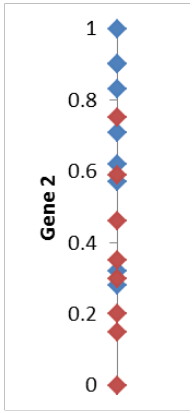


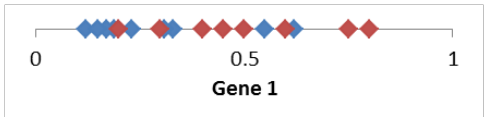
Systems biology?

Modeling emergent properties/interactions in biological systems









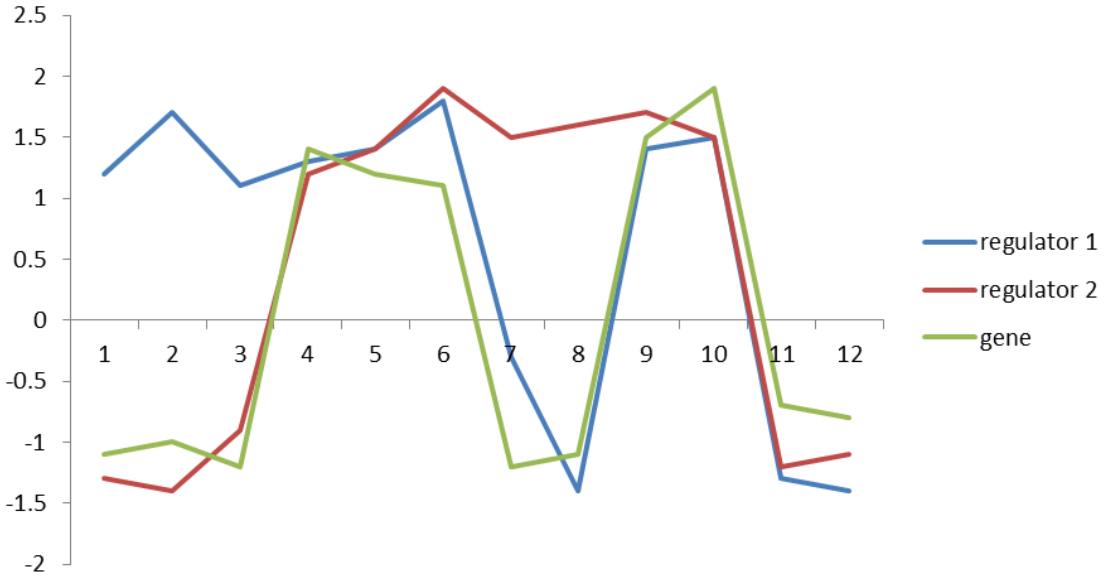
Complex differential expression

The combination of two genes are needed to differentiate the classes

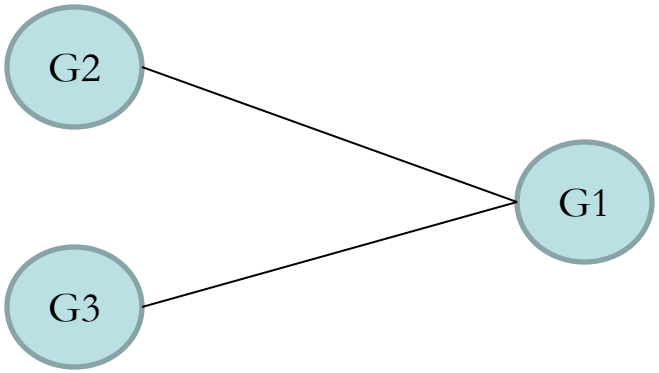
Complex gene regulation

The combination of two regulators are needed to establish the expression of the gene

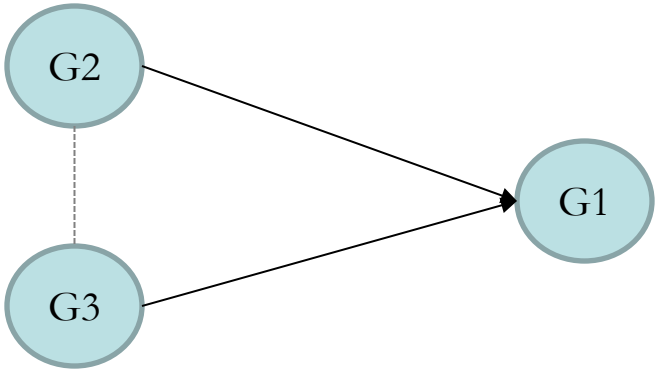
AND-logic



Co-expression networks versus gene networks



Co-expression network:
Expression of G2 correlates with that of G1
Expression of G3 correlates with that of G1

