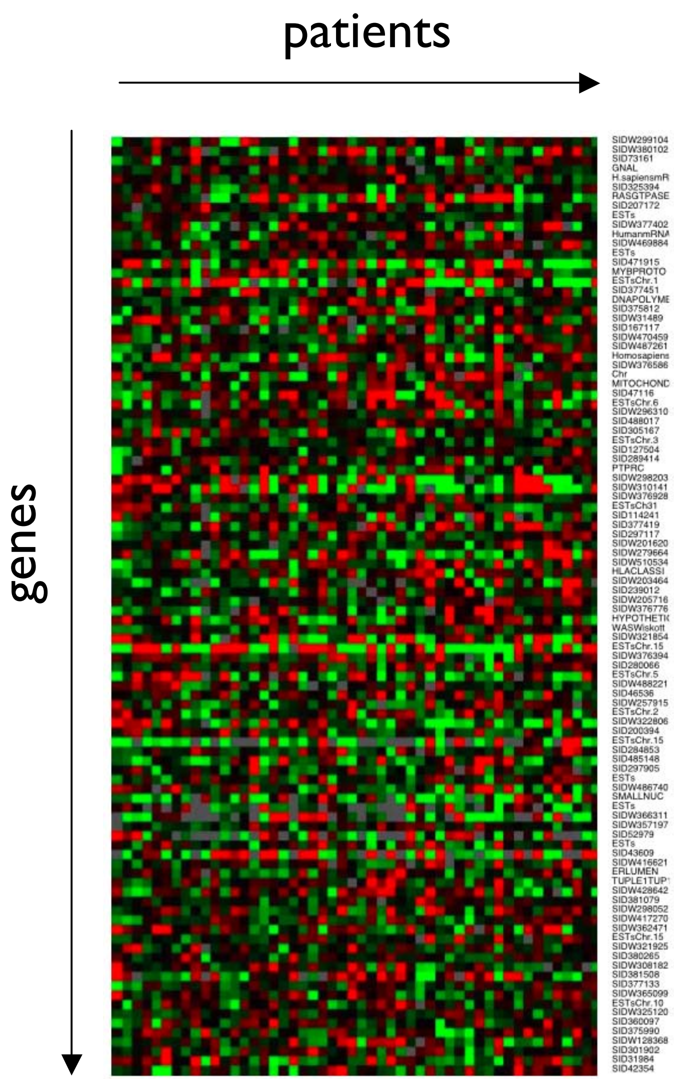
Clustering and classification with applications to microarrays and cellular phenotypes

Gregoire Pau, EMBL Heidelberg
gregoire.pau@embl.de

EMBL European Molecular Biology Laboratory

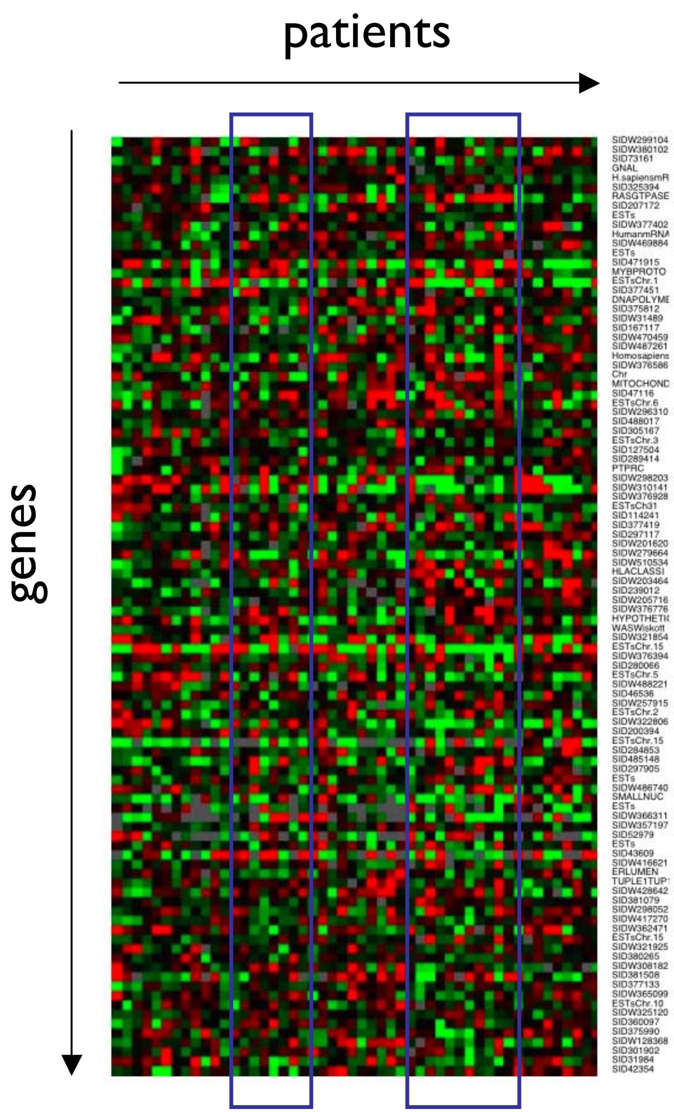
Microarray data



- Clustering
  - Are there patient groups with similar expression profiles ?
  - Are there groups of genes behaving similarly ?

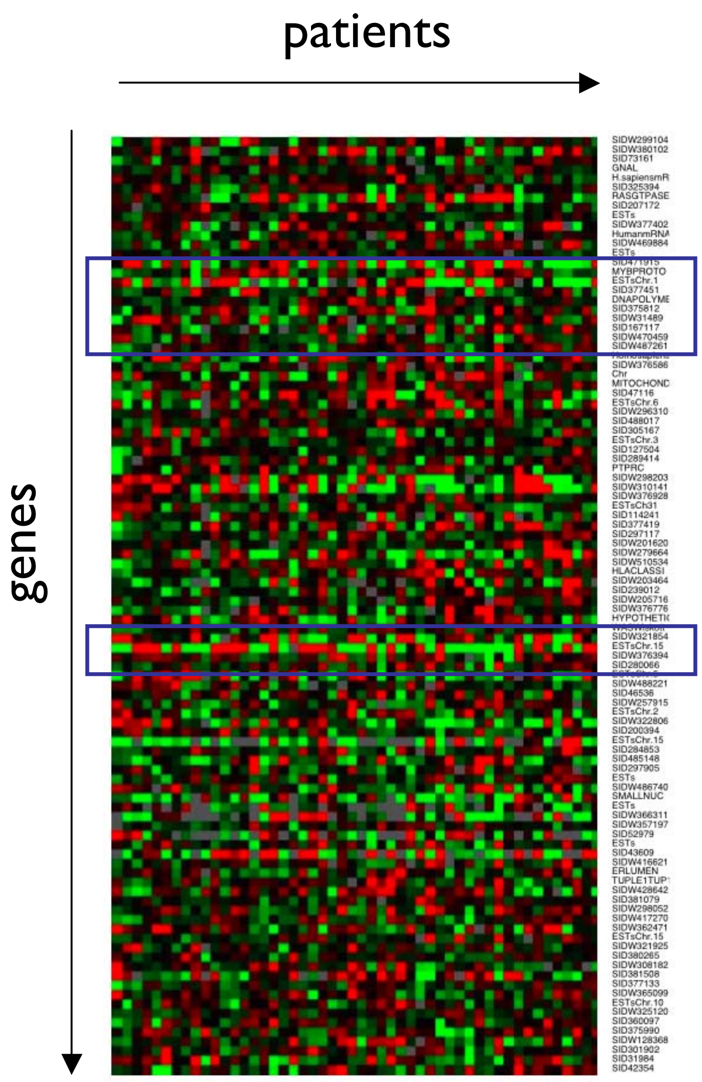
- Classification
  - Given known cancer type profiles
  - Which cancer type has a patient given his expression profile ?

Microarray data



- Clustering
  - Are there patient groups with similar expression profiles ?
  - Are there groups of genes behaving similarly ?
- Classification
  - Given known cancer type profiles
  - Which cancer type has a patient given his expression profile ?

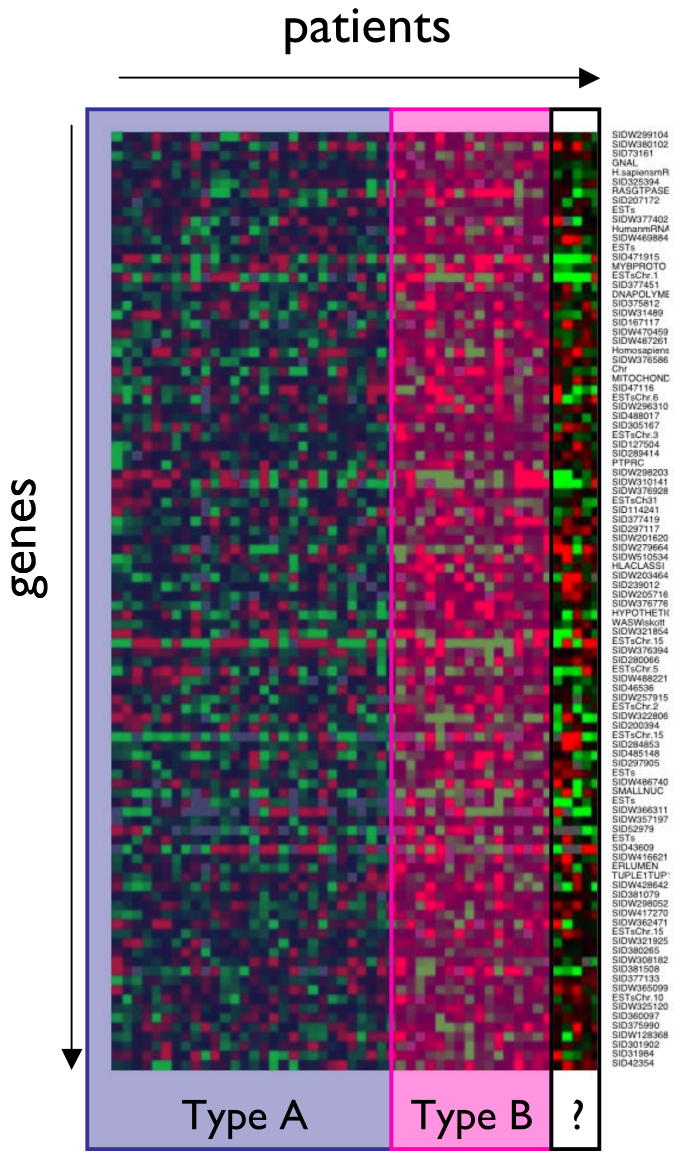
Microarray data



- Clustering
  - Are there patient groups with similar expression profiles ?
  - Are there groups of genes behaving similarly ?

- Classification
  - Given known cancer type profiles
  - Which cancer type has a patient given his expression profile ?

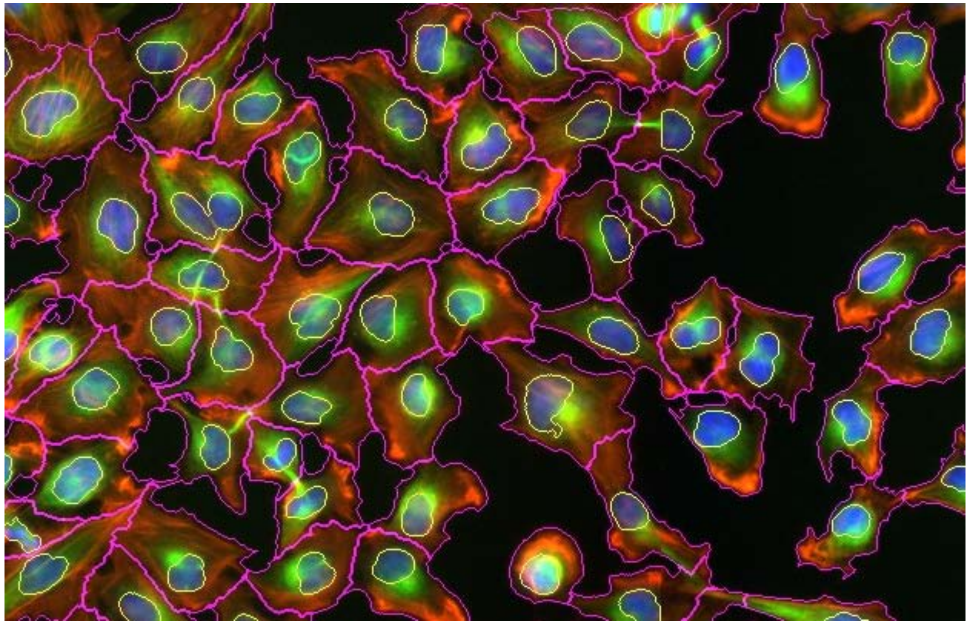
Microarray data

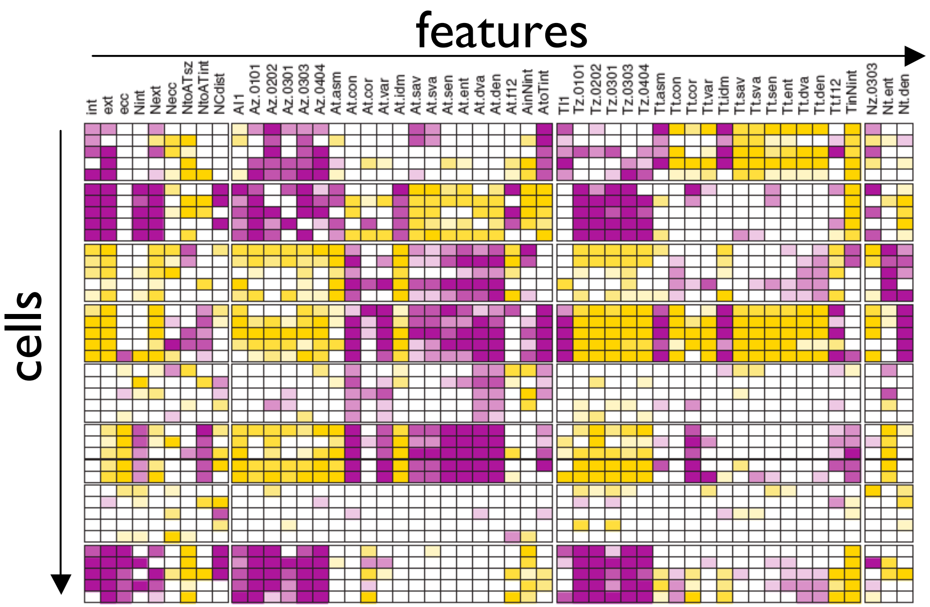


- Clustering
  - Are there patient groups with similar expression profiles ?
  - Are there groups of genes behaving similarly ?

- Classification
  - Given known cancer type profiles
  - Which cancer type has a patient given his expression profile ?

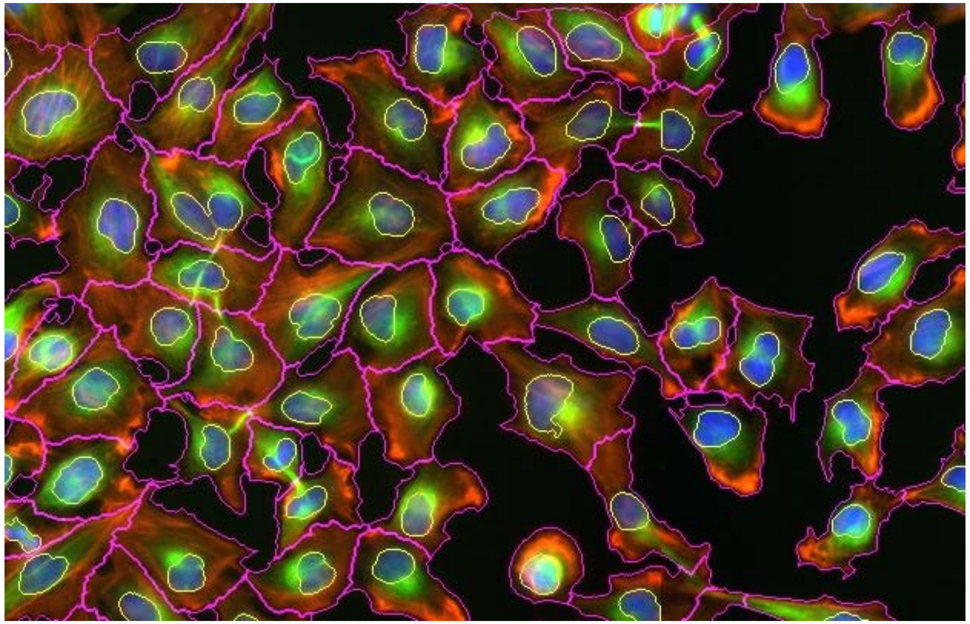
Cell phenotypes

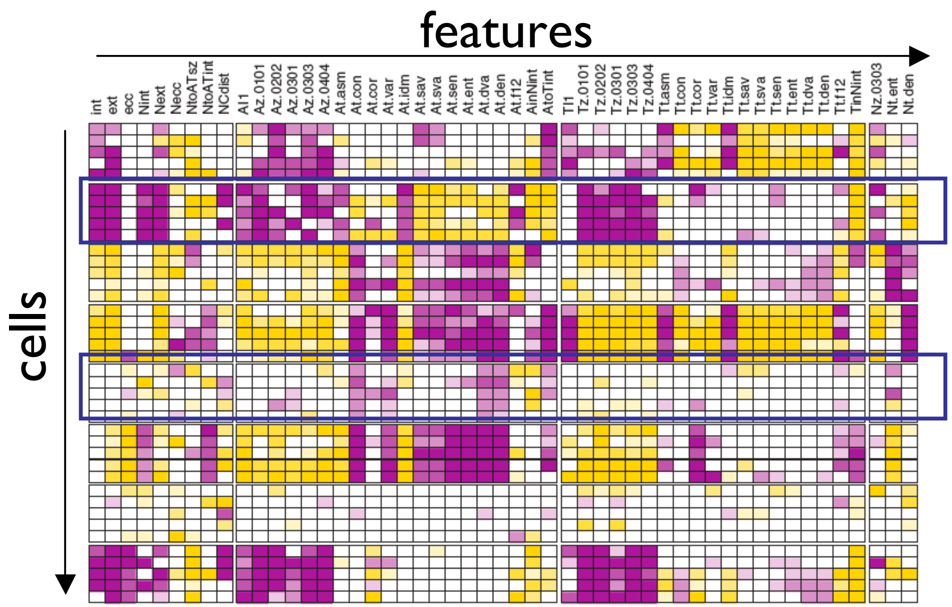




- Clustering
  - Are there similar cell groups ?
- Classification
  - Given known cell types
  - What is this cell type ?

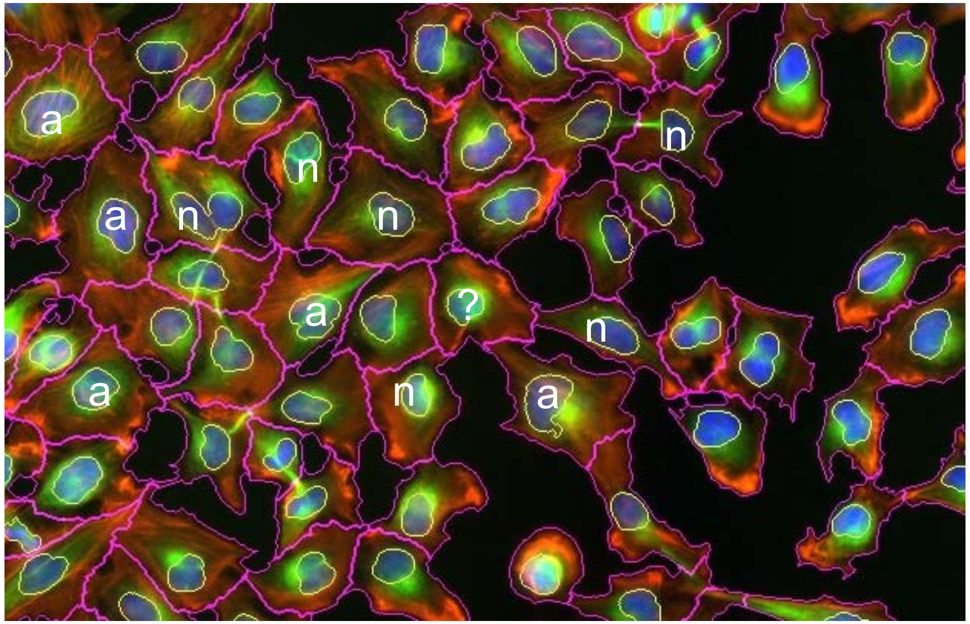
Cell phenotypes



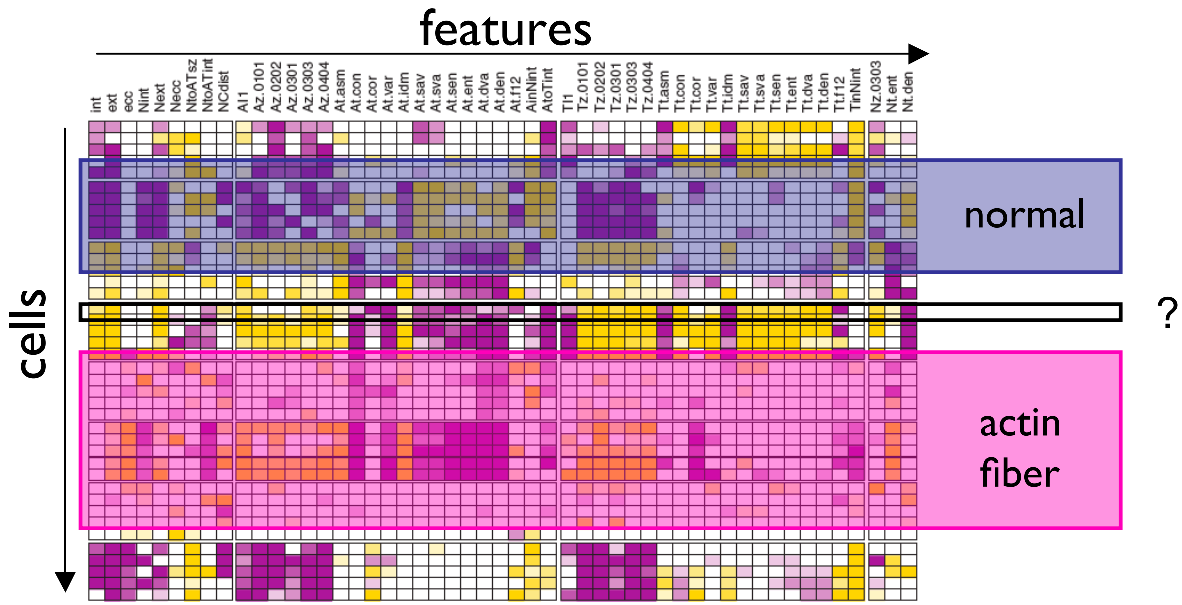


- Clustering
  - Are there similar cell groups ?
- Classification
  - Given known cell types
  - What is this cell type ?

Cell phenotypes



- Clustering
  - Are there similar cell groups ?
- Classification
  - Given known cell types
  - What is this cell type ?