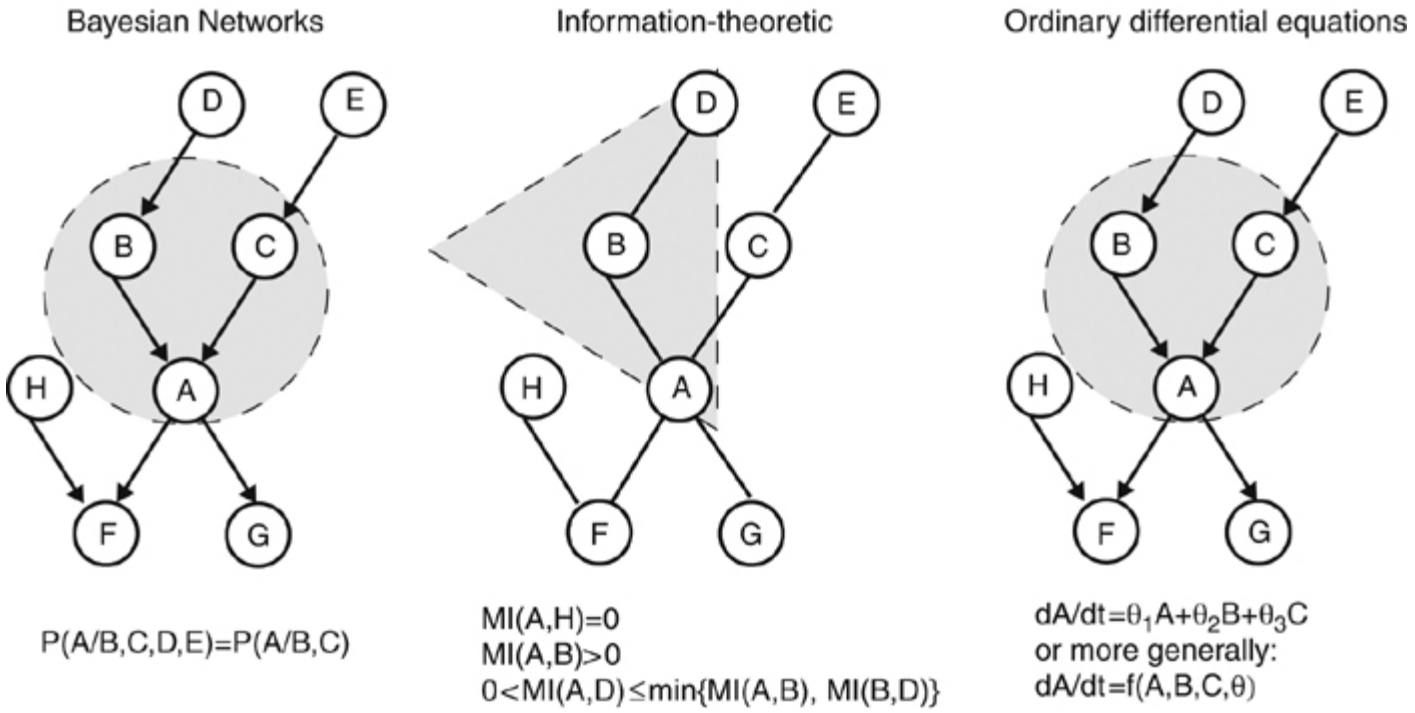


Gene network/regulatory networks:
G2 and G3 influence the expression of G1

1. Physical interaction: transcription factors regulate the gene through binding DNA
2. Influence interaction: physical interaction or indirect interaction via proteins, metabolites, ncRNA, …

Methods for network inference

(Reverse-engineering of networks)



The information-theoretical approach is basically a co-expression network (unsupervised).