Clustering versus classification

- Clustering
  - Unknown class labels
  - Given a measure of similarity between objects
  - Identification of similar subgroups
  - Ill-defined problem

- Classification
  - Known class labels
  - Prediction/classification/regression of class labels
  - Well-defined

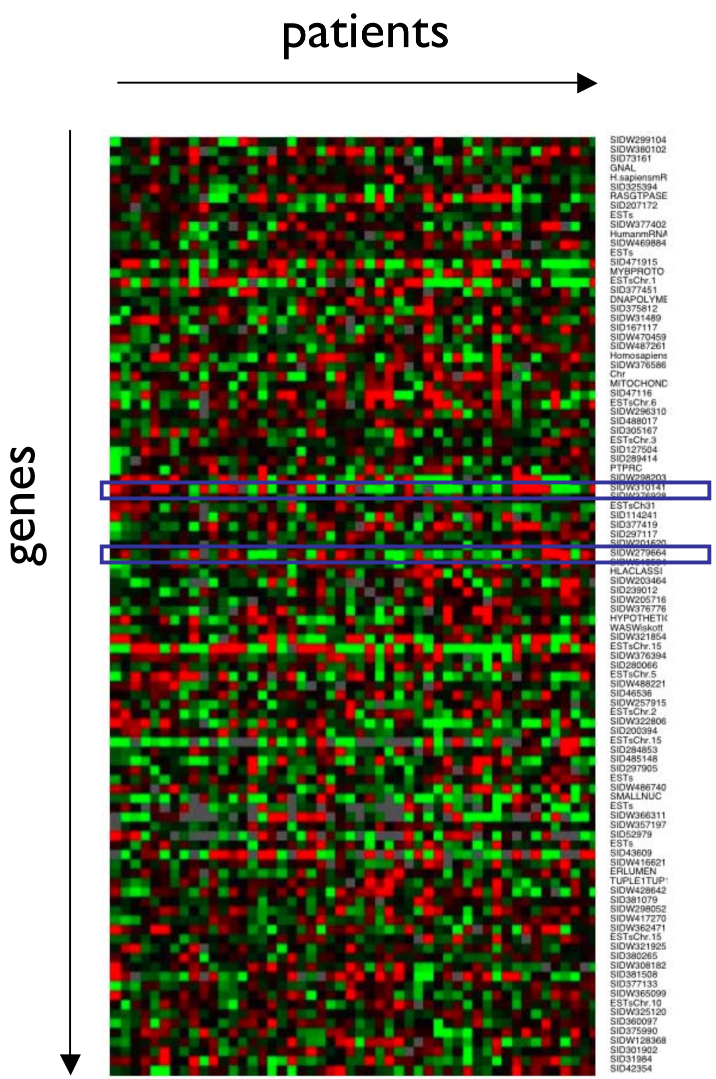
Clustering

Clustering

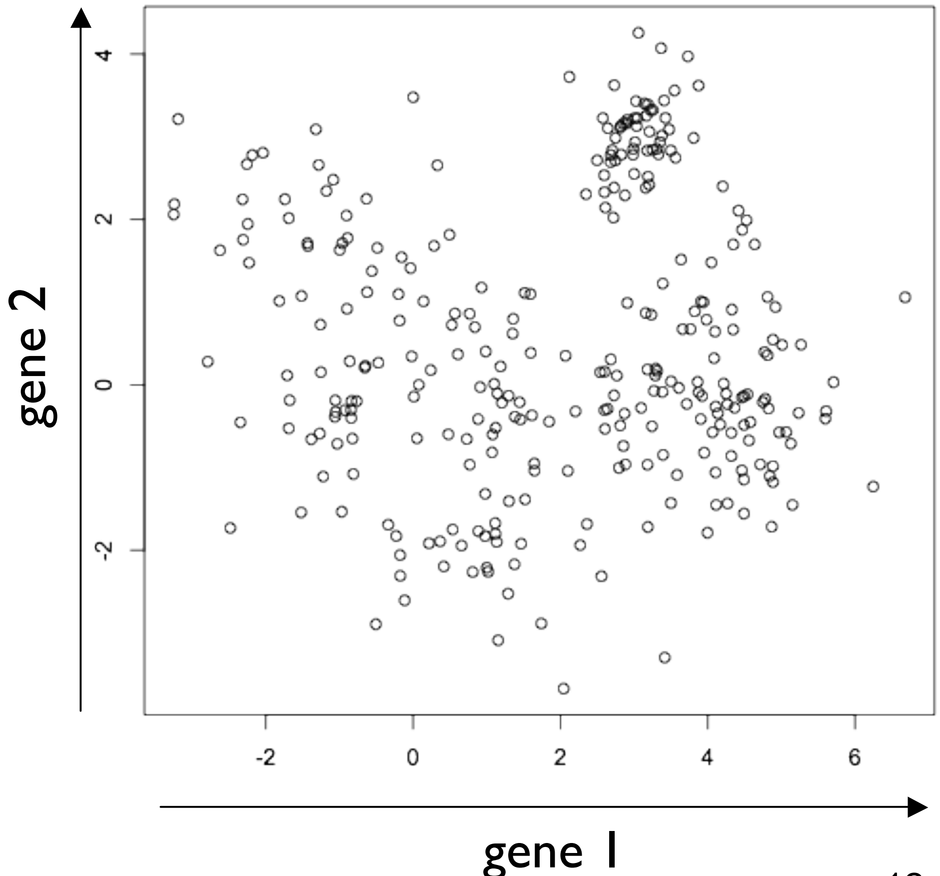
- Identification of similar subgroups within data
- Using a similarity measure between objects
- Ill-defined problem ⇒ many algorithms exist

- Non-parametric clustering
  - Agglomerative: Hierarchical clustering
  - Partitive: K-means
  - Partitive: Partitioning Around Medioids (PAM)
  - Other: Self-organising maps
- Parametric clustering
  - Gaussian mixture estimation

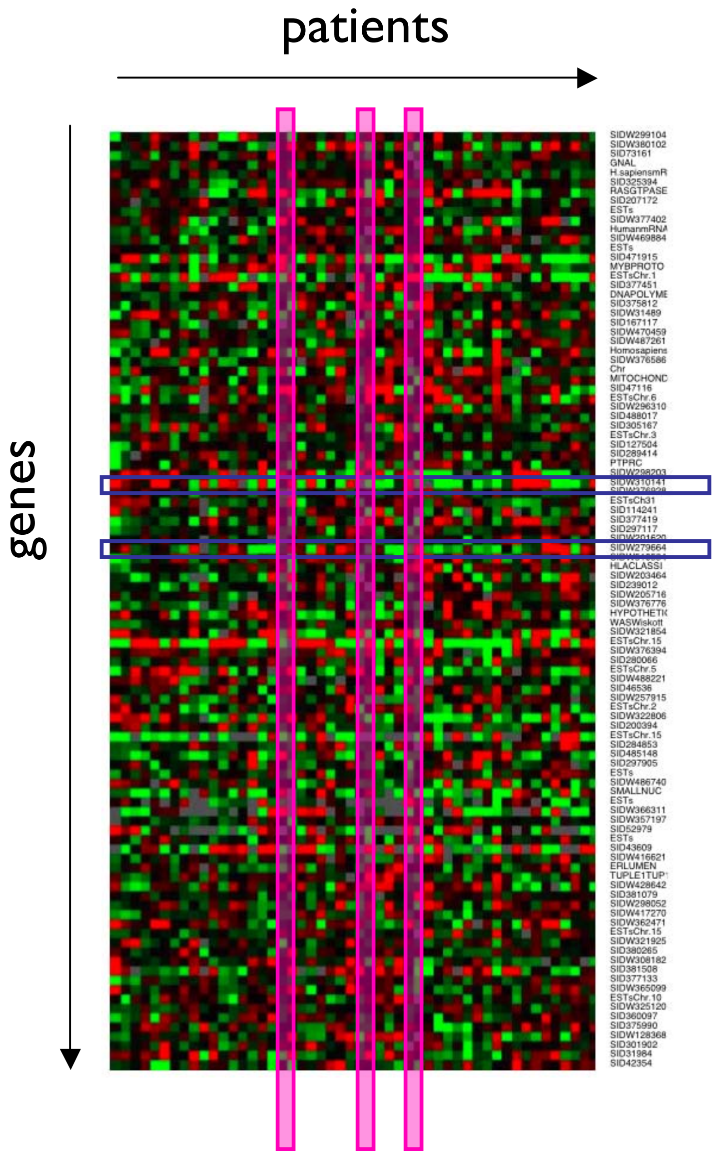
Similarity



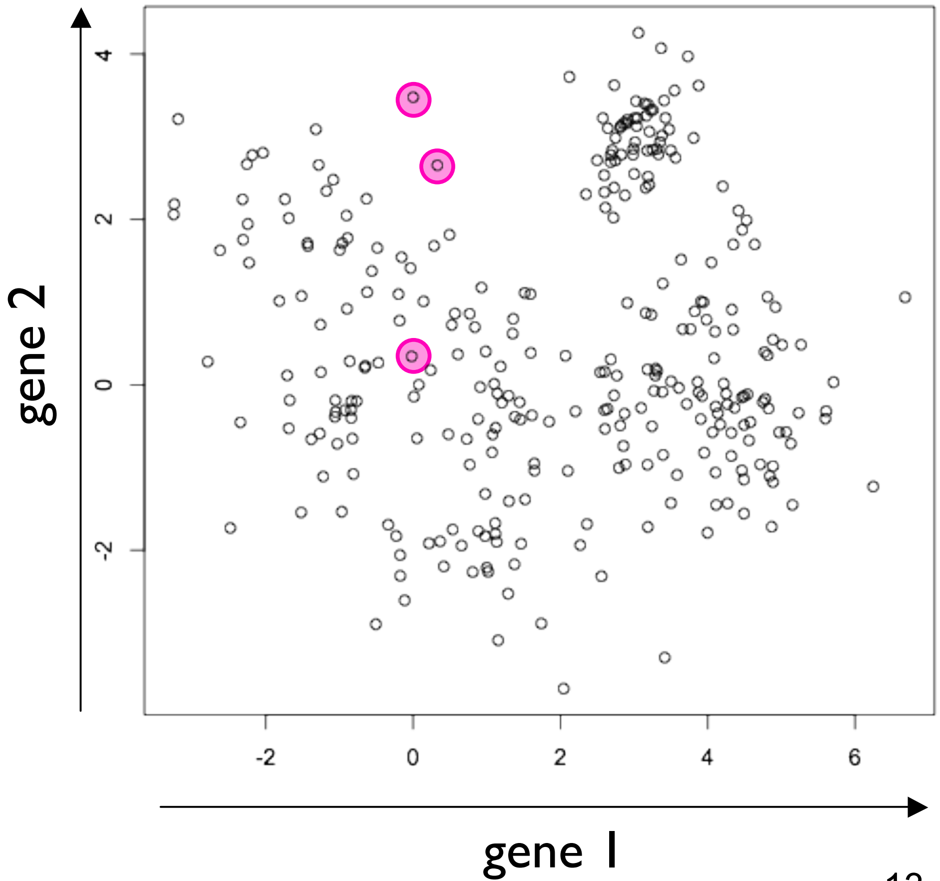
- n objects (here, patients)
- p parameters (here, genes)



Similarity



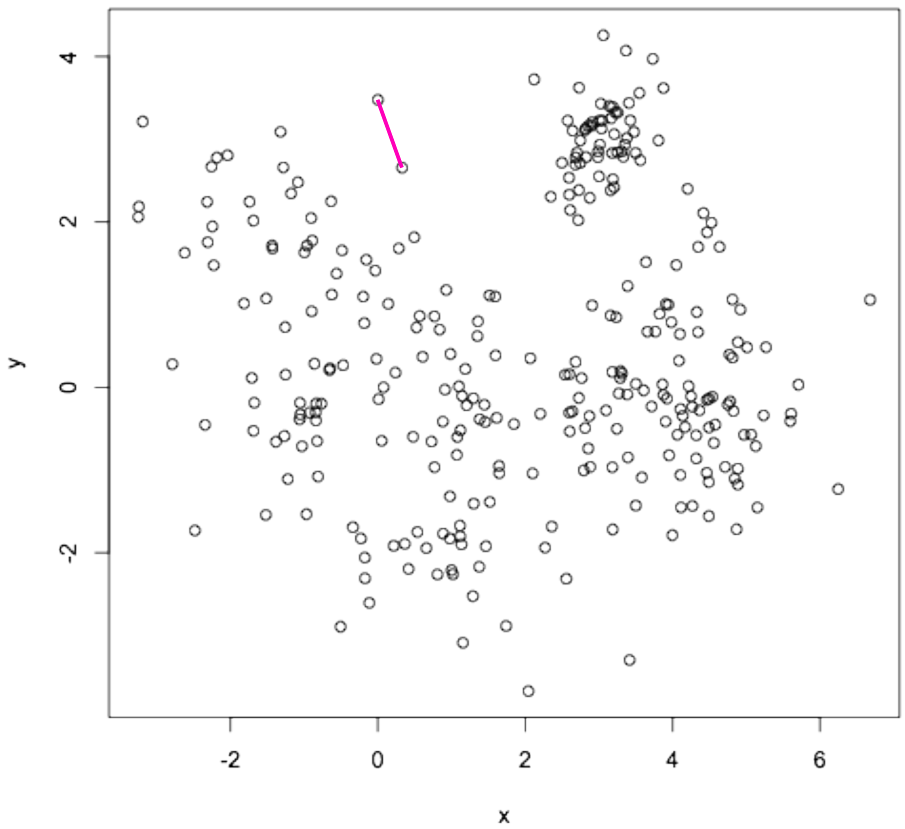
- n objects (here, patients)
- p parameters (here, genes)



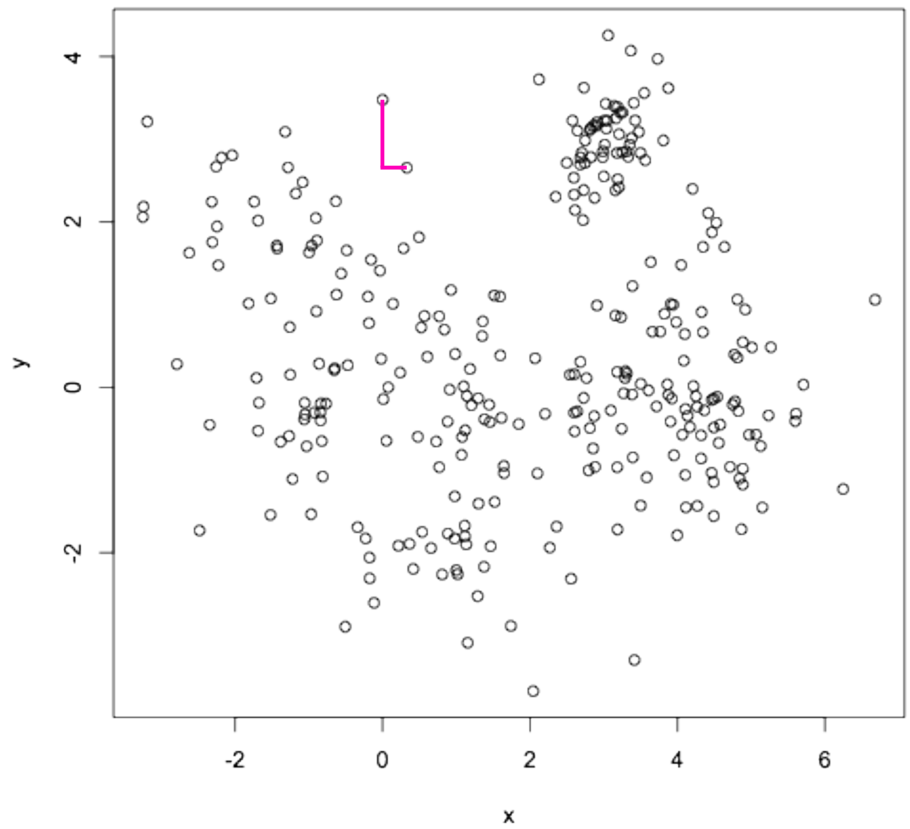
Many similarity measures

Dissimilarity measures

- L^2 distance (Euclidean distance)

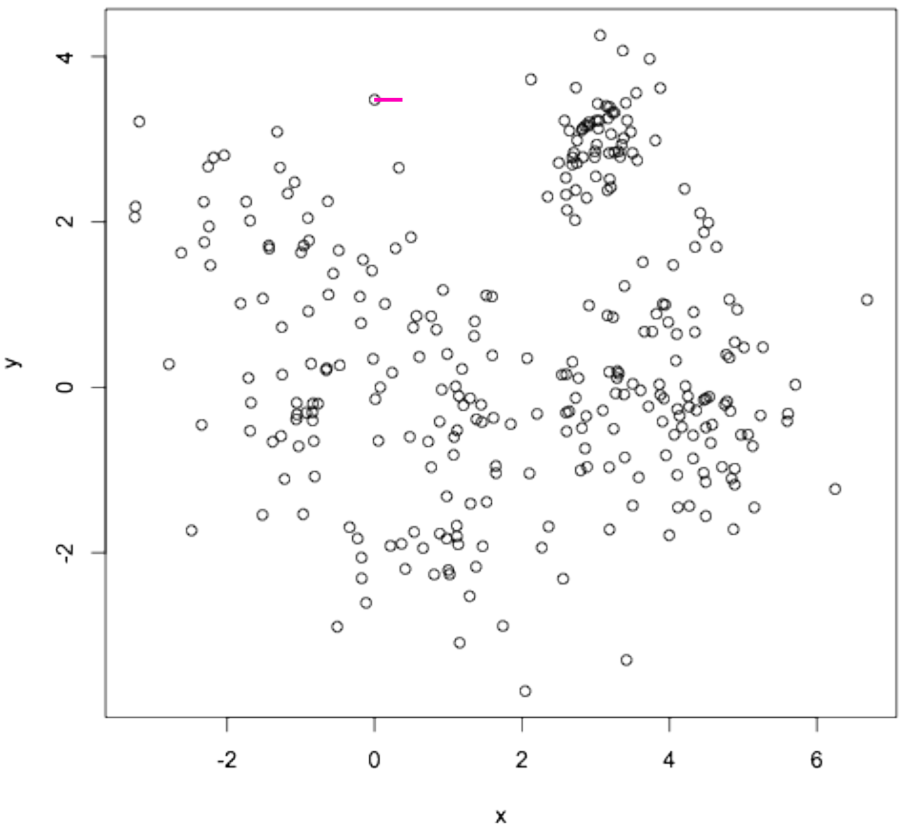


- L^1 distance (Manhattan distance)



Dissimilarity measures

- Weighted L^2 distance



- 1 - Pearson correlation

Dissimilarity measures

- L^p family
  - L^1 type: $d(x, y) = \Sigma_i |x_i - y_i|$
  - L^2 type: $d(x, y) = \text{sqrt}(\Sigma_i (x_i - y_i)^2)$
  - More generally, L^p type: $d(x, y) = (\Sigma_i |x_i - y_i|^p)^{1/p}$
  - Metrics: positive, symmetric, triangle inequality
- Transformations
  - Transformation of covariates with f and computation of d(f(x), f(y))
  - f could be a log, a normalization method, a weighting function
  - Example, weighted Euclidean: $d(x, y) = \text{sqrt}(\Sigma_i w_i(x_i - y_i)^2)$
  - Example, Mahalanobis distance: $d(x, y)^2 = (x-y)^t A(x-y)$, with $A = \Sigma^{-1}$

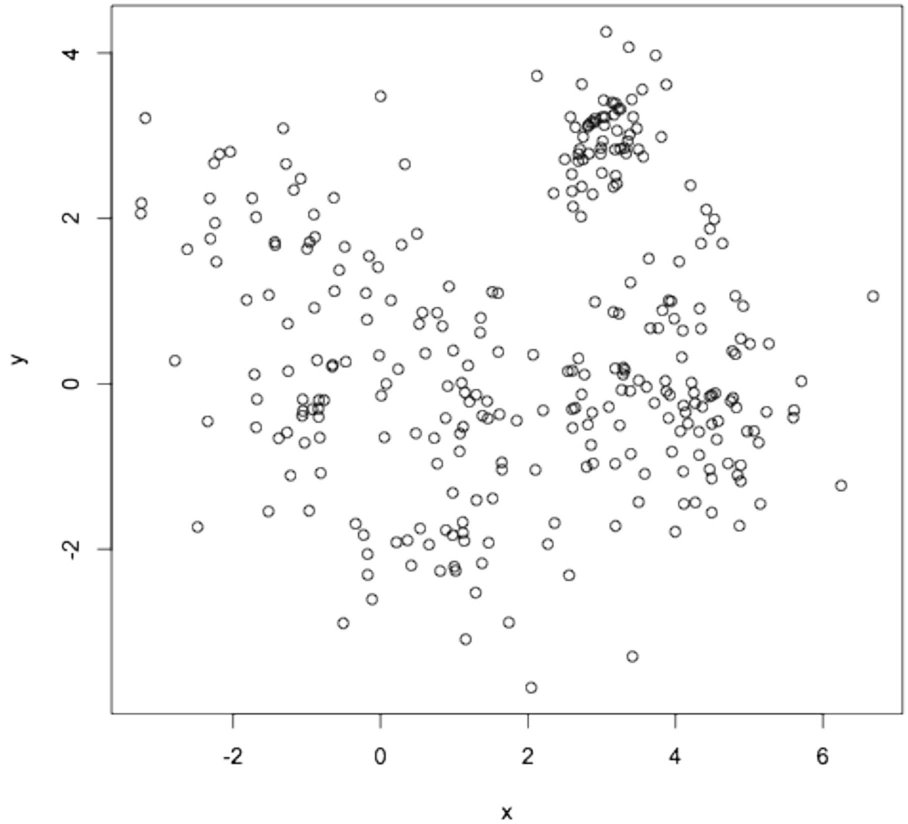
Which dissimilarity measure ?

- No universal solutions: it all depends on the objects
- L^1 distance is less sensitive to outliers
- L^2 distance is more sensitive
- If the object parameters have similar distributions
  - Ex: gene expression after normalization
  - Correlation distance is a popular choice

- If not, object parameters have to be transformed
  - Ex: heterogeneous parameters (cellular phenotypes)
  - Cell A: (size=120, ecc=0.3, x.position=134)
  - Cell B: (size=90, ecc=0.5, x.position=76)
  - Cell C: (size= 140, ecc=0.4, x.position=344)
  - Ex: un-normalized gene expression sets

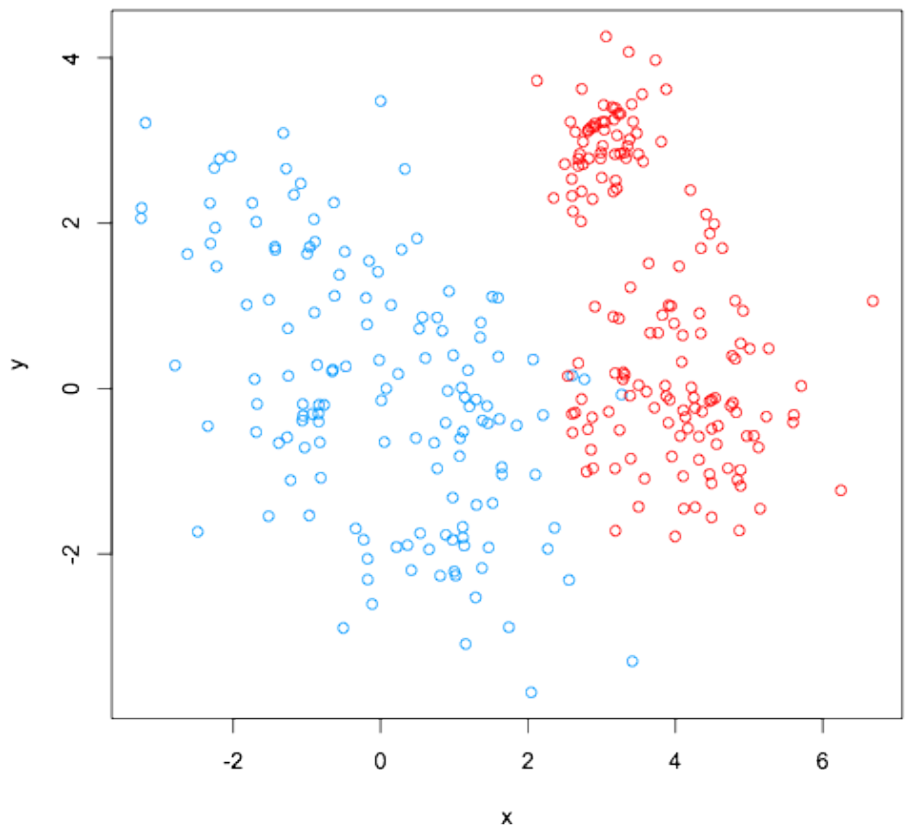
Clustering

- " Identification of similar subgroups within data "
- Ill-defined problem ⇒ many algorithms exist
  - Tradeoffs between agglomerative properties, sensitivity, robustness, speed



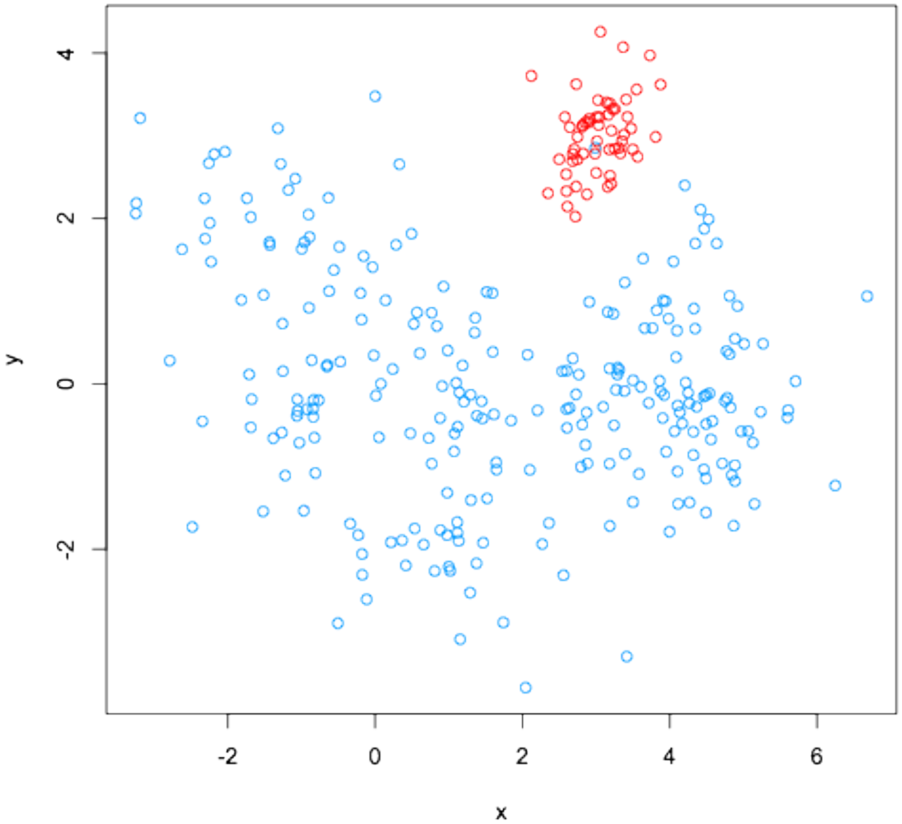
Clustering

- " Identification of similar subgroups within data "
- Ill-defined problem ⇒ many algorithms exist
  - Tradeoffs between agglomerative properties, sensitivity, robustness, speed



Clustering

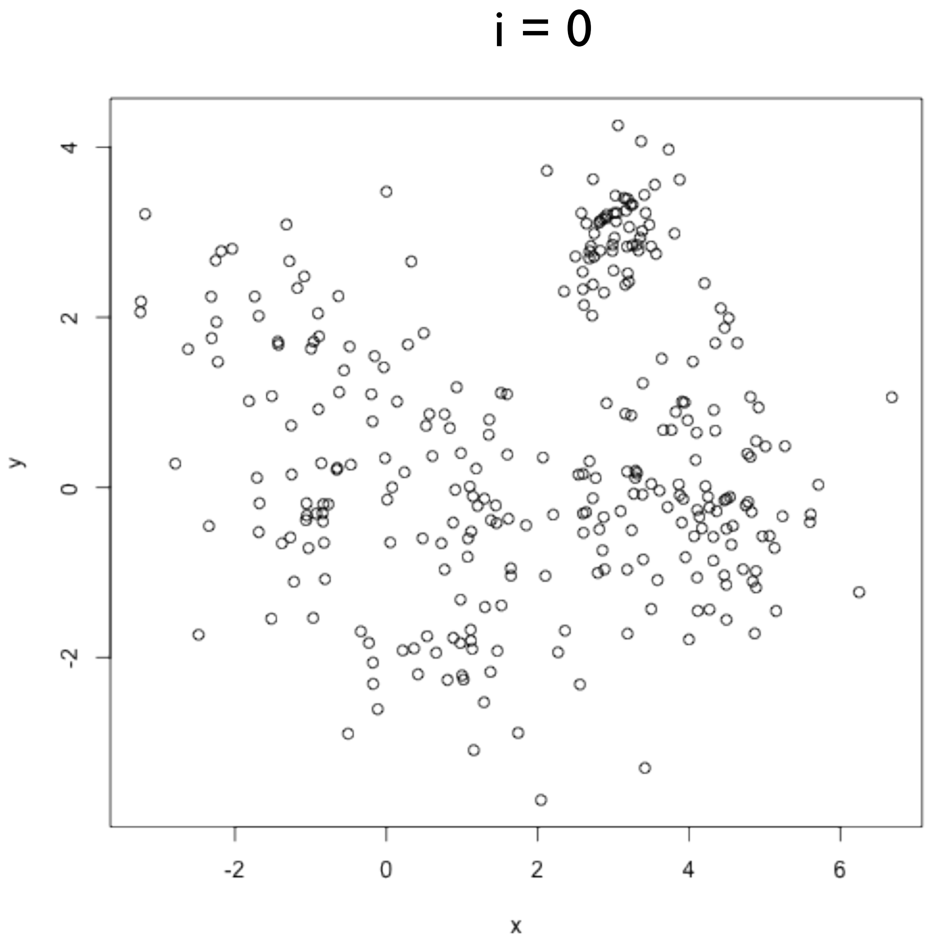
- " Identification of similar subgroups within data "
- Ill-defined problem ⇒ many algorithms exist
  - Tradeoffs between agglomerative properties, sensitivity, robustness, speed



Hierarchical clustering

- Iterative agglomerative method
- Initialisation: each data point is assigned to an unique cluster
- At each step: join most similar clusters, using between cluster dissimilarity measure
- Iterate until there is only one cluster

- Several linkage variants
- In R, function hclust

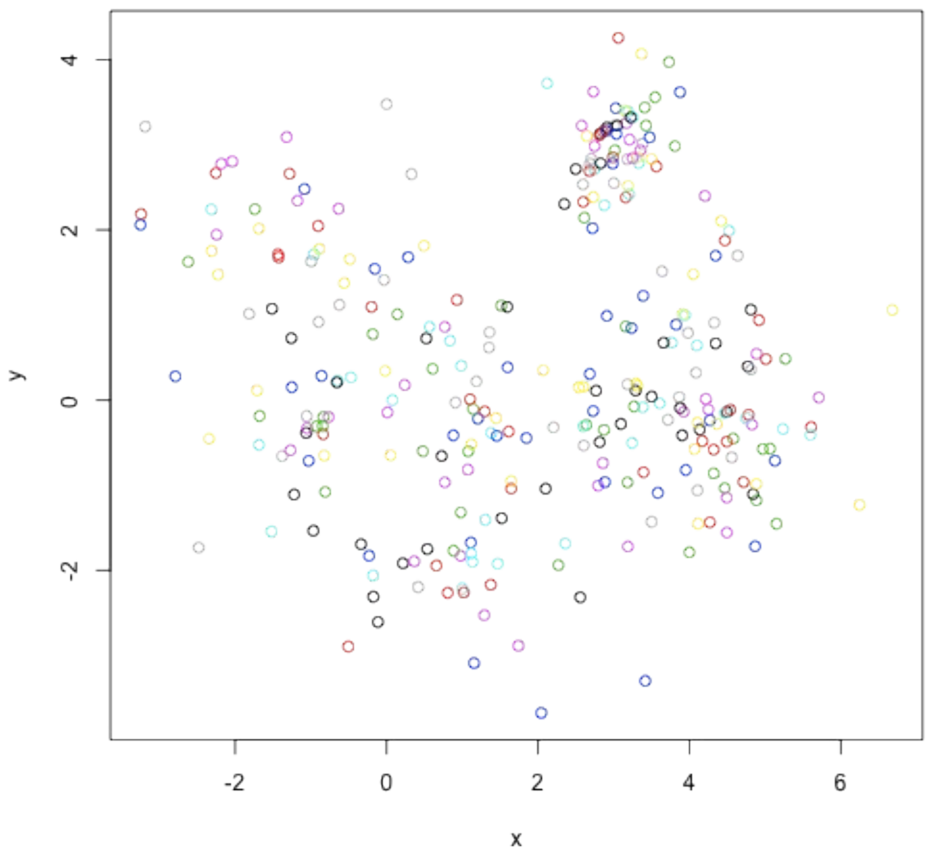


Hierarchical clustering

- Iterative agglomerative method
- Initialisation: each data point is assigned to an unique cluster
- At each step: join most similar clusters, using between cluster dissimilarity measure
- Iterate until there is only one cluster

- Several linkage variants
- In R, function hclust

i = 300

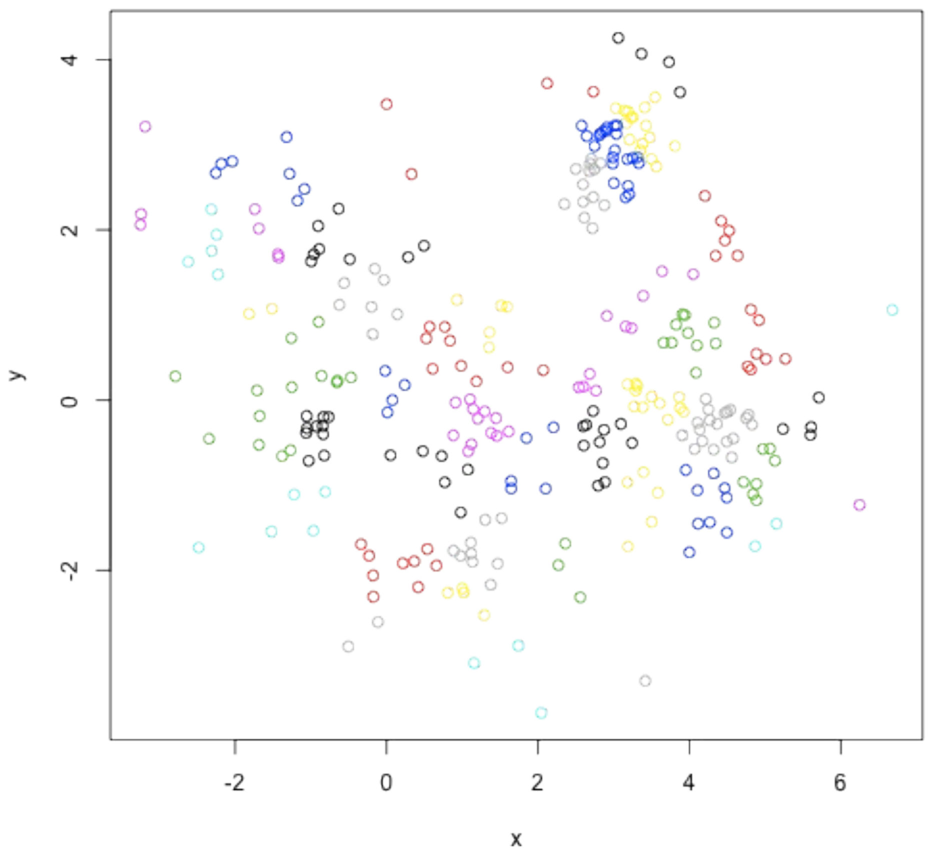


Hierarchical clustering

- Iterative agglomerative method
- Initialisation: each data point is assigned to an unique cluster
- At each step: join most similar clusters, using between cluster dissimilarity measure
- Iterate until there is only one cluster

- Several linkage variants
- In R, function hclust

i = 50

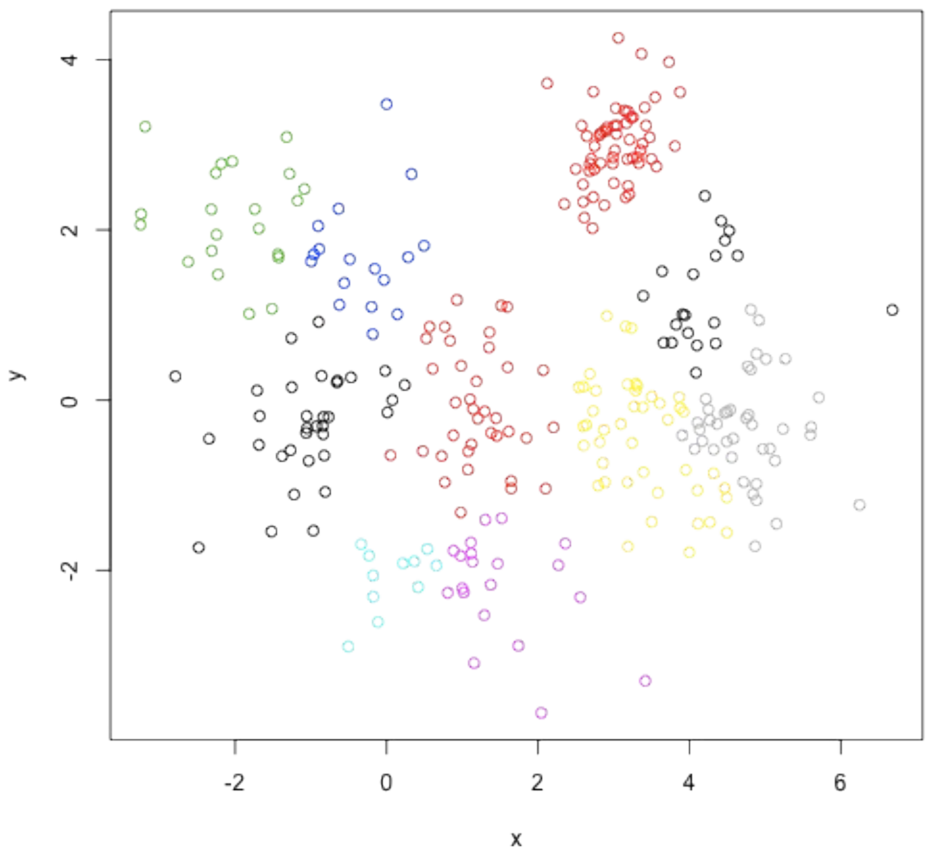


Hierarchical clustering

- Iterative agglomerative method
- Initialisation: each data point is assigned to an unique cluster
- At each step: join most similar clusters, using between cluster dissimilarity measure
- Iterate until there is only one cluster

- Several linkage variants
- In R, function hclust

i = 10

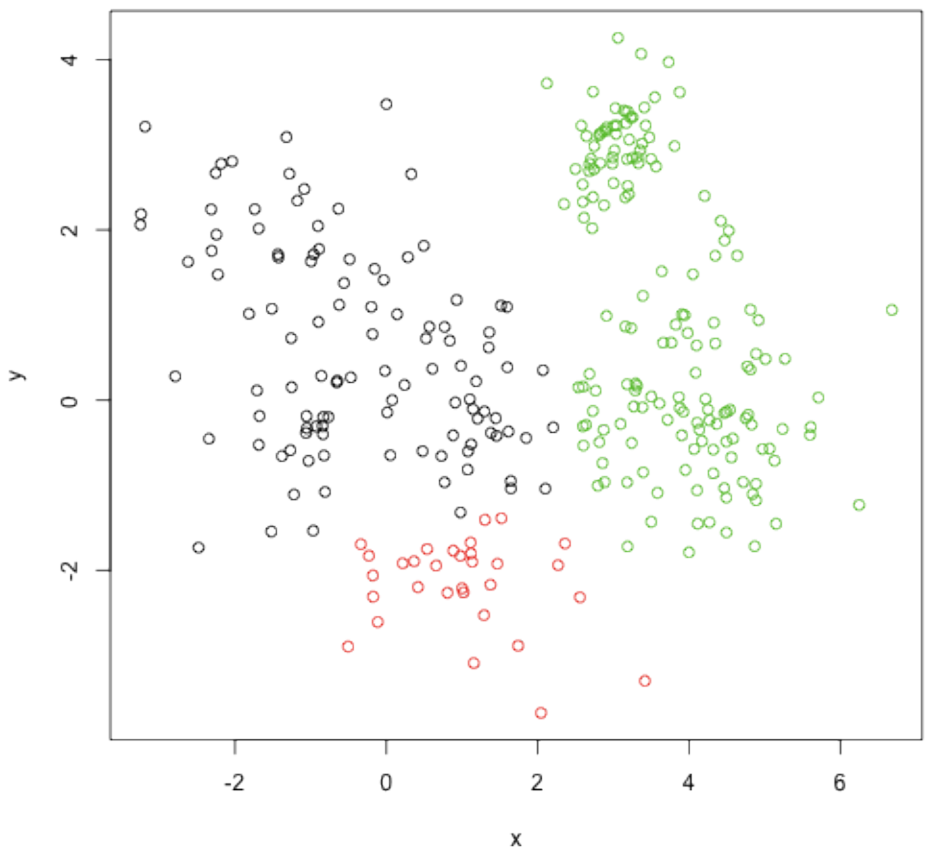


Hierarchical clustering

- Iterative agglomerative method
- Initialisation: each data point is assigned to an unique cluster
- At each step: join most similar clusters, using between cluster dissimilarity measure
- Iterate until there is only one cluster

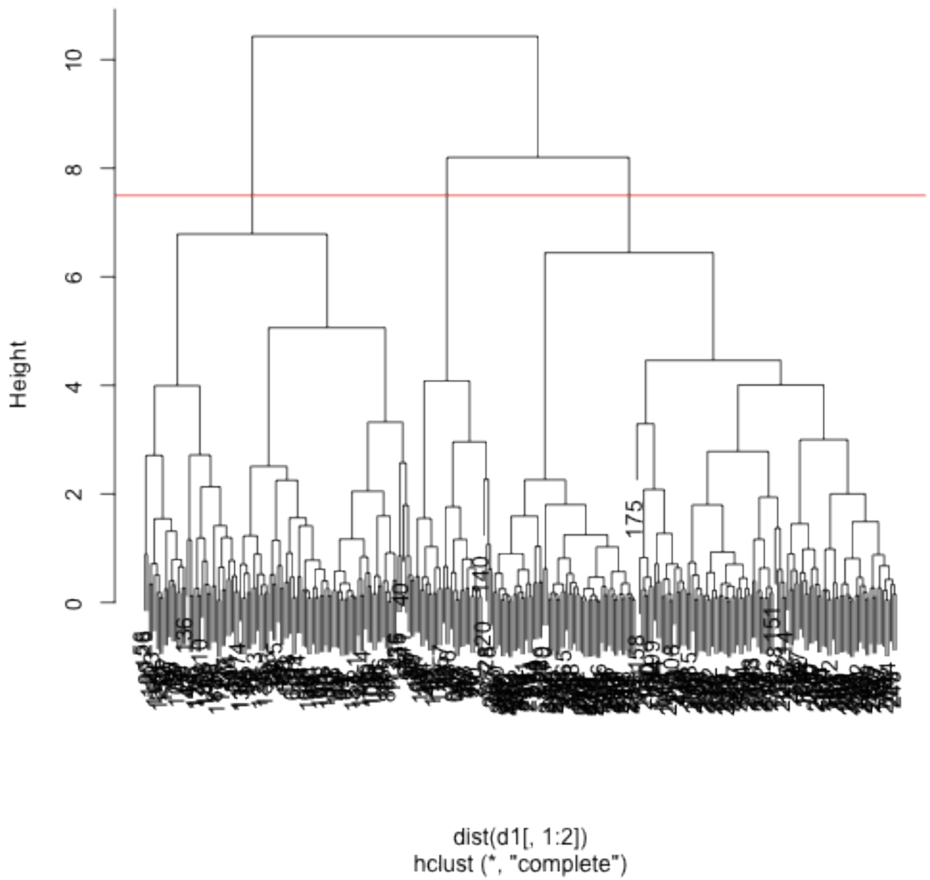
- Several linkage variants
- In R, function hclust