Ordinary differential equations with steady state data (dA/dt = 0) becomes a regression approach

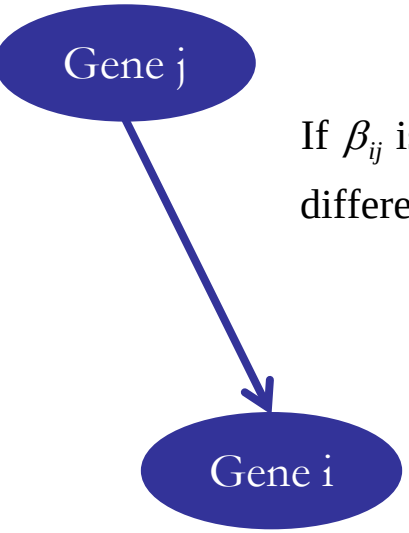
Regression example: Three genes

$\alpha = -0.46$
$\beta_{12} = 0.43$
$\beta_{13} = 0.50$

$$y_1 = \alpha + \beta_{12} y_2 + \beta_{13} y_3$$

Expr	y_2	y_3	y_1	y_1 predicted	
Cond. A	1.2	-1.3	-1.1	$a + \beta_{12} \cdot 1.2 - \beta_{13} \cdot 1.3$	-0.594
Cond. B	1.7	-1.4	-1	$a + \beta_{12} \cdot 1.7 - \beta_{13} \cdot 1.4$	-0.429
Cond. C	1.1	-0.9	-1.2	$a + \beta_{12} \cdot 1.2 - \beta_{13} \cdot 0.9$	-0.437
Cond. D	1.3	1.2	1.4	$a + \beta_{12} \cdot 1.3 + \beta_{13} \cdot 1.2$	0.699
Cond. E	1.4	1.4	1.2	$a + \beta_{12} \cdot 1.4 + \beta_{13} \cdot 1.4$	0.842
Cond. F	1.8	1.9	1.1	$a + \beta_{12} \cdot 1.8 + \beta_{13} \cdot 1.9$	1.264
...	...	...	...	...	...

Correlation: 0.78



If β_{ij} is significantly different from 0!

Choose α, β_{12} and β_{13} so that the correlation between observed (y_1) and predicted (y_1 predicted) expression is maximized!

NB: This is called regression because we predict a continuous value rather than classes

Linear versus non-linear models

- Linear model:

$$y_1 = \alpha + \beta_{12} y_2 + \beta_{13} y_3$$

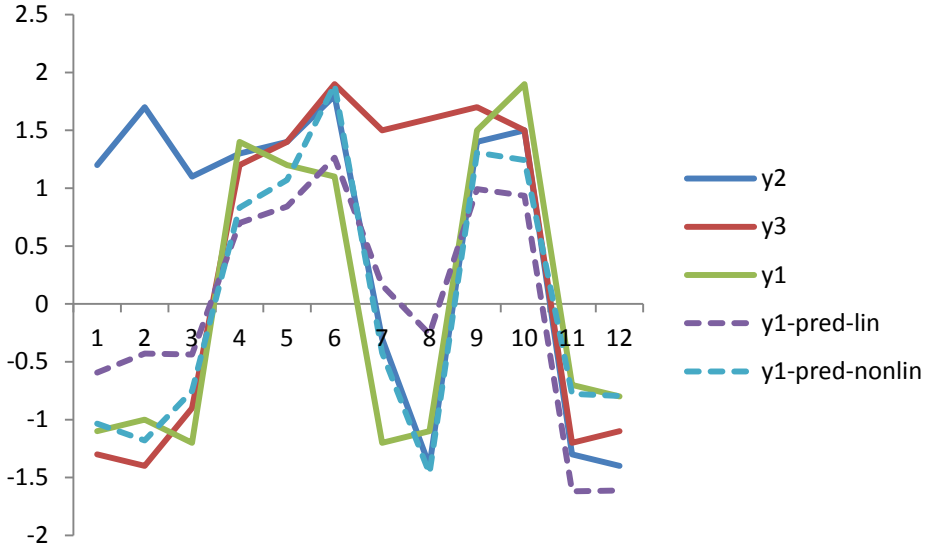
- Non-linear model:

$$y_1 = \alpha + \beta_{12} y_2 + \beta_{13} y_3 + \beta_{123} y_2 y_3$$

$\beta_{123} > 0$: synergistic interactions

$\beta_{123} < 0$: competitive relationship

AND - logic



Linear model:

α = -0.46

β_{12}= 0.43

β_{13}=0.50

Non-linear model:

α = -0.55

β_{12}= 0.37

β_{13}=0.27

β_{123}=0.37

Correlation between observed and predicted:

linear model: 0.77 (P < 0.0029)

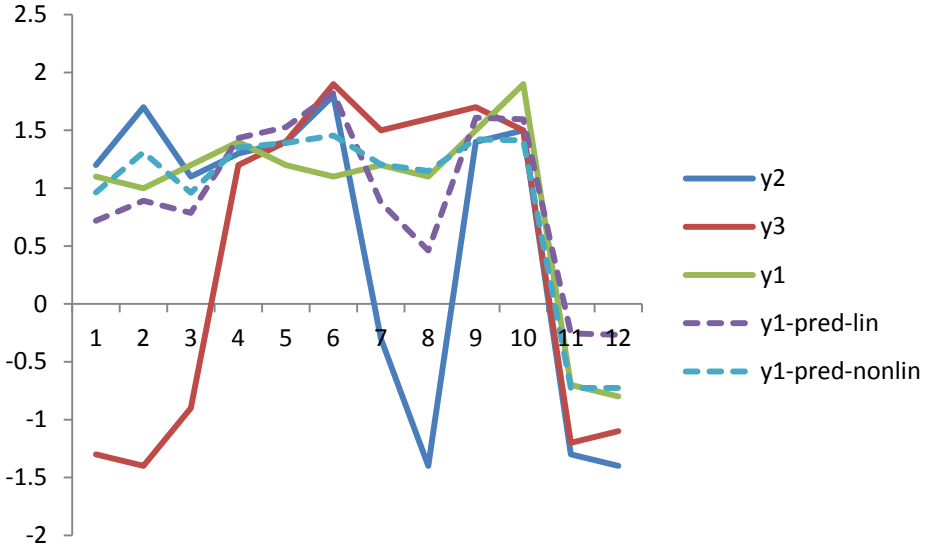
non-linear model: 0.91 (P < 1.2e-5)

Correlation between gene 1 and

gene 2: 0.55 (P < 0.061)

gene 3: 0.65 (P < 0.022)

OR - logic



Linear model:

$\alpha = 0.59$

$\beta_{12} = 0.40$

$\beta_{13} = 0.27$

Non-linear model:

$\alpha = 0.64$

$\beta_{12} = 0.43$

$\beta_{13} = 0.40$

$\beta_{123} = -0.21$

Correlation between observed and predicted:

linear model: 0.85 ($P < 4.5e-4$)

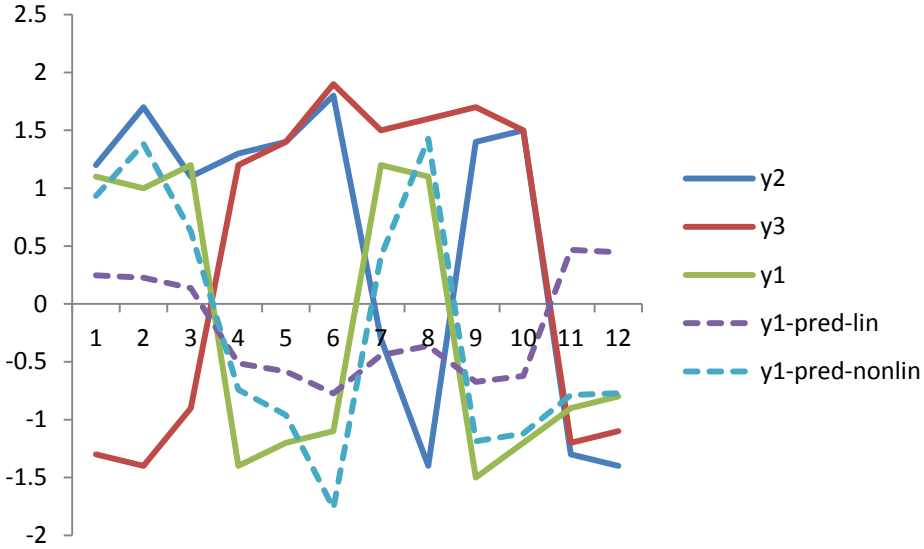
non-linear model: 0.96 ($P < 7.9e-7$)