i = 3

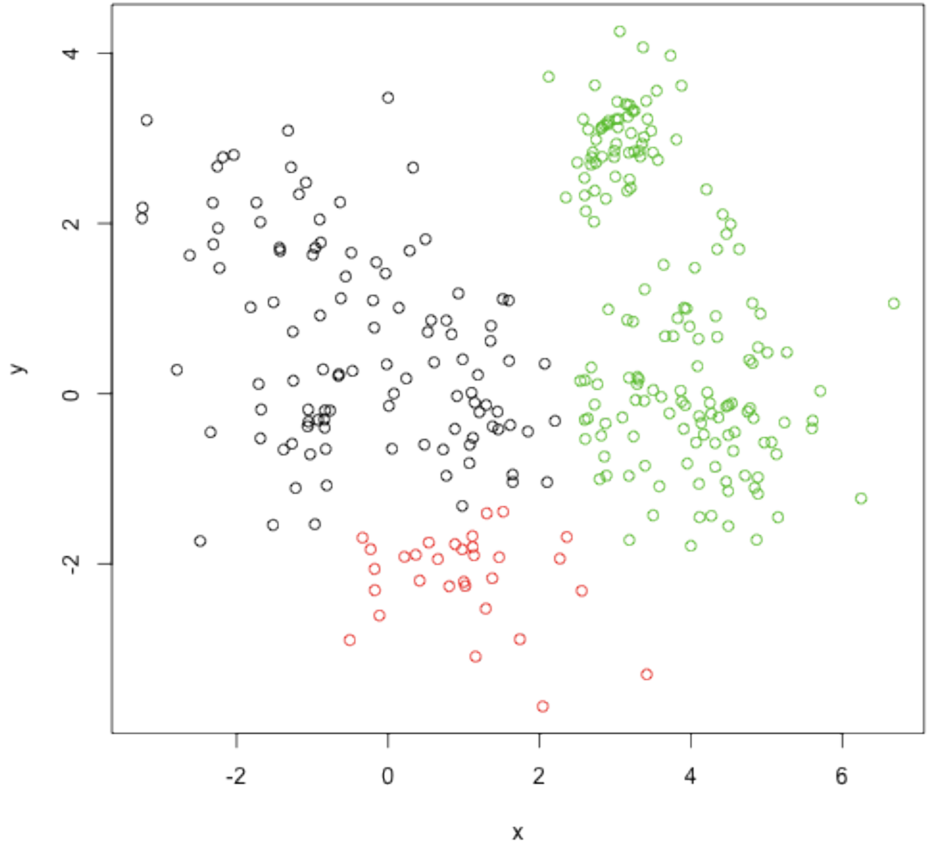


Hierarchical clustering

- Clustering data dendrogram

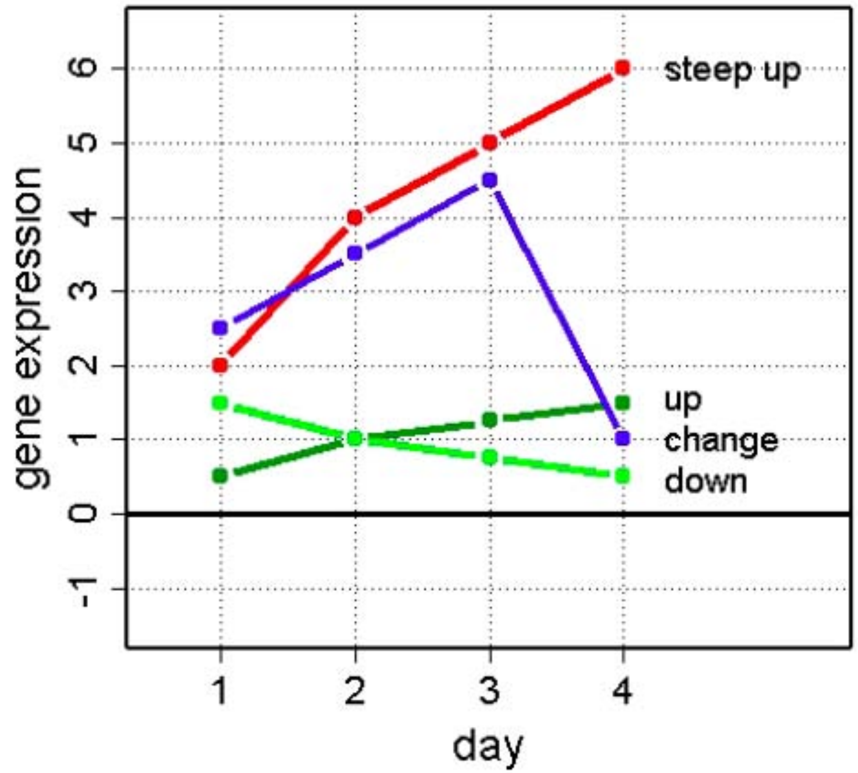


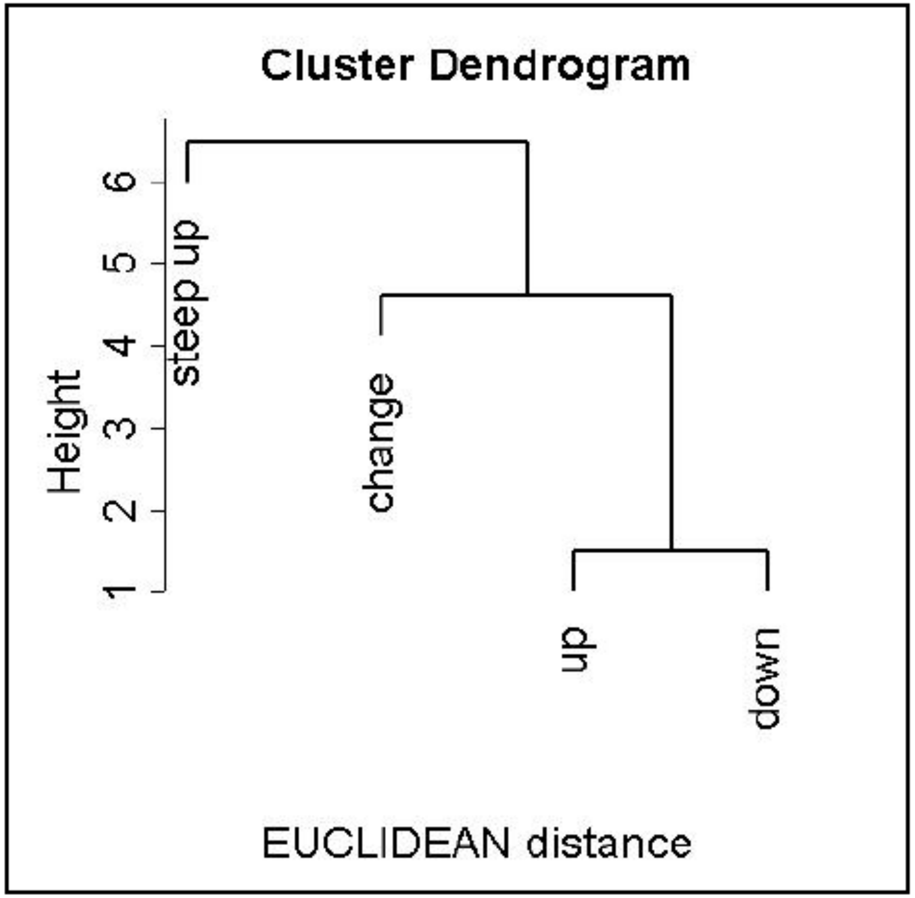
i = 3



Examples

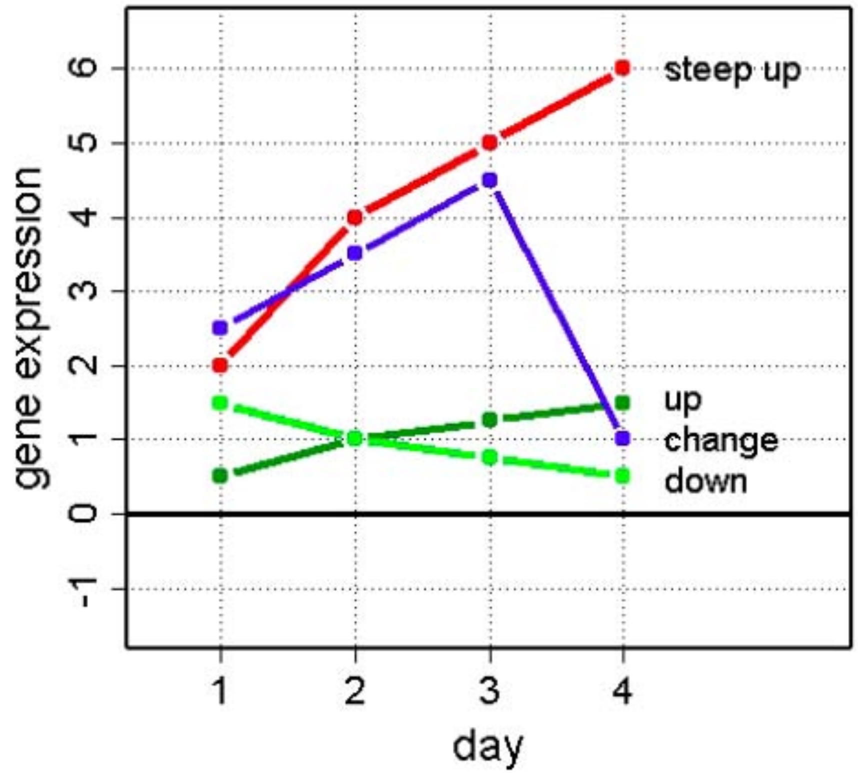
- Gene expression time series

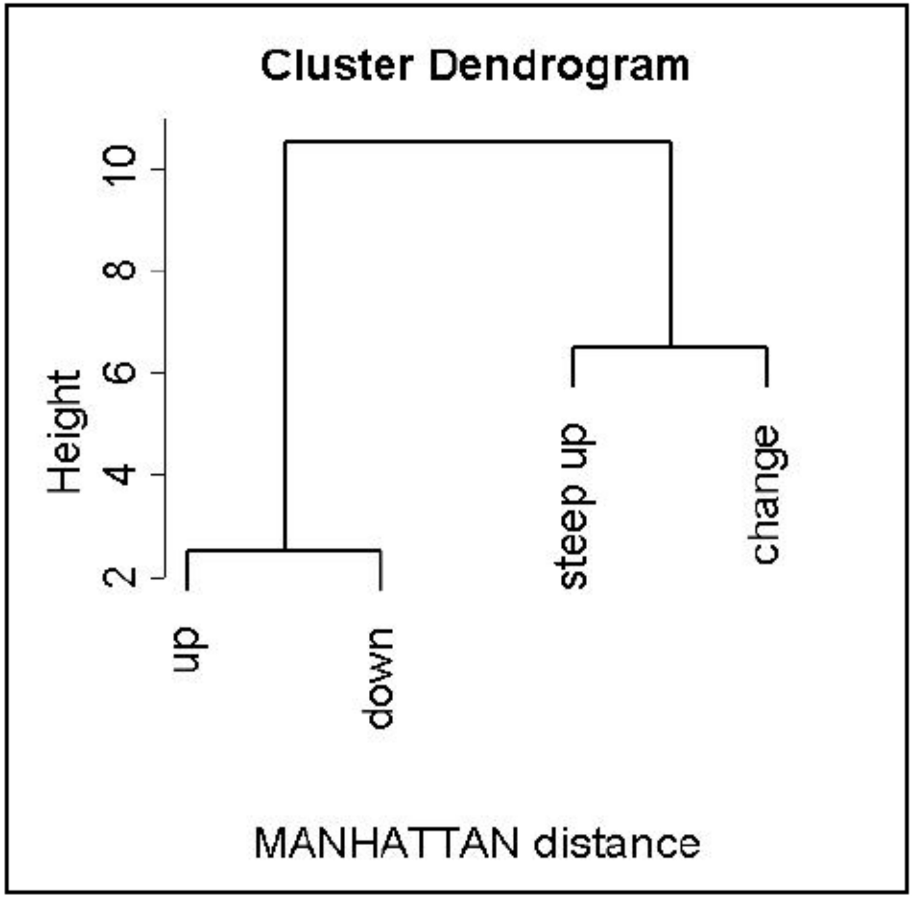




Examples

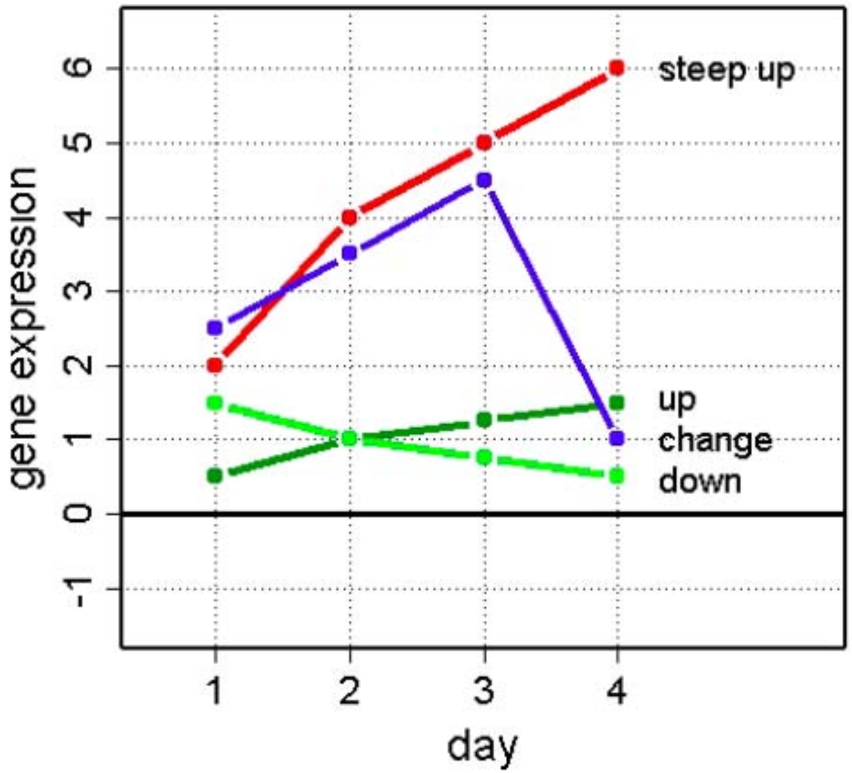
- Gene expression time series

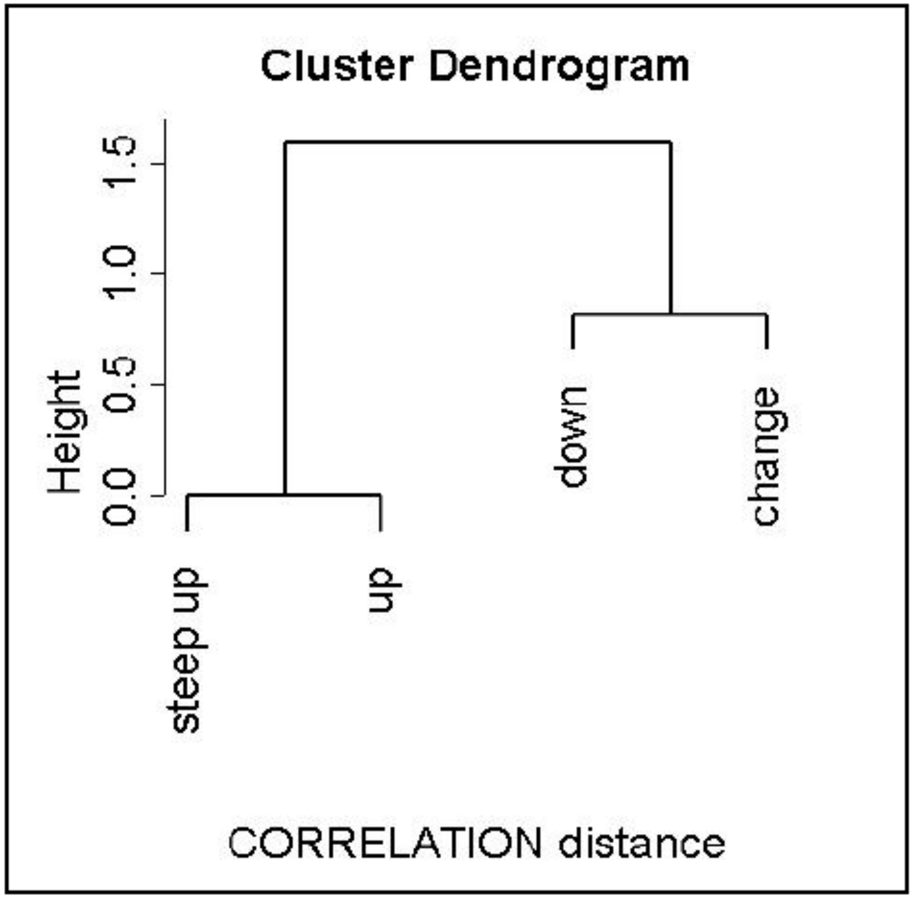




Examples

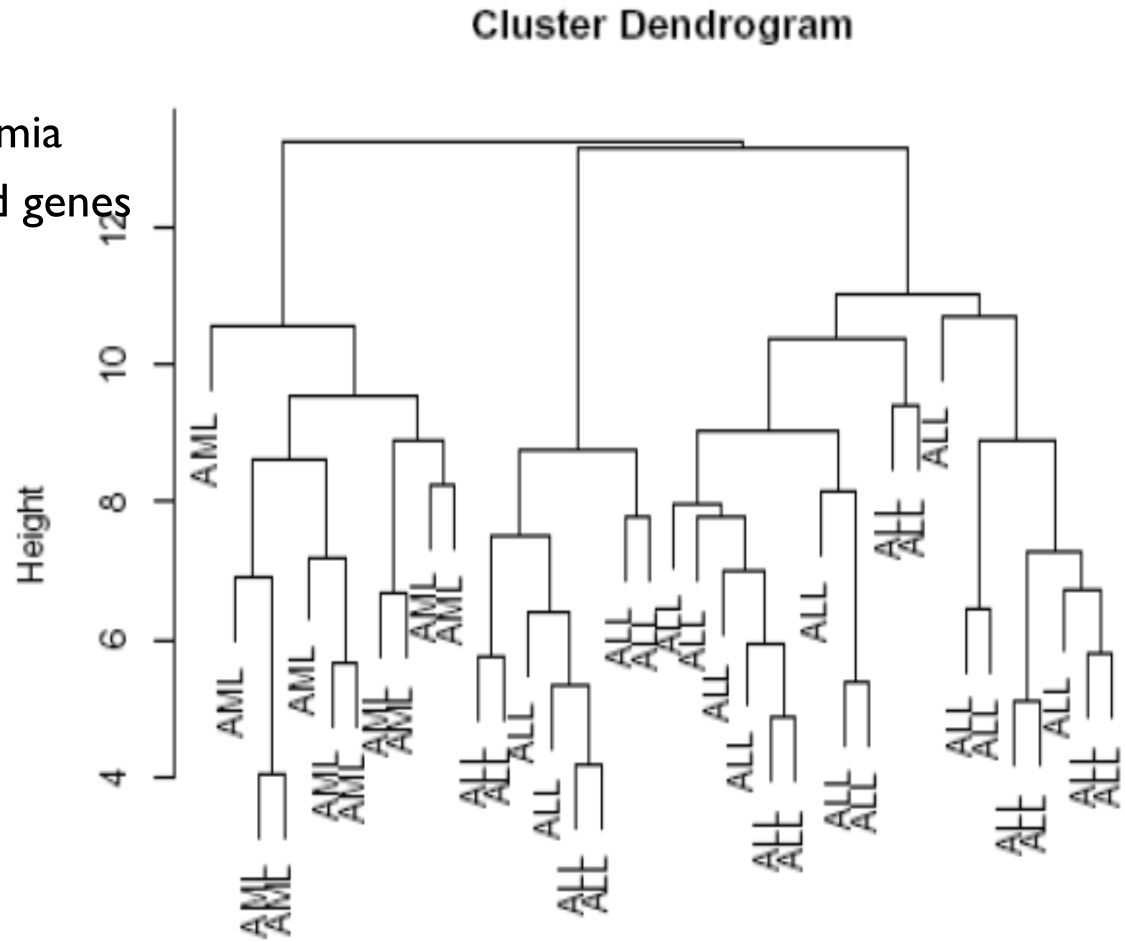
- Gene expression time series





Golub et al. leukemia dataset

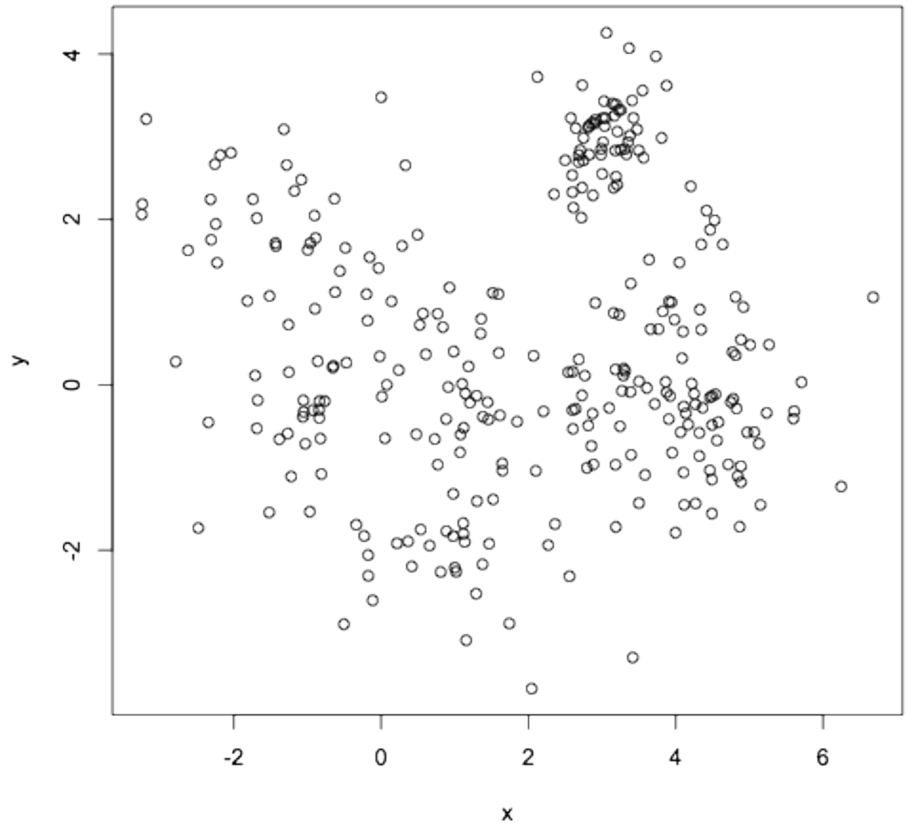
- Gene expression data of
  - 25 acute myeloid leukemia
  - 47 acute lymphoblastic leukemia
  - Using 400 most differentiated genes
- Perfect separation



k-means

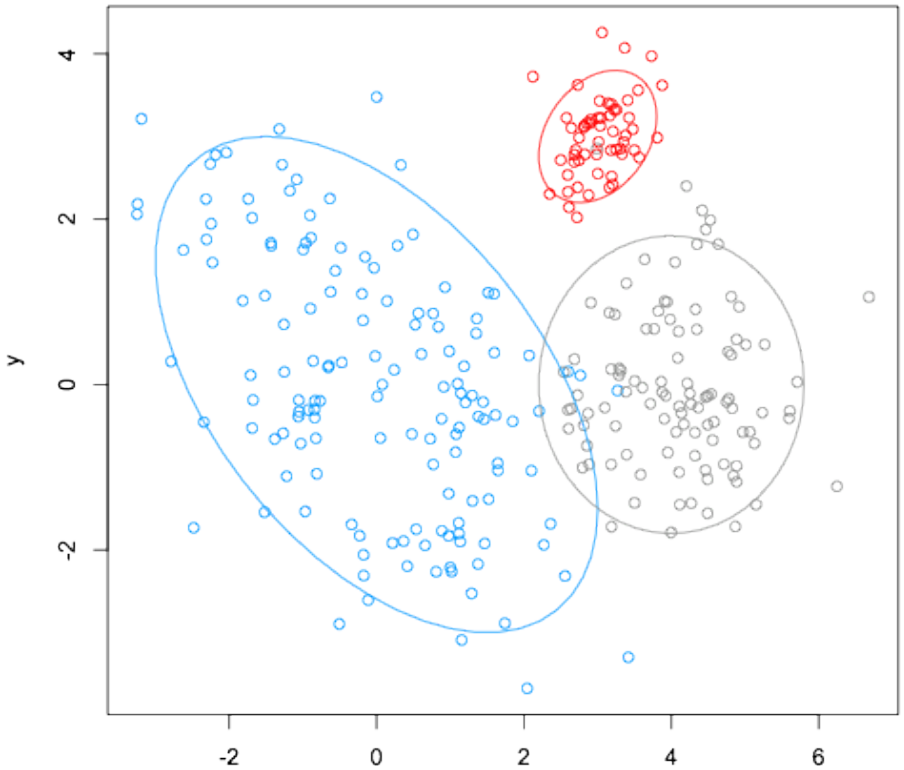
- Iterative partitioning method
- Initialisation: *k* random clusters
- Assignment: each point is assigned to its closest cluster center
- Update: cluster centers are updated with the new members
- Iterate through convergence

- In R, function kmeans
- Known number of clusters



Gaussian mixture estimation

- Well-defined estimation problem
  - Data is believed to come from a mixture of *k* Gaussian distributions
  - $X \sim \omega\mathcal{N}(\mu_1, \Sigma_1) \oplus (1-\omega)\mathcal{N}(\mu_2, \Sigma_2)$



- EM algorithm
  - Expectation: Given parameter estimates, compute class membership probability
  - Maximization: Given class membership, estimate parameters by maximum likelihood
  - Iterate through convergence

- Works well if n >> p
  - Package mclust

Clustering

- Ill-defined problem $\Rightarrow$ many algorithms exist
- Most important: a relevant dissimilarity measure
- Requires cautious interpretation
- Still useful tool for data exploration
- Prior knowledge (model, dissimilarity measure) should be used, if available

Clustering phenotype populations by genome-wide RNAi and multiparametric imaging

Gregoire Pau, Oleg Sklyar, Wolfgang Huber
EMBL, Heidelberg

Florian Fuchs, Dominique Kranz, Christoph Budjan,
Thomas Horn, Sandra Steinbrink, Angelika Pedal, Michael Boutros
DKFZ, Heidelberg

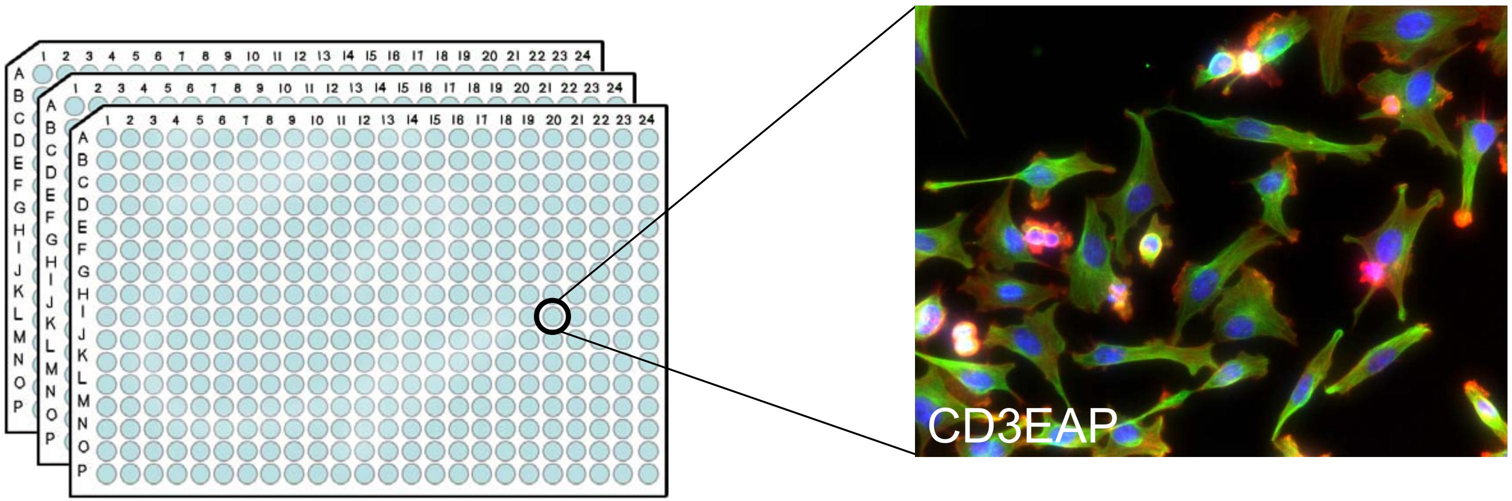
Molecular Systems Biology, 2010

EMBL-EBI

dkfz. GERMAN CANCER RESEARCH CENTER
IN THE HELMHOLTZ ASSOCIATION

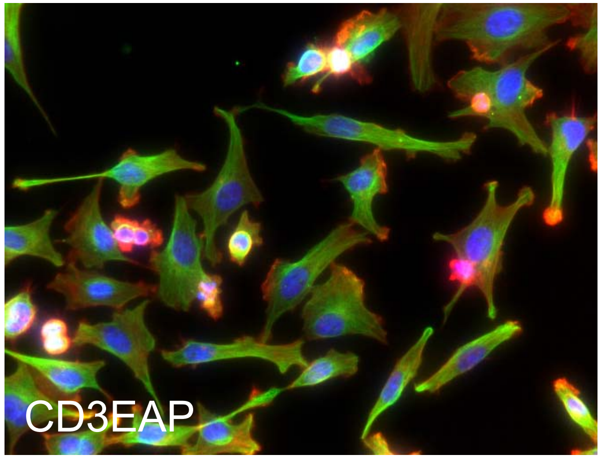
Experimental setup

- Human cervix carcinoma HeLa cells
- Genome-wide RNAi screen, testing 22839 genes
- Cells are incubated for 48 h and fixed
- Staining using DNA (DAPI), Tubulin (Alexa), Actin (TRITC)
- Readout: microscopy images



Phenotypic profile

- Phenotype expressed by a population of cells
- Phenotypic profile, vector of p = 13 parameters
  - Number of cells
  - Statistics on cell features (size, eccentricity, …)
  - Cell types distribution (normal, metaphase, condensed, protruded…)

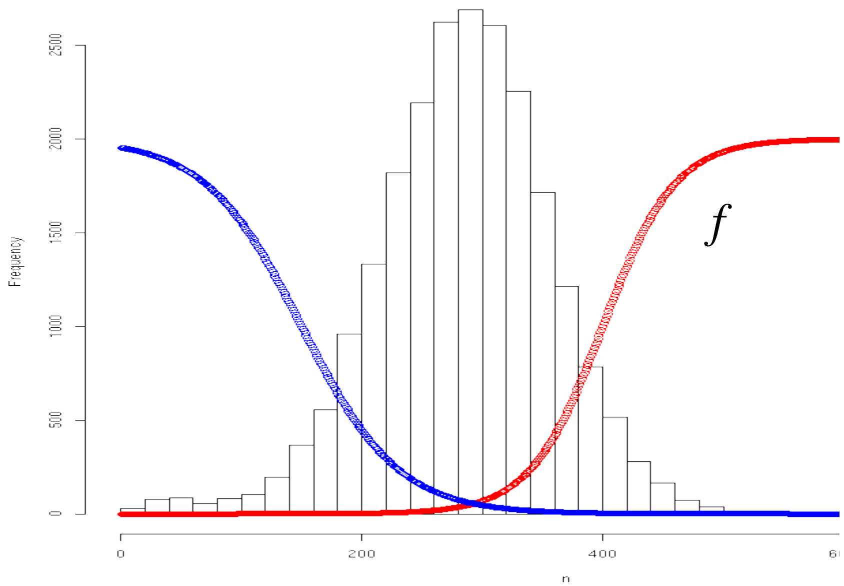


x =

n	289
ext	34.33118
ecc	0.472934
Next	2857.356
Nint	485.2710
a2i	0.828876
Next2	0.098647
AF %	0.049594
BC %	0.081746
C %	0.158817
M %	0.179339
LA %	0.009249
P %	0.219697

Transformation of phenotypic descriptors

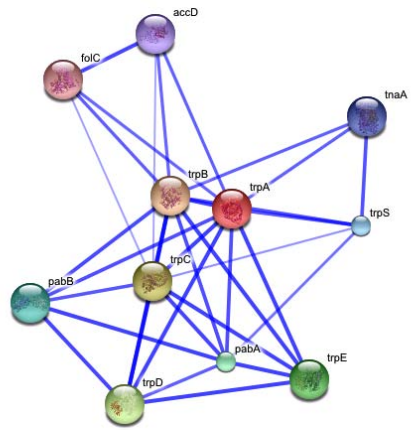
- Let x be a phenotypic profile in R^P
- Transformation into a phenoprint, vector of [0,1] scores
  - For each descriptor k, $f(x_k) = 1 / (1 + \exp(-\alpha_k(x - \beta_k))$
- Phenotypic distance = L^1 distance between phenoprints
- 20 parameters $(\alpha, \beta)_k$ to be determined



Distance metric learning

- Perturbation of related genes lead more likely to similar phenotypes than random ones

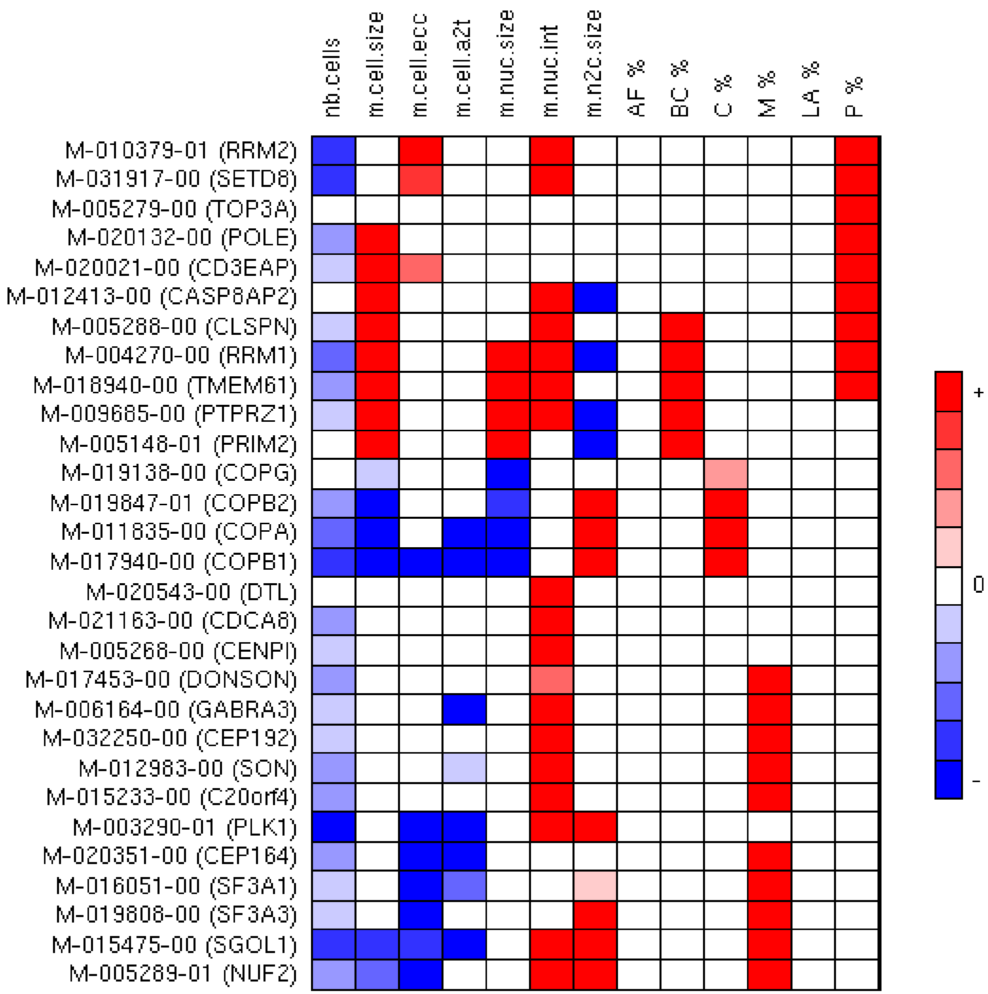
- EMBL STRING database
  - About 6,000,000 related protein pairs
  - Physical interaction, regulation, literature co-citation
  - Rich but noisy



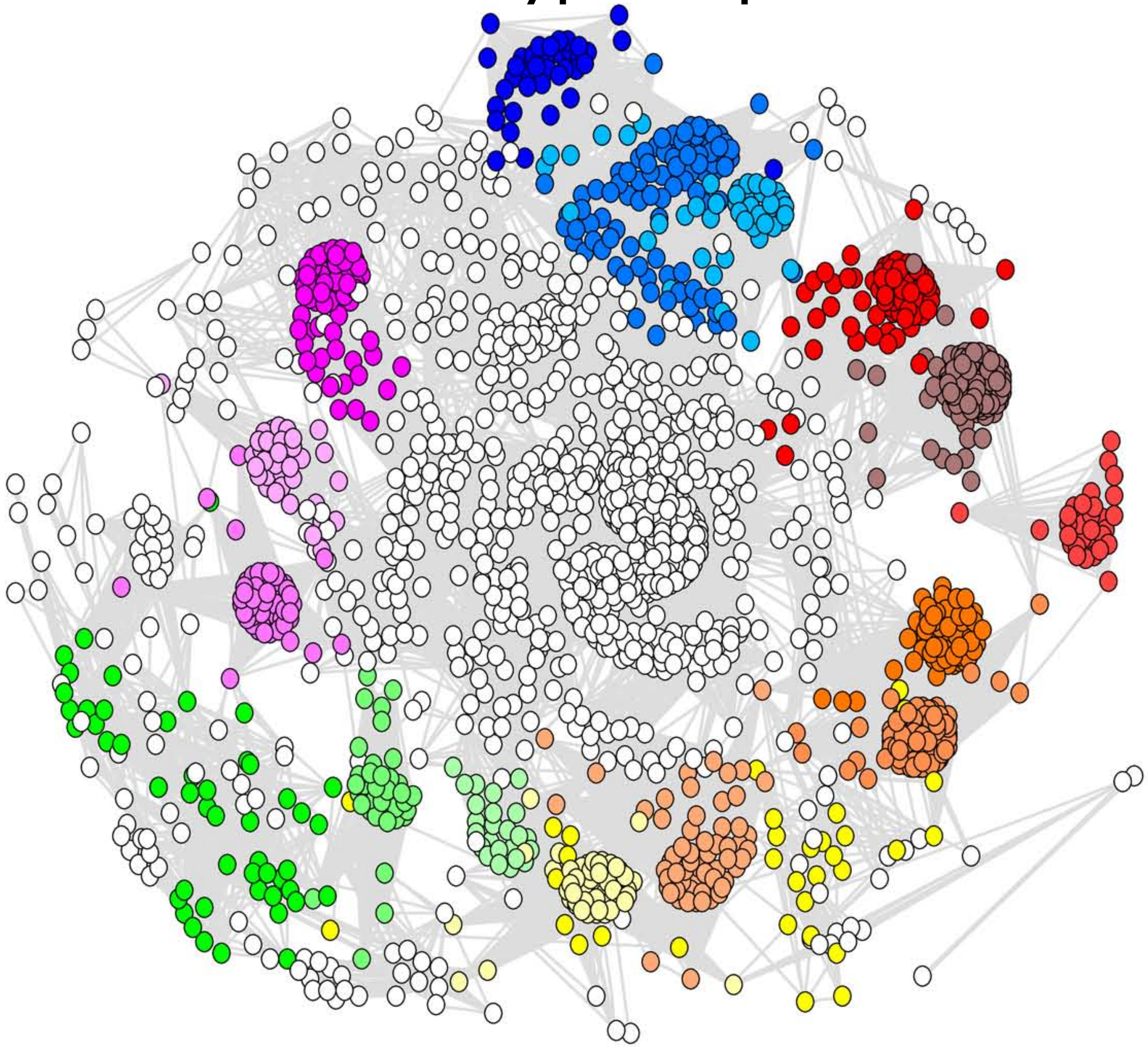
- We design our distance to be lower in average on related gene pairs than random ones
  - Parameters $(\alpha, \beta)_k$ are fitted by minimization of a criterion
  - Similar to PAM matrices to compute protein alignment scores

Phenotypic hits

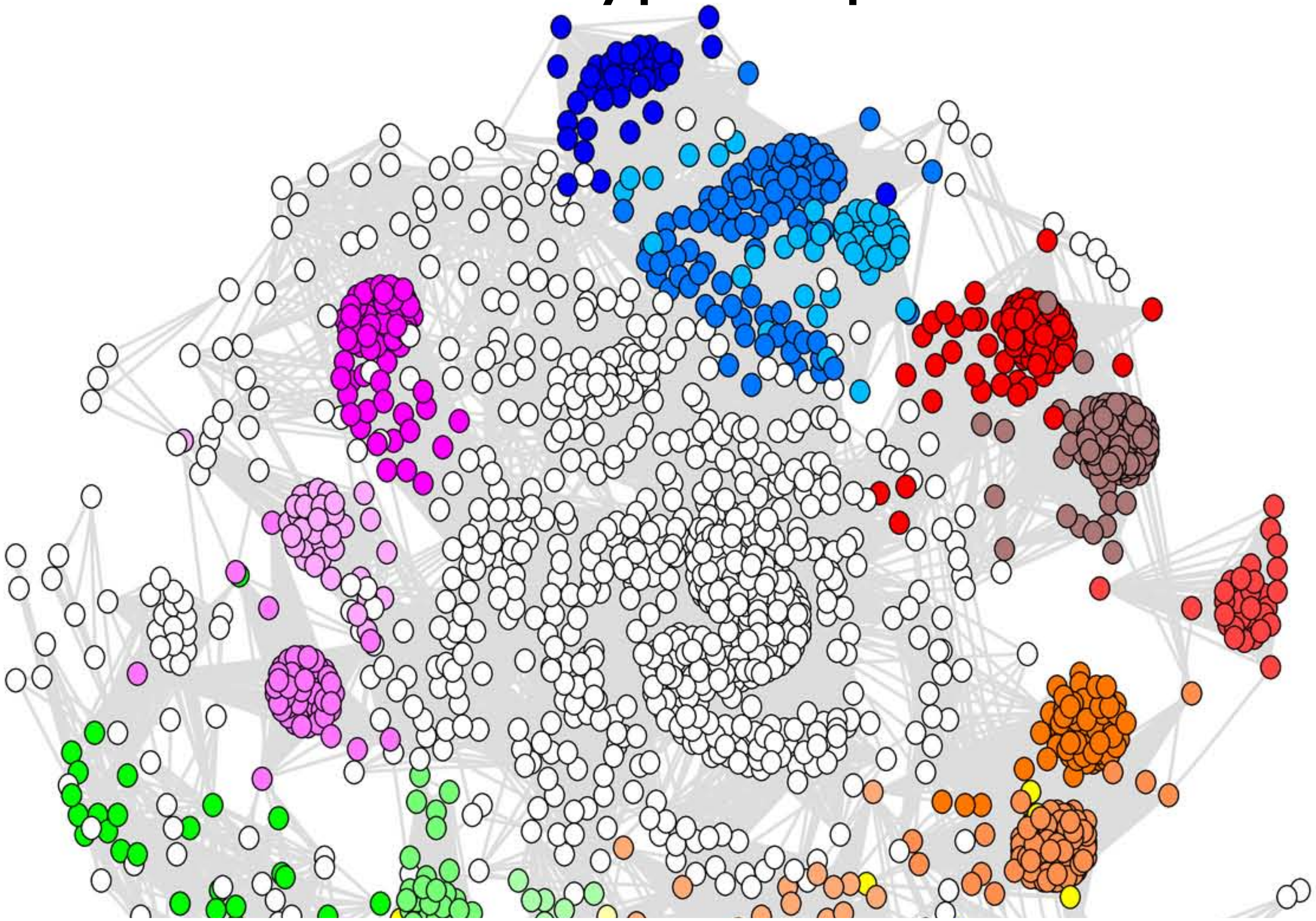
- 1820 perturbations show non-null phenoprints