Correlation between gene 1 and

gene 2: 0.72 ($P < 0.0086$)

gene 3: 0.60 ($P < 0.041$)

XOR - logic



Linear model:

α = -0.02

β_{12}= -0.10

β_{13}=-0.30

Non-linear model:

α = 0.11

β_{12}= -0.01

β_{13}=0.03

β_{123}=-0.56

Correlation between observed and predicted:

Linear model:	0.40 (P < 0.19)
Non-linear model:	0.92 (P < 1.83e-5)

Correlation between gene 1 and

gene 2:	-0.19 (P < 0.56)
gene 3:	-0.39 (P < 0.21)

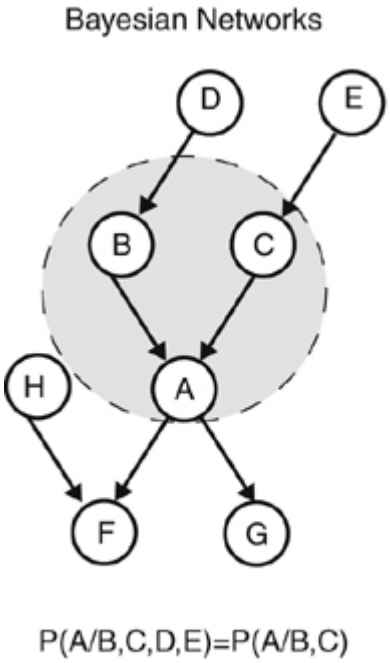
Bayesian networks

- Network: G = (V, E) where V is the set of nodes (genes) and E is a set of directed links
- D is the expression data