Phenotypic map

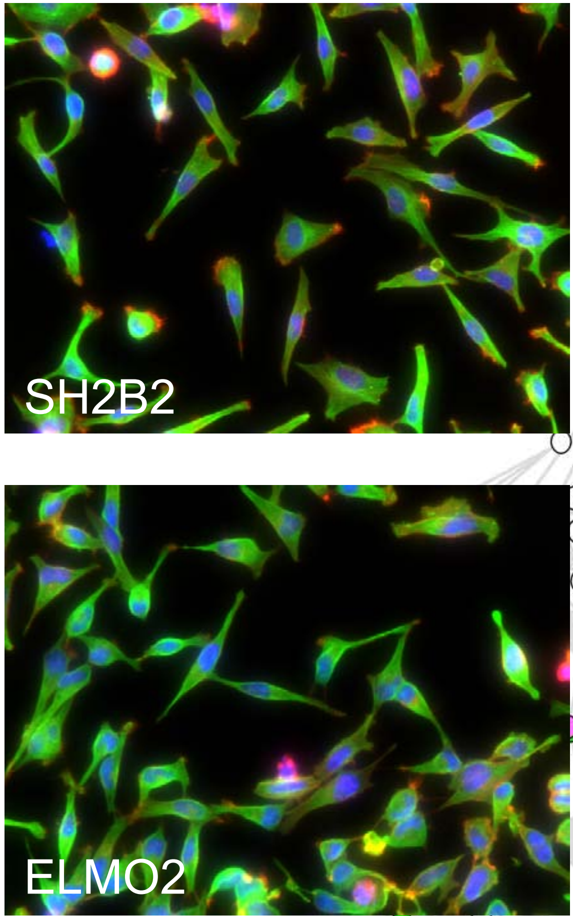


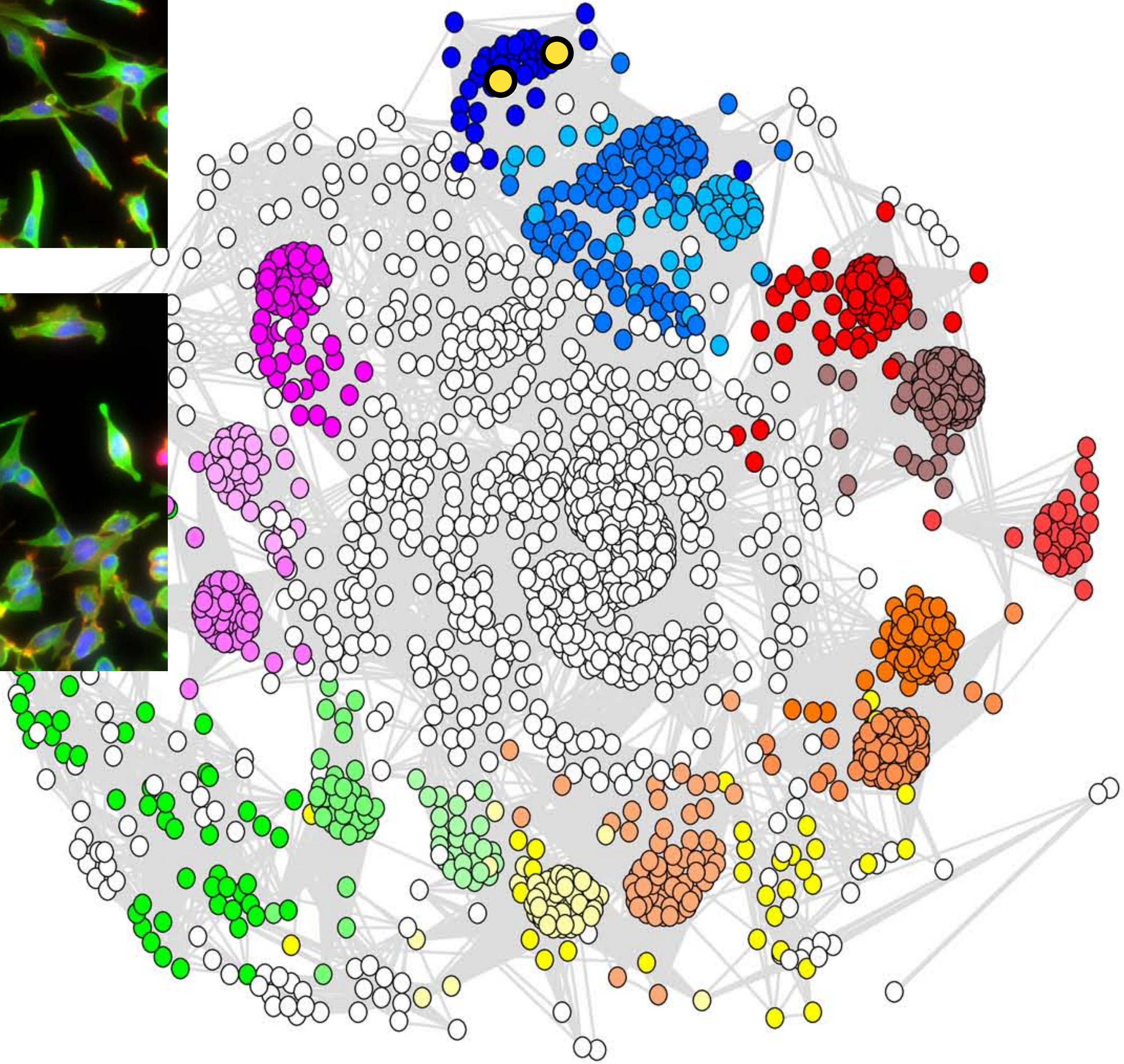
Phenotypic map



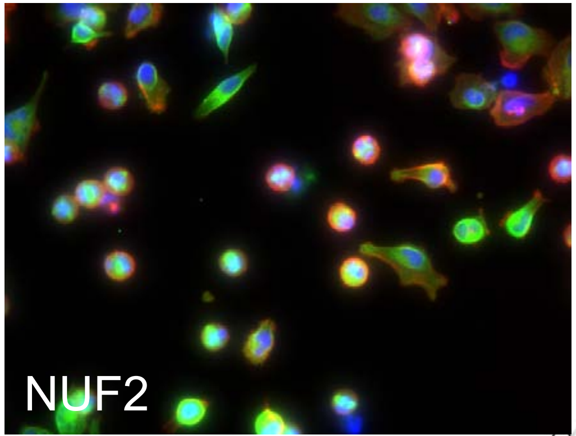
- BL phenotype
- Bright nuclei
- Large nuclei
- Cells with protrusions
- Elongated cells
- Elong. cells with protrusions
- SM phenotype
- Small cells
- Low eccentricity cells
- High actin ratio cells
- Metaphase cells
- Other phenotype
- Actin fiber cells
- Big cells
- Large cells
- Lamellipodia cells
- Lamell. + high actin ratio cells
- Proliferating cells

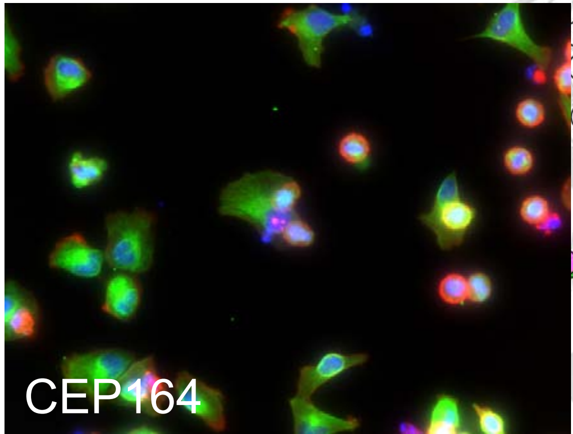
Elongated phenotype

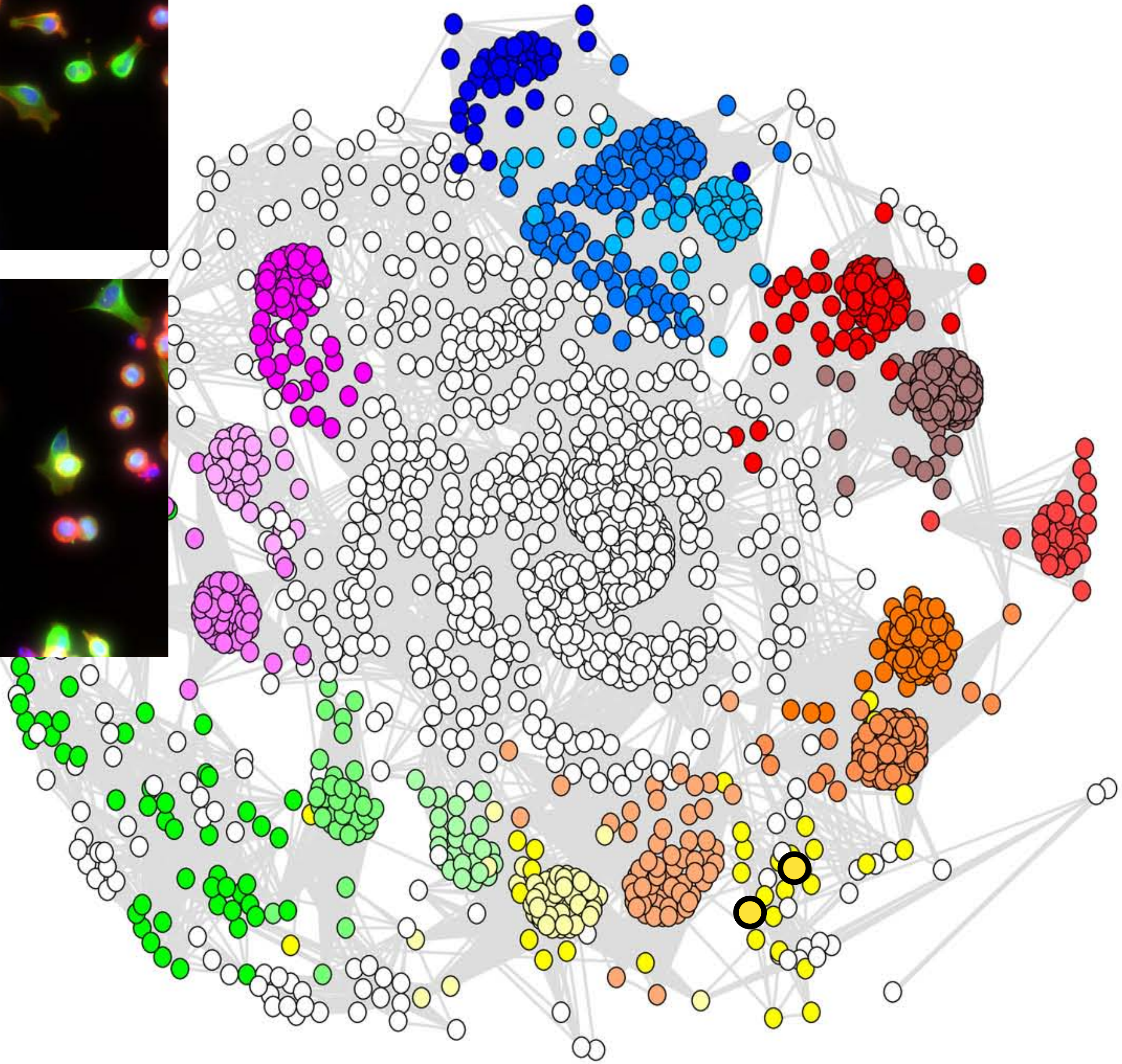


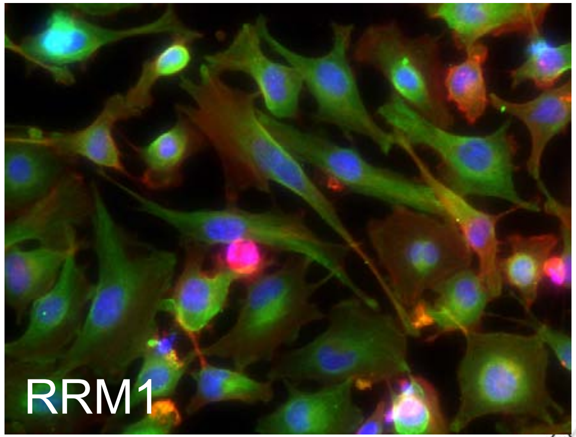


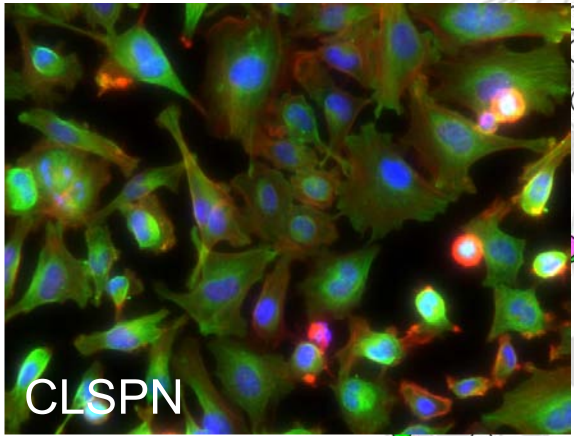
Mitotic phenotype



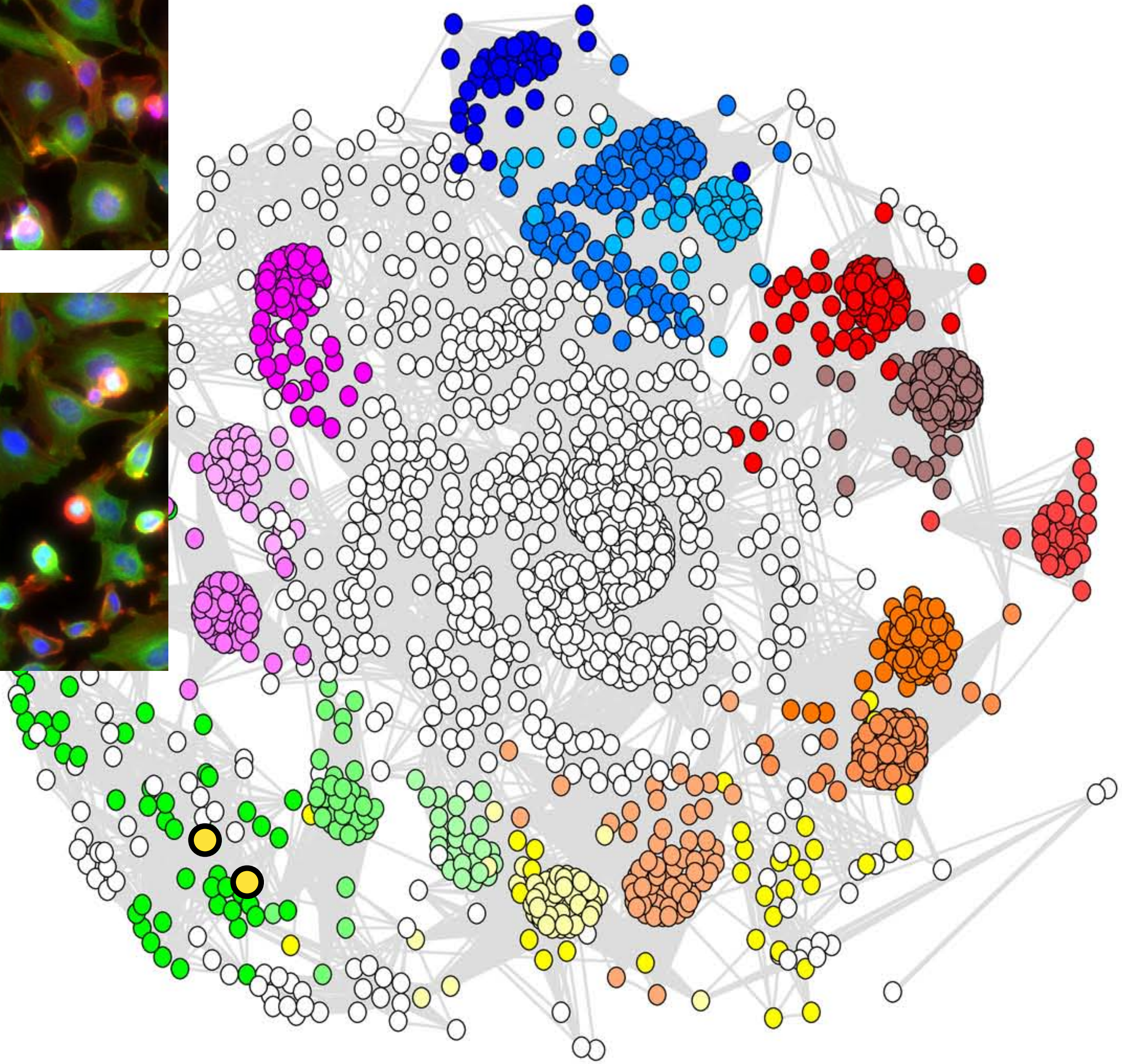






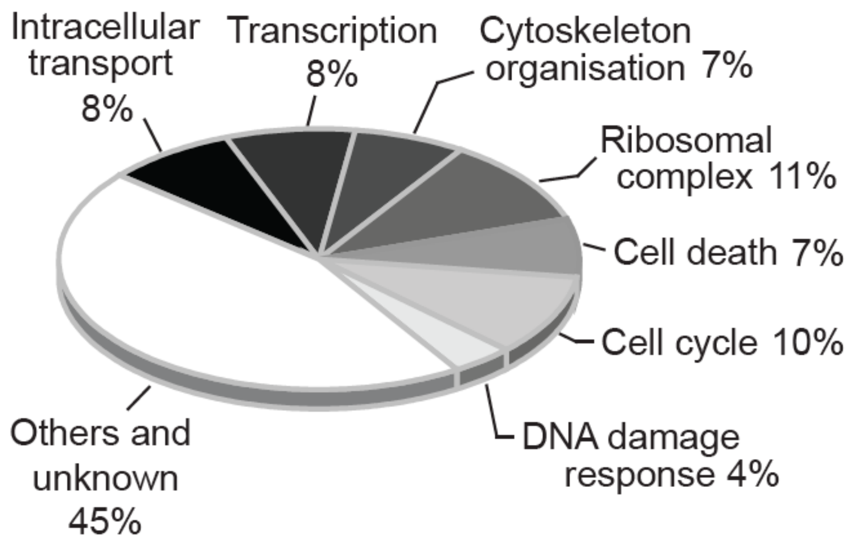


Giant cells with large nucleus



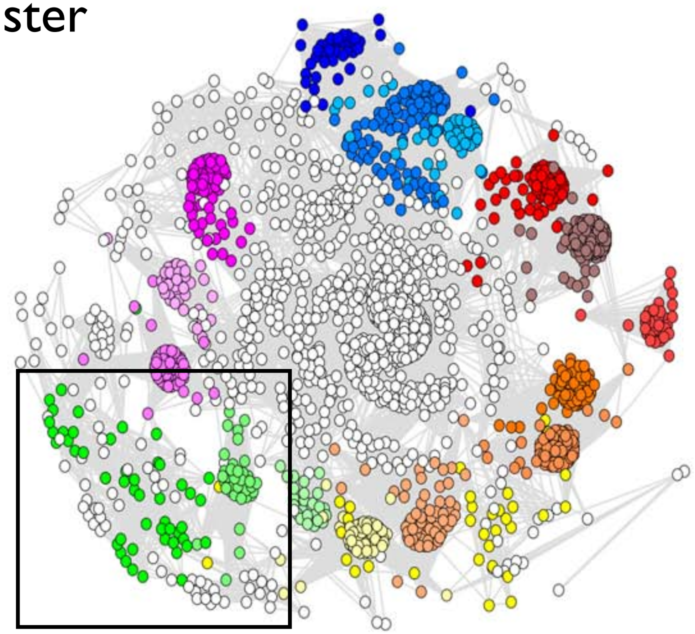
Validation

- 22839 siRNA perturbations
- 1820 non-null phenoprints
- 604 perturbations were retested
  - 310 reproduced the phenotypes with an independent siRNA library
  - Among them, 280 reproduced the phenotypes on U2OS cells



Functional inference

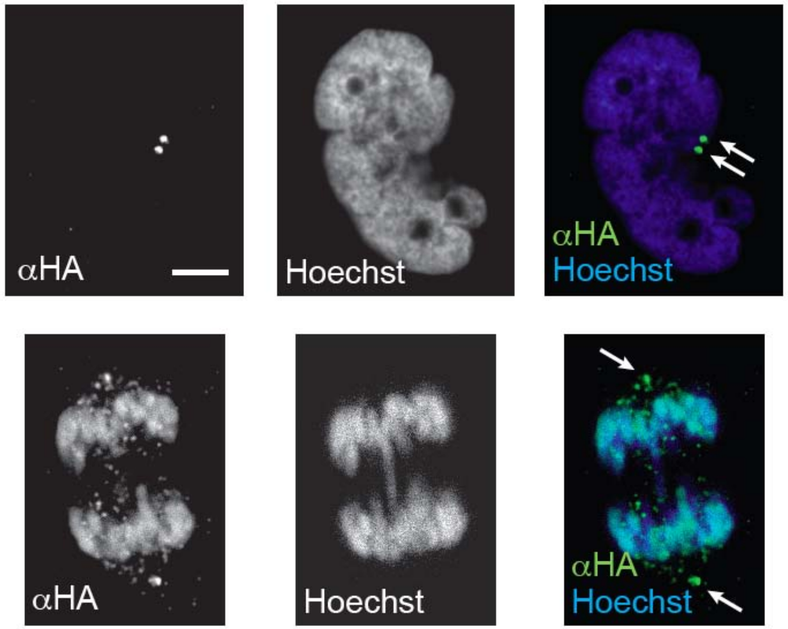
- "Giant cells with large nucleus" phenotypic cluster
  - 50 genes
  - RRM1, CLSPN, PRIM2 and SETD8
  - Mediators of the DNA damage response



- Secondary assays
  - Cell cycle progression upon depletion
  - Protein subcellular localization
  - Monitoring γH2AX foci formation upon depletion
  - Monitoring pChk1 response after gamma irradiation

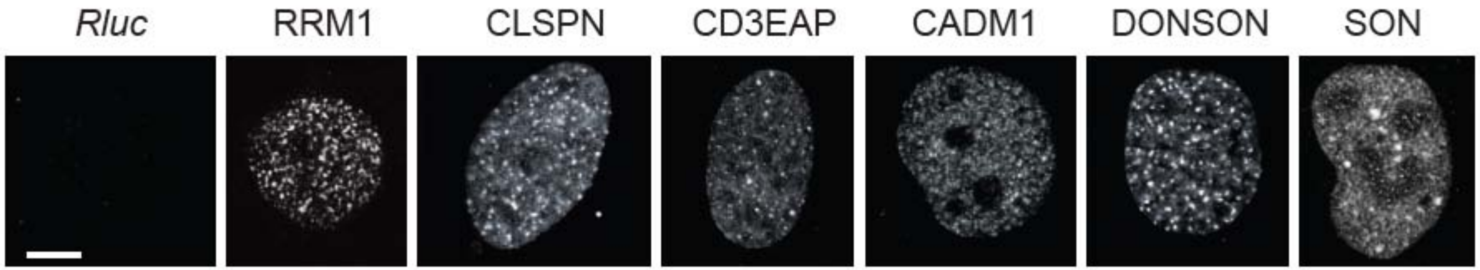
Subcellular localization

- DONSON localizes to the centrosomes
- Centrosomes are linked to DNA damage repair

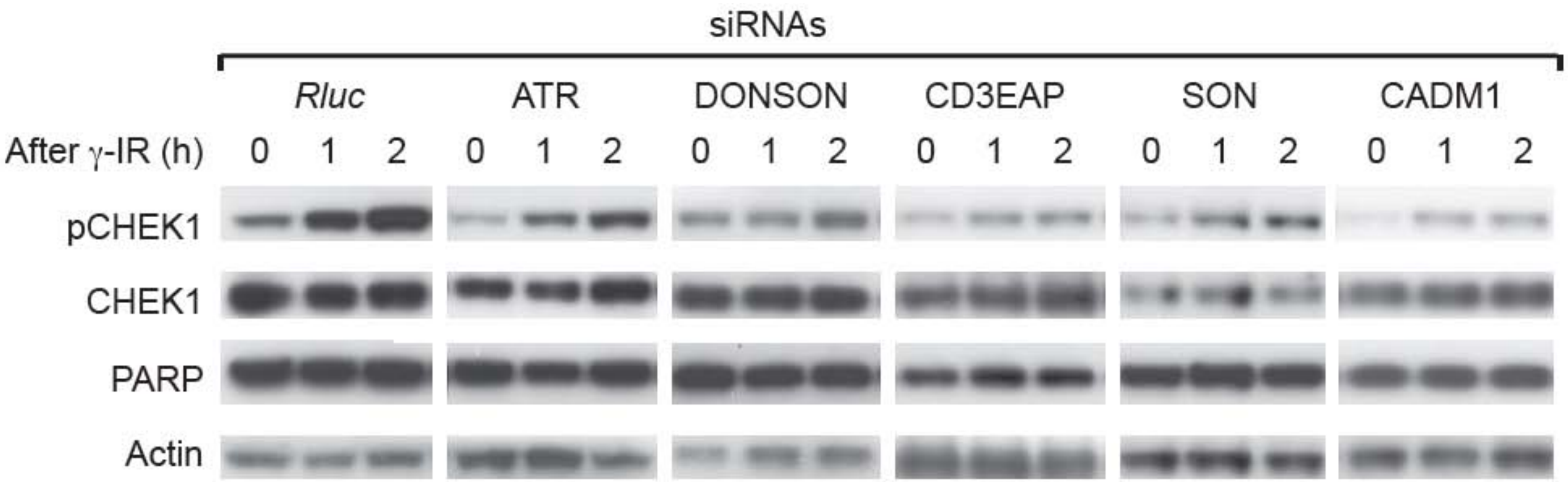


Monitoring DDR response

- Depletion of DONSON, SON, CD3EAP and CADM1
- Induction of γH2AX foci formation, an early DDR marker



- Inhibition of CHEK1 phosphorylation response upon gamma irradiation



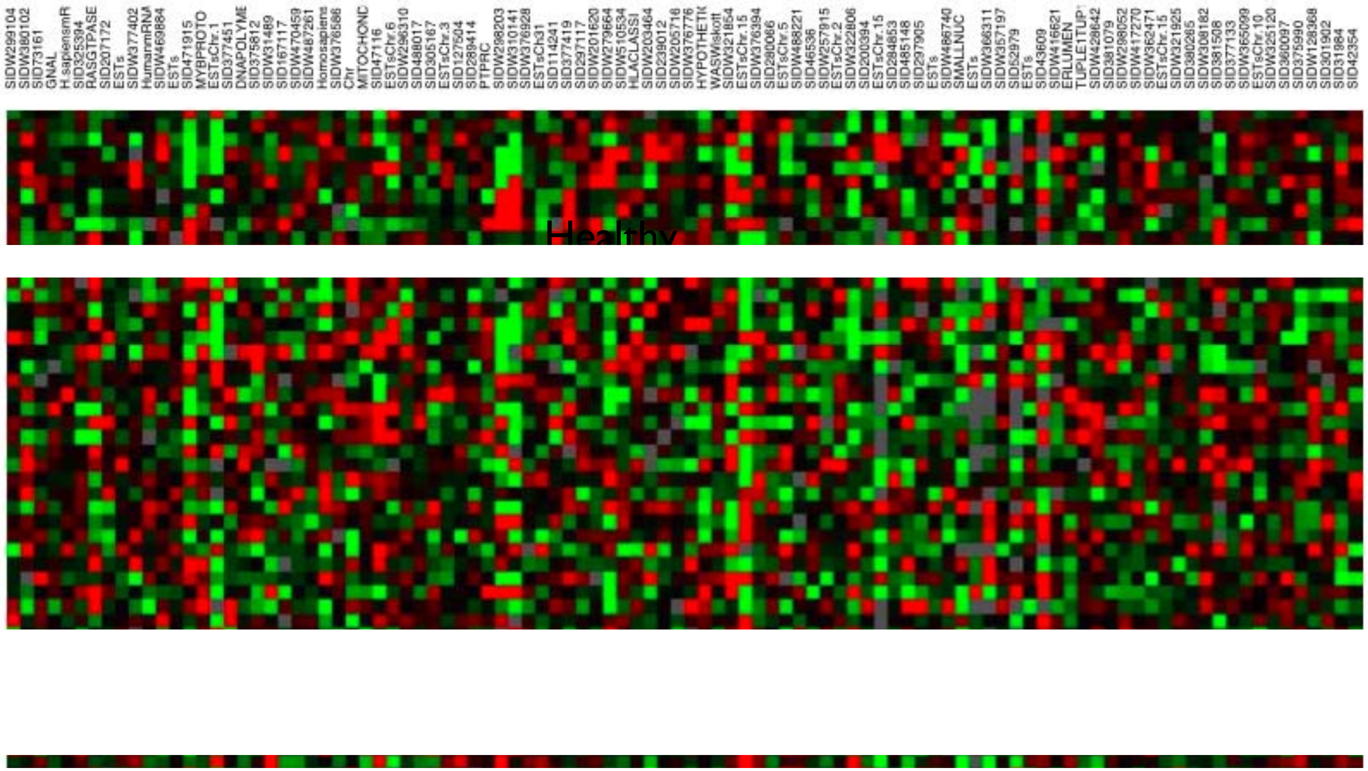
Conclusion

- Automated phenotyping method from microscopy images
- Prediction of gene function by loss-of-function phenotype similarity
- Association of DONSON, SON, CD3EAP and CADM1 to DDR
- Data available at http://www.cellmorph.org
- Bioconductor/R package: EBImage, imageHTS

Classification

Cancer prediction

- Known patient gene expression profiles



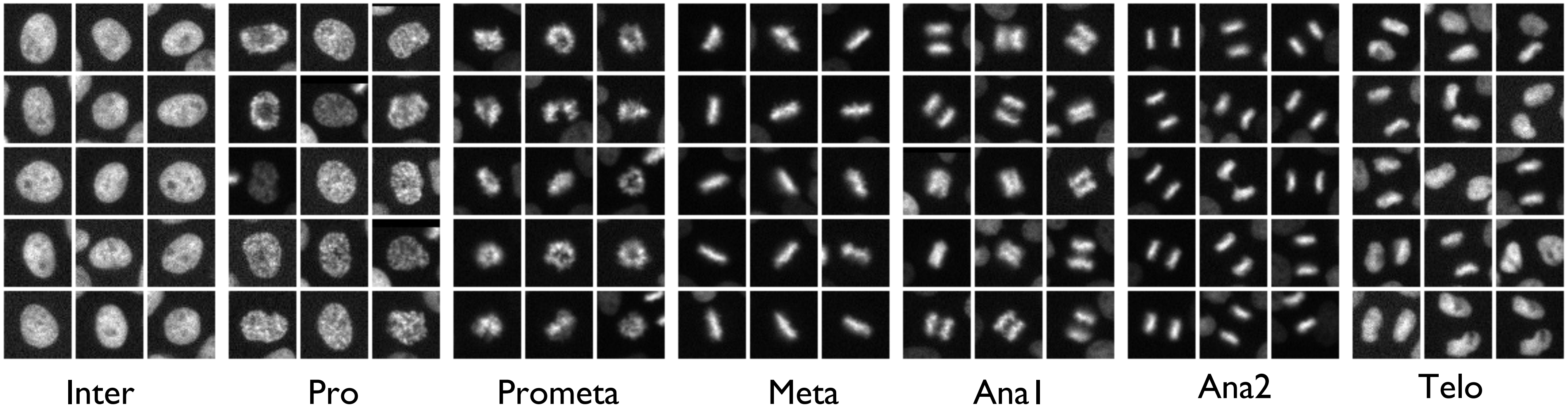
Healthy

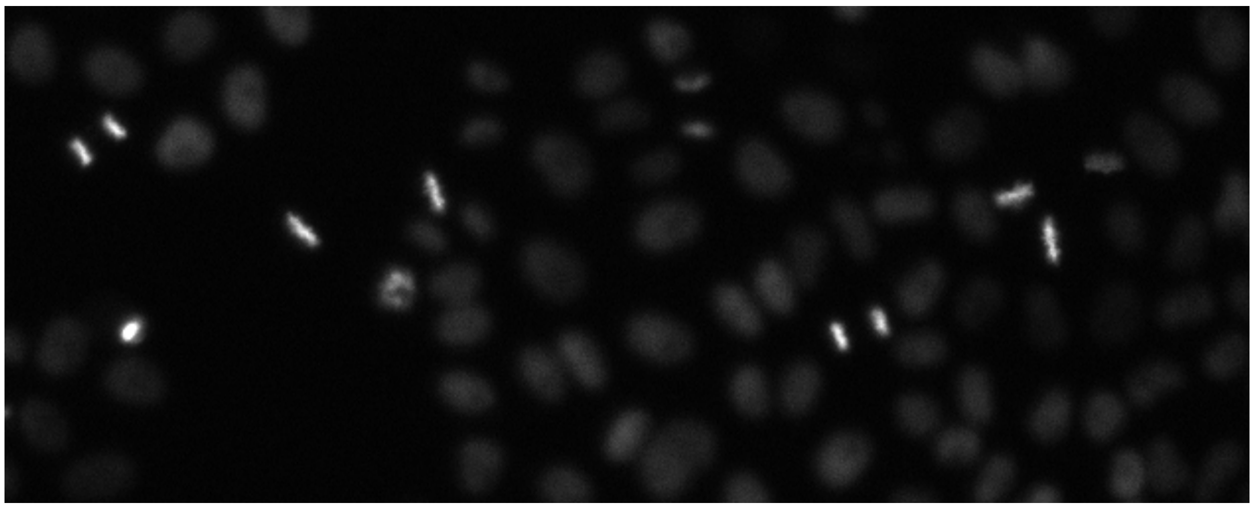
Acute myeloid leukemia

?

Automatic cell annotation

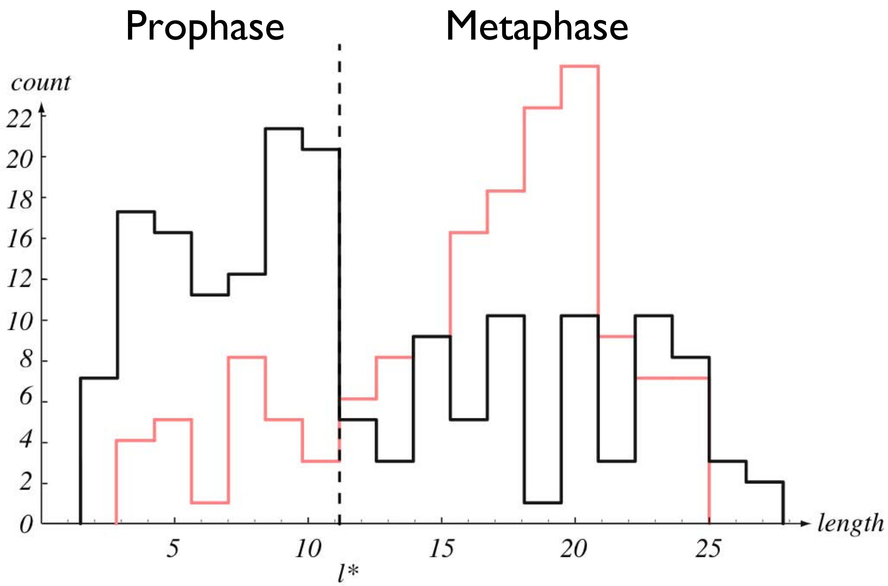
Training set

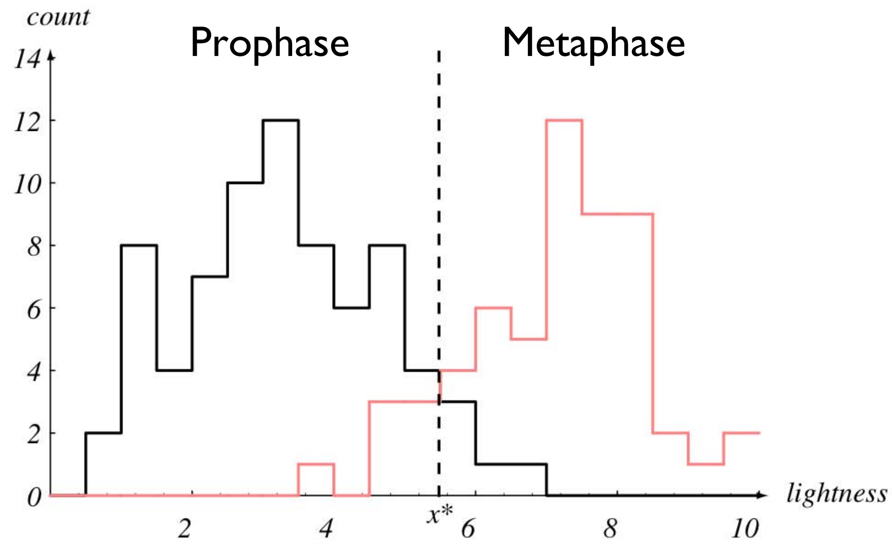




Prediction of mitotic state

- Based on nucleus size
- Based on nucleus intensity





- None of the two features is a good predictor of mitotic state
- Combining them ?

Classification

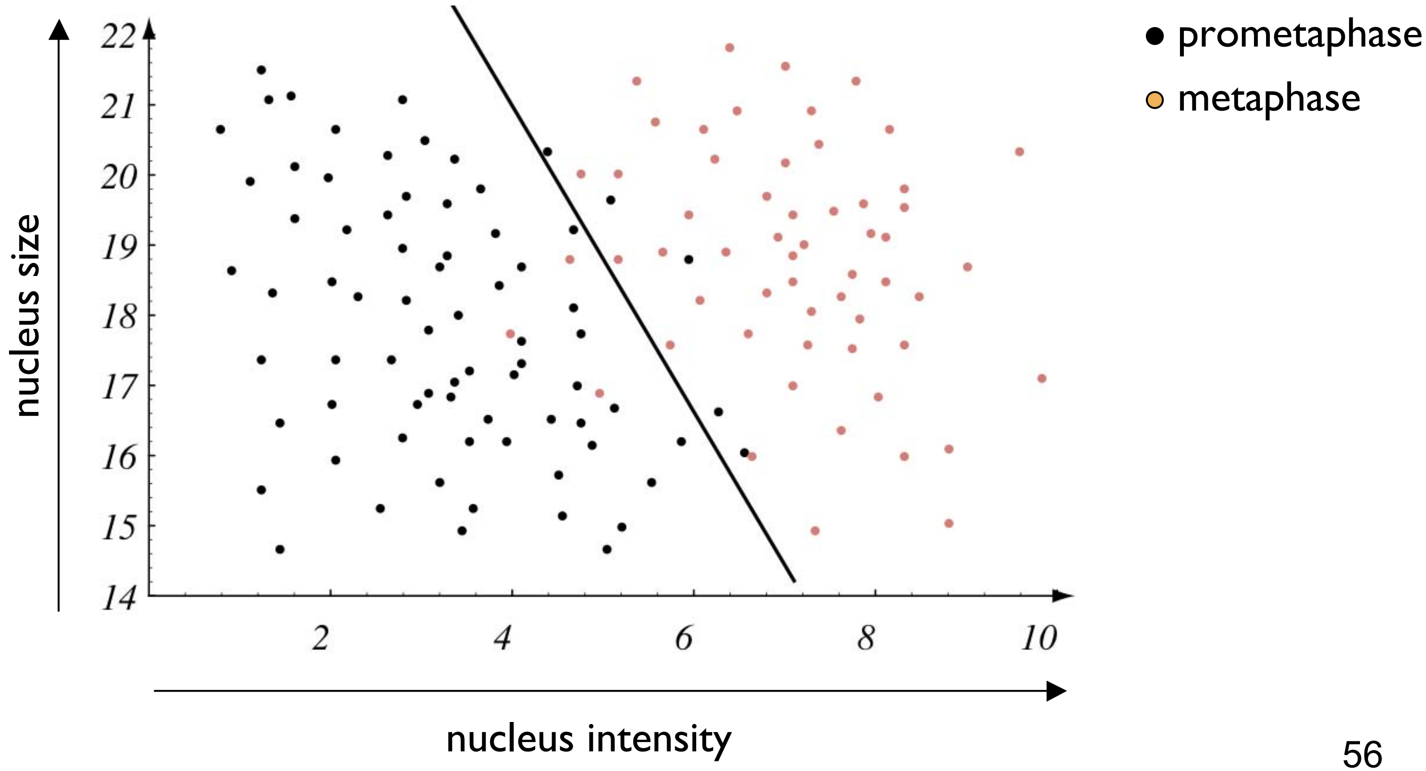
- Given objects with known labels, predict the label of an unknown object
- Well-defined but hard problem
- Optimal answers
  - Denote Y the outcome and X the data
  - Bayes formalism: $P(Y|X) = P(X|Y) * p(Y) / p(X)$
  - But $P(X|Y)$ is unknown and has to be estimated or modelled
  - Regression problem: $\mathrm{Argmin}_f \, ||Y - f(X)||^2 \Rightarrow f(x) = E(Y|X=x)$
  - But $E(Y|X=x)$ has to be estimated

- Algorithms
  - Linear regression
  - *k*-nearest neighbors
  - Support vector machines
  - Kernel methods

- Validation, cross-validation and overfitting

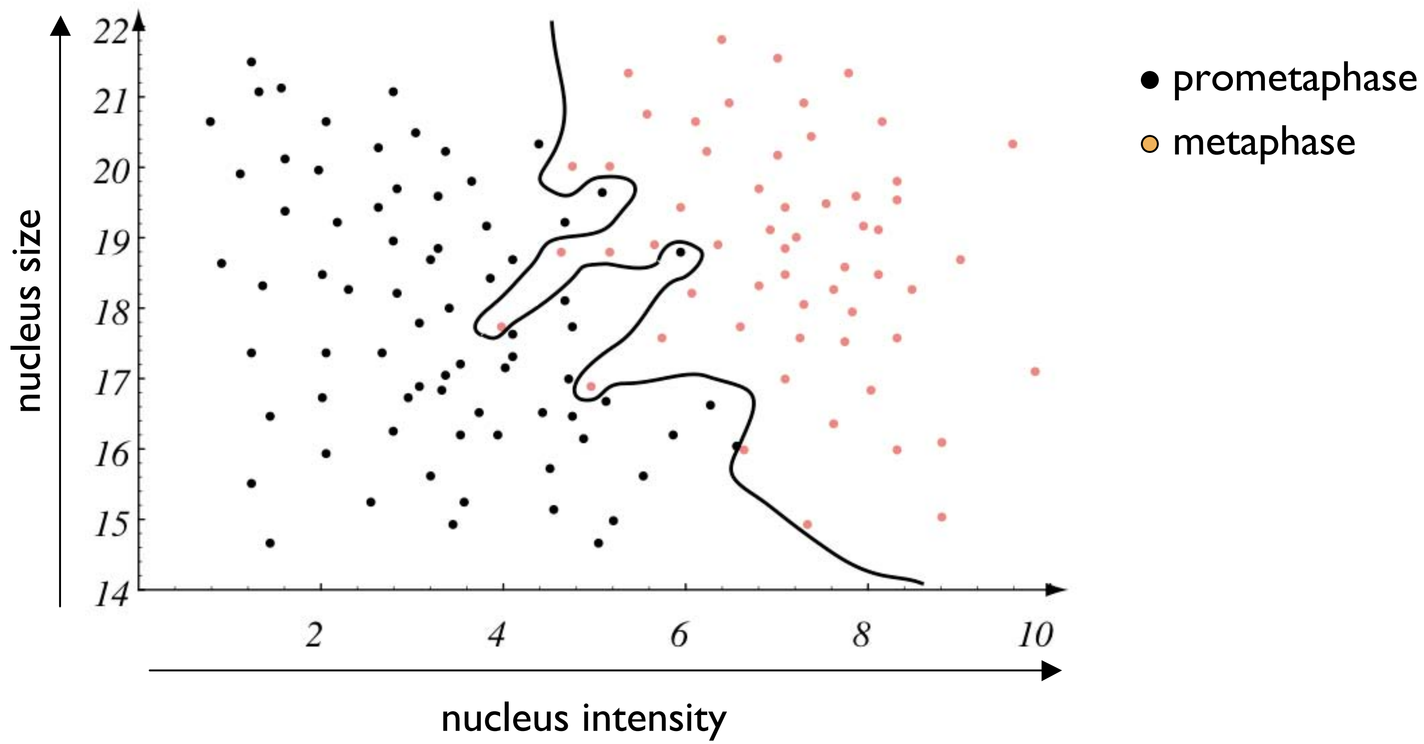
Linear classifier

- Denote by X the matrix of features: n samples, p features
- Denote by Y the vector of outcomes
- Find $\hat{\beta}$ that minimize $||Y - X\beta||^2$
- In R, using lm



k-nearest neighbors

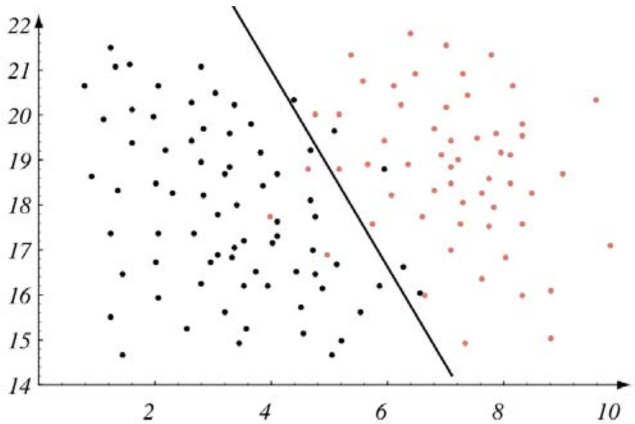
- Each point is assigned to the dominant label among its k-nearest neighbors
  - $f(x) = \mathrm{Avg}_{k \text{ in neighb}(x)}(y_k)$ approximates $E(Y|X=x)$
- In R, using knn

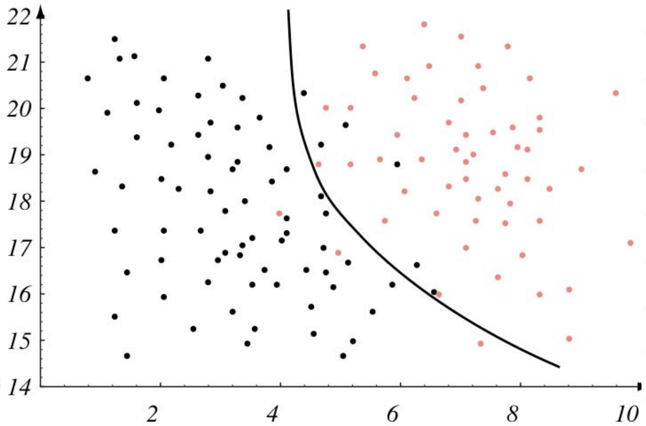


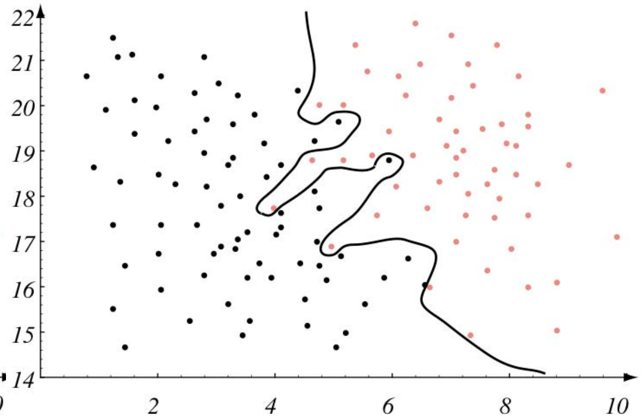
Which decision boundary ?