Posterior Likelihood Prior

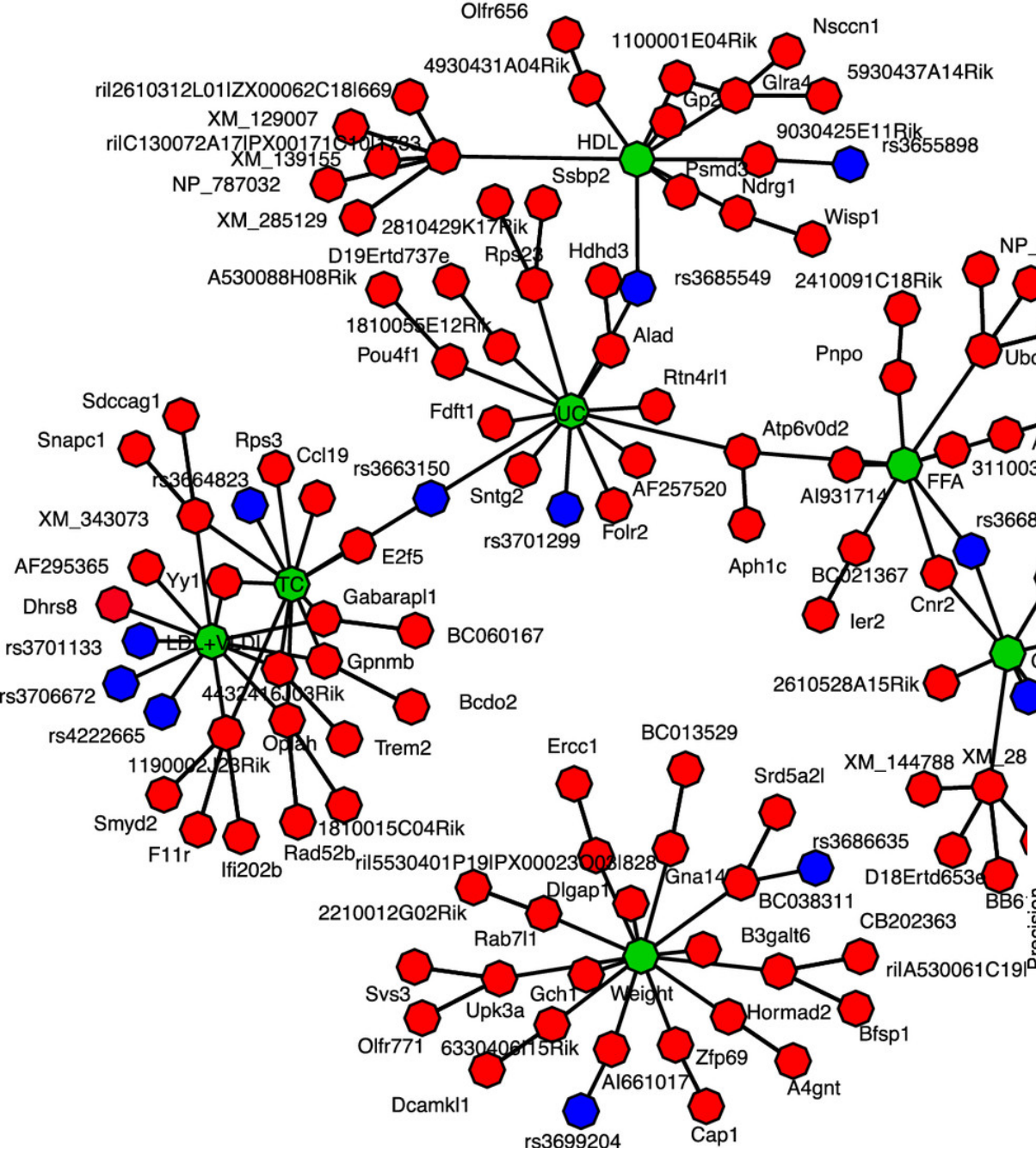
$$P(G|D) = \frac{P(D|G)P(G)}{P(D)}$$

Normalization constant

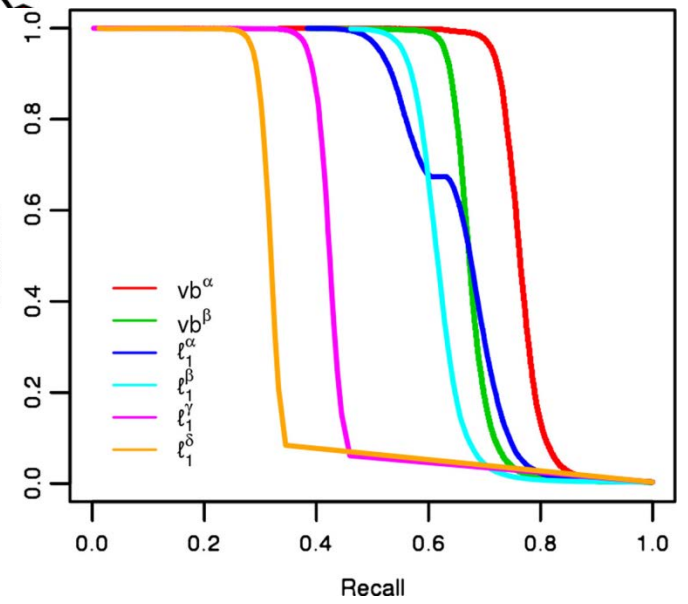


- Computing the posterior probability for all possible networks (Gs) is NP hard: Approximation algorithms!
- One approach to compute the likelihood is to discretize the expression data (e.g. up- and down-regulated) …
  - $L = \sum_{v \varepsilon V} \sum_{o \epsilon D} \log P(v|pa(v), o)$
  - E.g. P(A=up | B=up,C=down) = "fraction of times it happens in the data"!
- … and use P(G) to penalize complex networks (many links)
- … but there are many methods for both searching for networks and scoring networks using data

Bayesian network to explain mouse obesity



The Bayesian approaches (vb) works better than lasso regression (l)



BMC Bioinformatics 2012, 13:53

Key points in this lecture

- Unsupervised learning (clustering)
  - Hierarchical clustering
  - Co-expression networks
- Supervised learning
  - Bayes decision rule
  - Linear and non-linear models
  - Gene/regulatory networks
  - Model evaluation/Cross validation
  - Overfitting/Curse of dimensionality / Occam's razor
  - Feature selection