High bias
Low variance
<-----------------

Low bias
High variance
------------------->





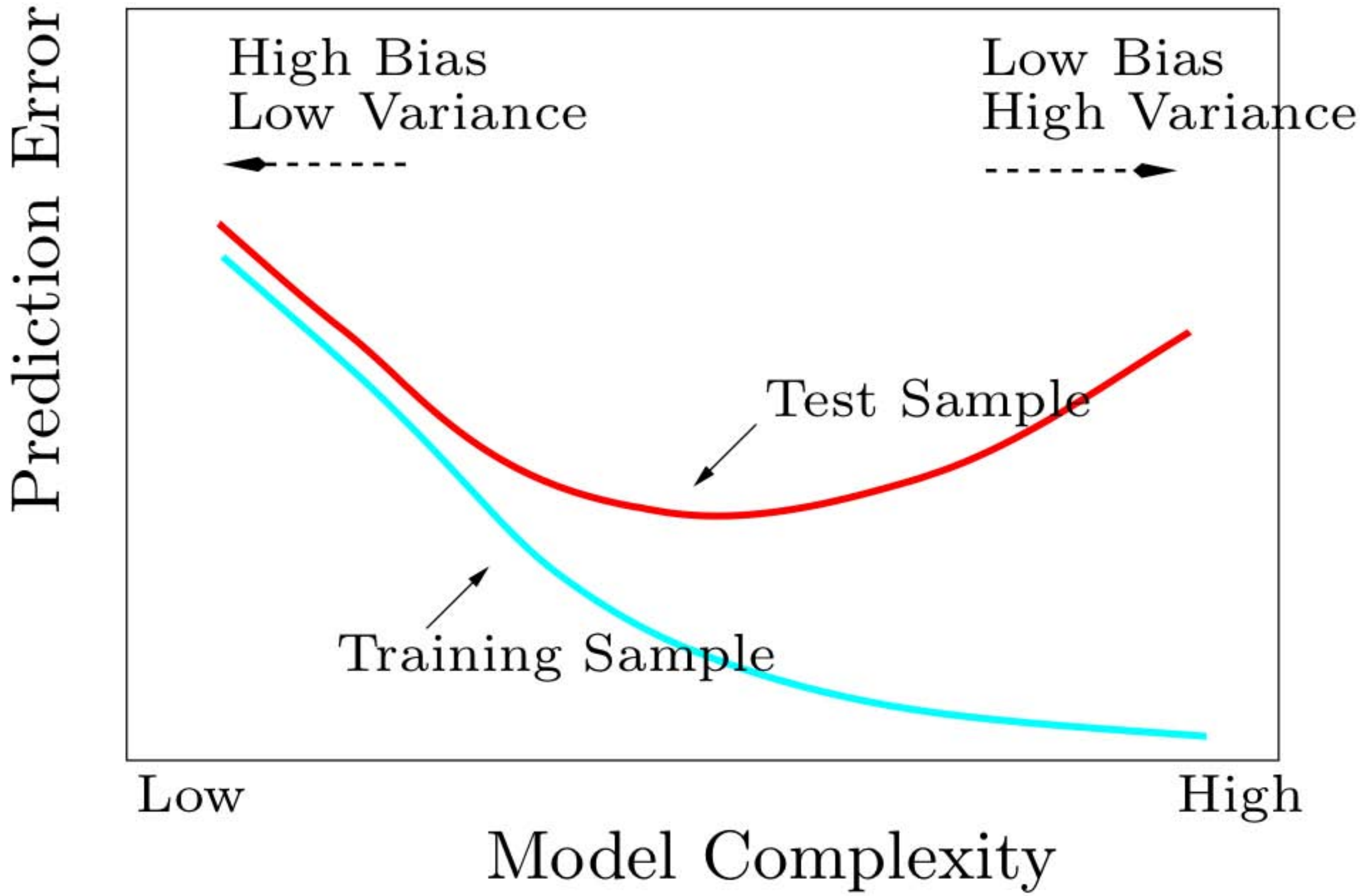


low model complexity
(needs 2 parameters to
describe the decision boundary)

high model complexity
(needs hundreds of parameter to
describe the decision boundary)

Which decision boundary has
the lowest
prediction error?

Bias-variance dilemma



Cross-validation

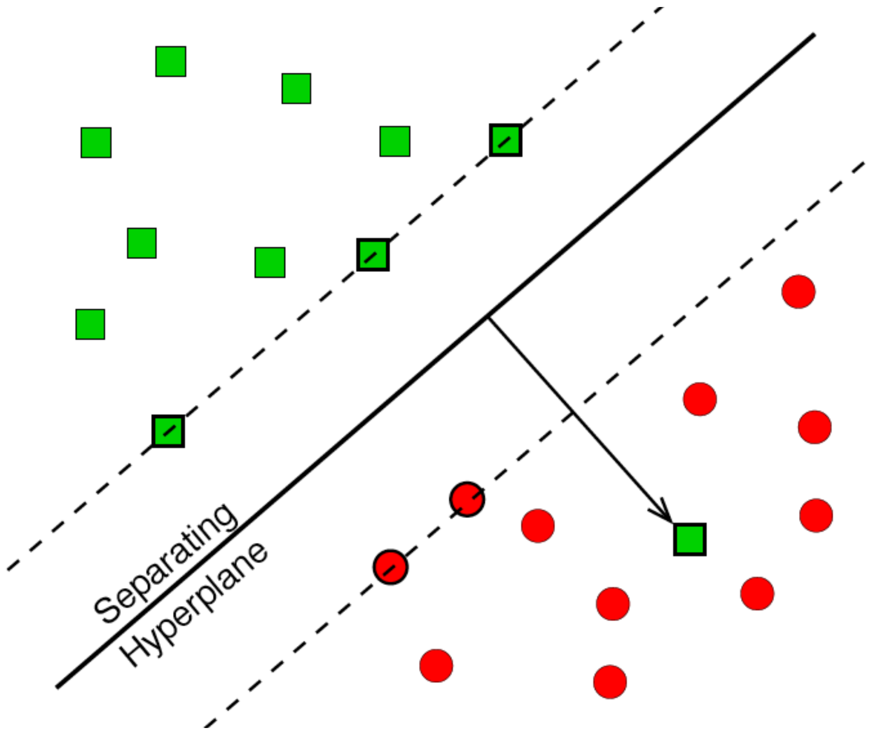
- Simple method to estimate the prediction error

- Method
  - Split the data in K approximately equally sized subsets
  - Train the classifier on (K-1) subsets
  - Test the classifier on the remaining subset. The prediction error is estimated by comparing the predicted class label with the true class labels.
  - Repeat the last two steps K times

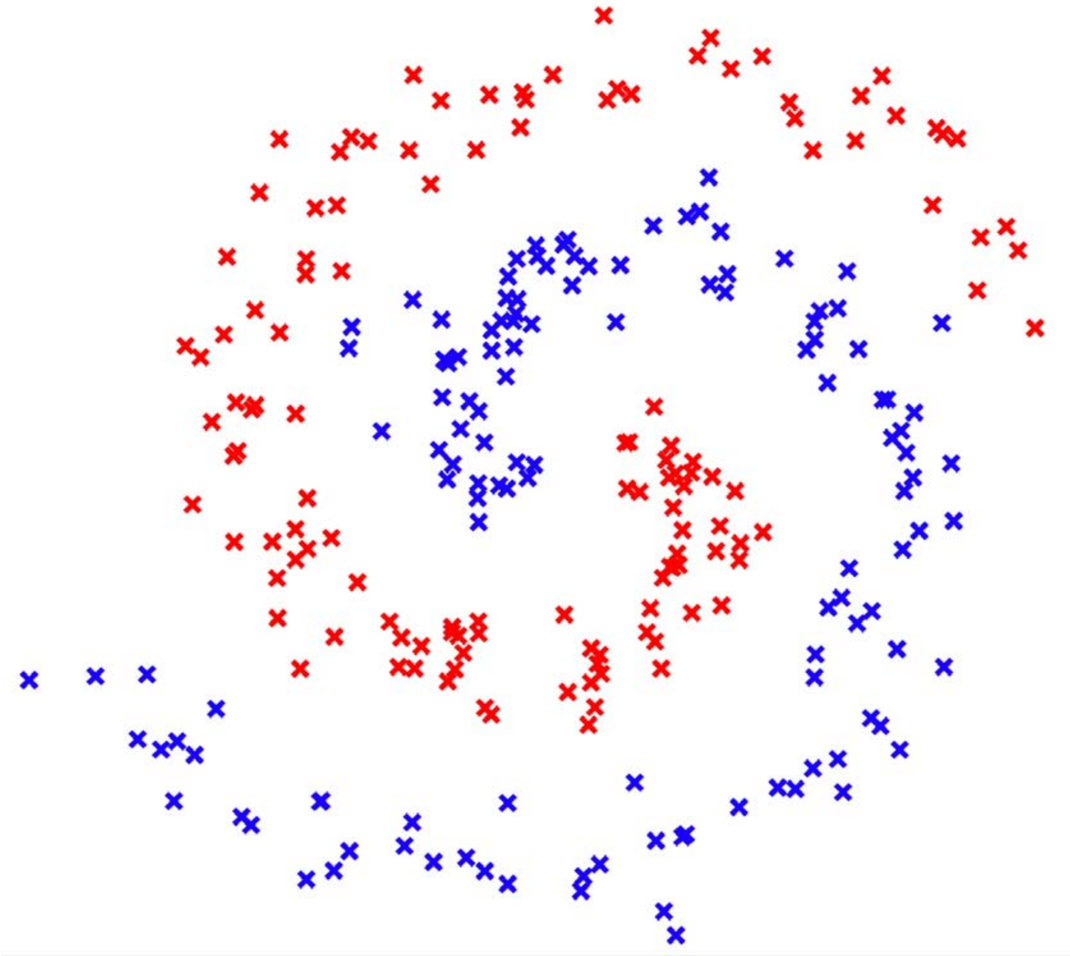
- Take the classifier that have the lowest prediction error

Support vector machine

- Find the hyperplane that best separates two sets of points
- Well-defined minimization problem, tolerant for misclassifications
  - Find $\hat{\beta}$ that minimize $||w||^2 + C|$
- In R, using svm from the package e1071

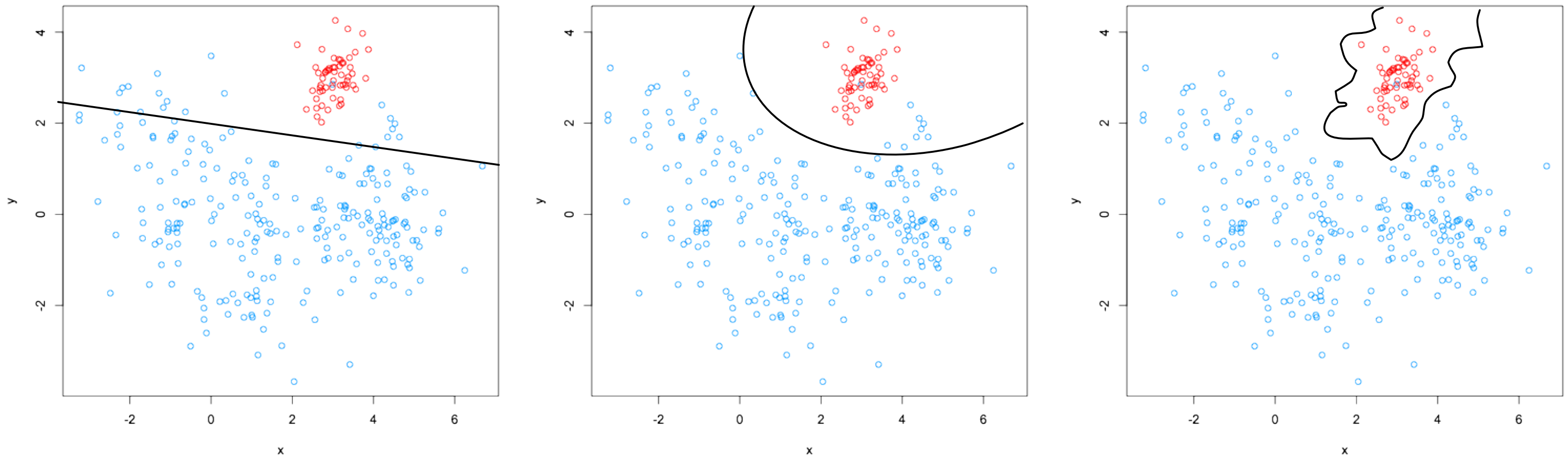


Non-linear case



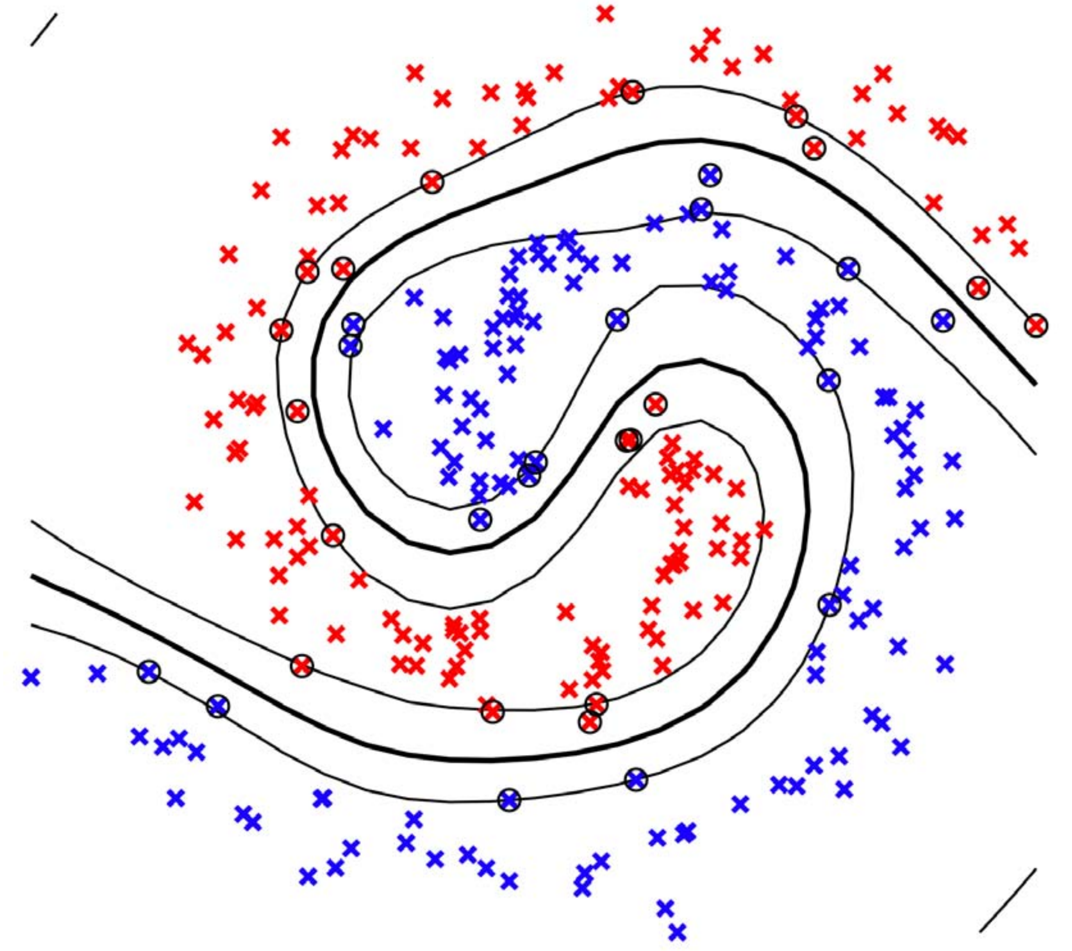
Basis expansion & the kernel trick

- Increase the dimension space if data cannot be linearly separated
  - Use X^2, X^3… e. g. $|| Y - [X;X^2;X^3]\beta ||^2$
  - Use splines or model-based separation curves

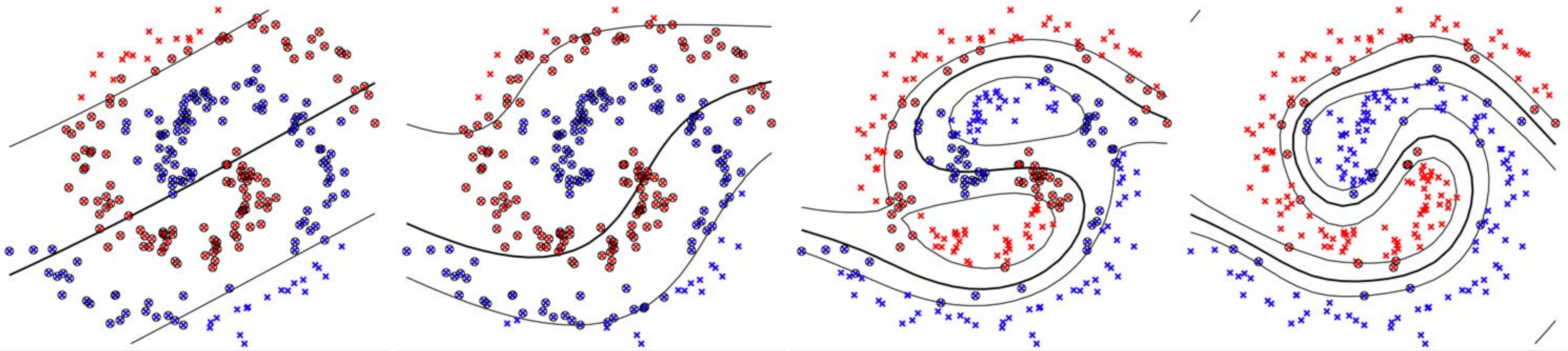


- Kernel trick
  - Scalar product $x^t y$ can be generalized by kernel functions $K(x, y)$
  - Kernel functions: $K(x, y) = x^t y$; $K(x, y) = \exp(-||x - y||/\gamma)$
  - SVM ⇒ kernel SVM ; LDA ⇒ kernel LDA ; PCA ⇒ kernel PCA
  - Complex separation in low-dimension ⇒ linear separation in high-dimension

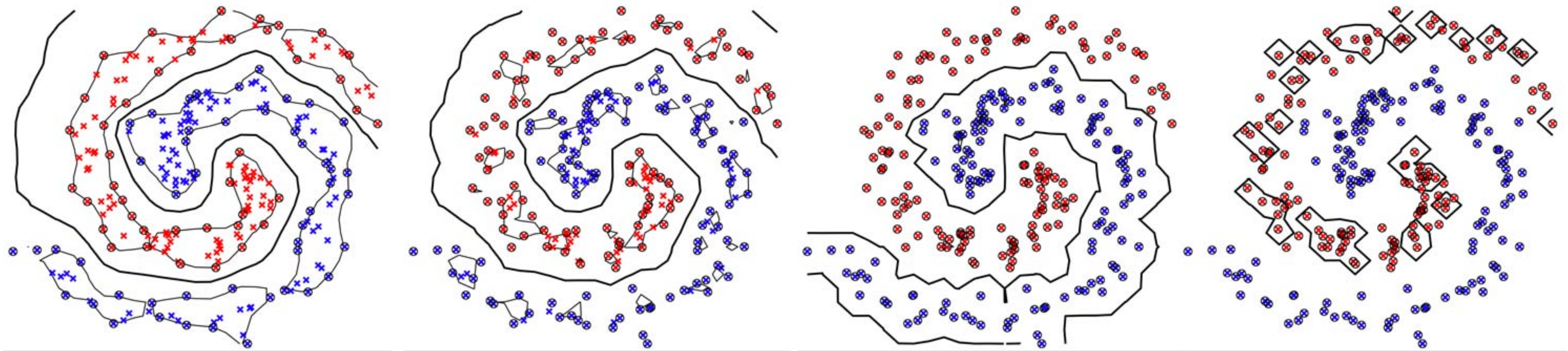
SVM + radial kernel



Influence of the kernel parameter



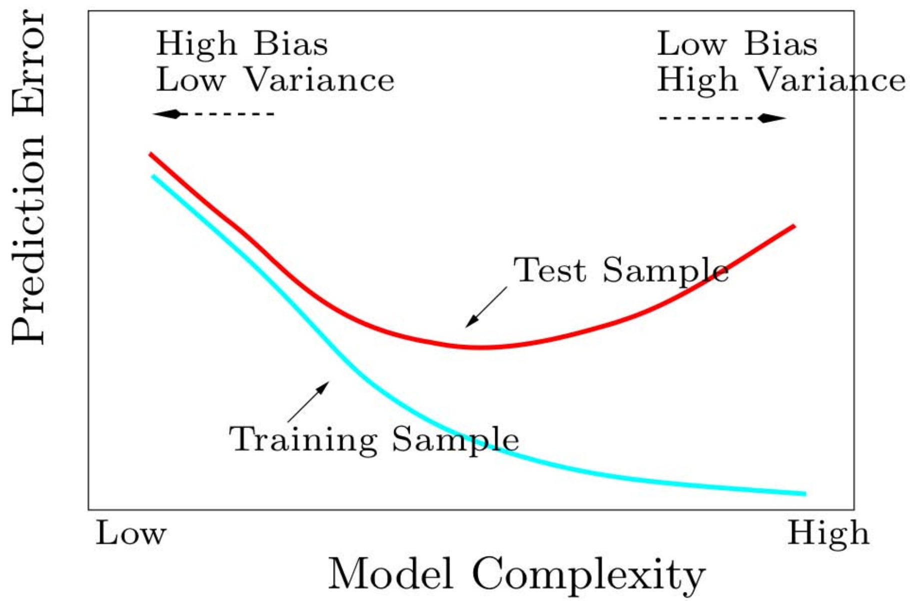
$\gamma = 0.001$ $\gamma = 0.005$ $\gamma = 0.03$ $\gamma = 0.1$



$\gamma = 1$ $\gamma = 2$ $\gamma = 20$ $\gamma = 200$

Curse of dimensionality

- Low number of parameters
  - Low complexity
  - Low variance
  - High bias



- High number of parameters
  - High complexity
  - High variance
  - Low bias
  - Space is too sparse; estimation is not reliable

- Trade-off must be found by prediction error estimation

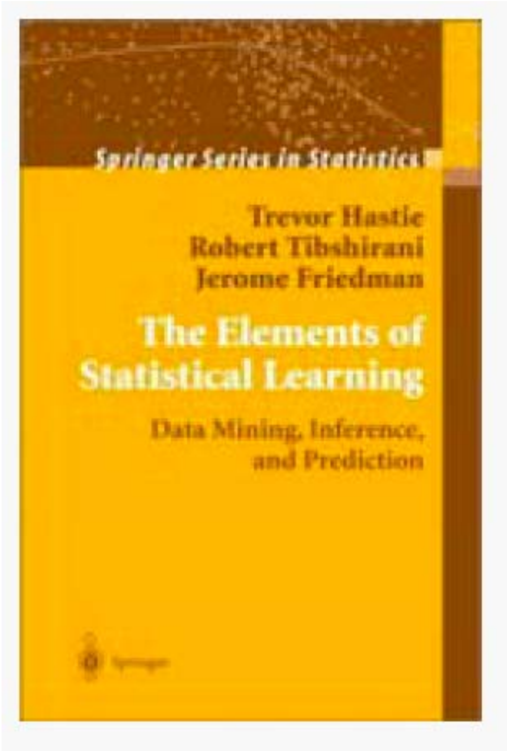
Conclusion

- Clustering
  - Ill-defined problem $\Rightarrow$ many algorithms around
  - Most important: a relevant dissimilarity measure
  - Requires cautious interpretation
  - Still useful tool for data exploration

- Classification
  - Well-defined problem
  - Kernel SVM is a fast and versatile algorithm suitable to many problems
  - Most important: prediction error estimation using cross-validation

- Feature selection
  - Supervised feature selection
  - Regular penalized methods (e.g. Lasso) are key techniques

Going further



The Elements of Statistical Learning

Hastie, Tibshirani and Friedman

- Statistical learning
- Machine learning
- Features selection
- Classification
- Unsupervised clustering
- Kernel methods
- Neural networks
- Boosting

Acknowledgments

- Thanks to Bernd Fisher, Richard Bourgon, Joerg Rahnenfuehrer !