LUXEMBOURG INSTITUTE OF HEALTH
RESEARCH DEDICATED TO LIFE
Proteome and Genome Research Unit

An Interdisciplinary Summer School on Mining of Biological Data for MSc and PhD students

Module 8:

Clustering

Petr Nazarov

petr.nazarov@lih.lu

2018-08-13, room 227

http://edu.sablab.net/nmbu2018/

Outline

2018-08-13

- Clustering task
- Iris dataset
- Hierarchical clustering
  - Consensus clustering
- Non-hierarchical methods
  - k-means
  - PAM
  - Choose number of clusters
  - Non-hierarchical methods
- Density-based clustering
- Dataset for practical work

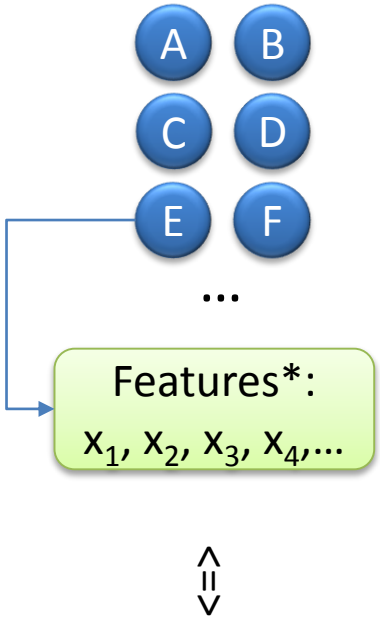
Followed by practical work

2018-08-14

- Single-cell (SC) transcriptomics
- SC data properties
- M1: Independent component analysis (ICA)
- M2: t-distributed stochastic neighbor embedding (tSNE)
- Examples

http://edu.sablab.net/nmbu2018/

General Overview



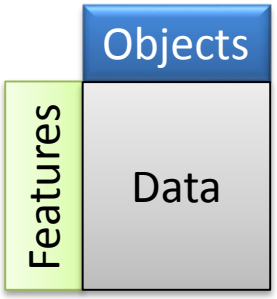
1. Feature selection/ construction
- all features*
- PCA, MDS, ICA, tSNE
- a priory knowledge

(*) Additionally, features could be transformed to ensure reasonable importance of each feature, e.g:
- log-transformation,
- standardization,
- ranking, etc.

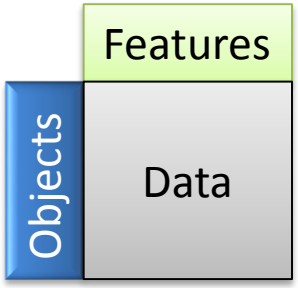
2. Distance
- Euclidean
- correlation: 1 - *r*
- Manhattan
- cosine (inner product)

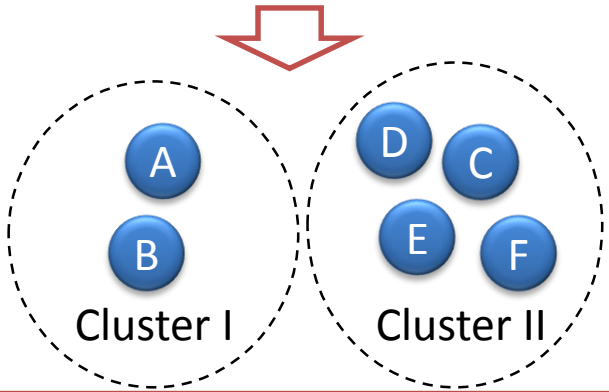
3. Clustering
- hierarchical
- k-means
- density-based (DBSCAN)

In genomics:



In machine learning:





Dataset 1

Iris Data (R.A.Fisher)

The Iris flower data set or Fisher's Iris data set is a multivariate data set introduced by Sir Ronald Aylmer Fisher (1936) as an example of discriminant analysis. It is sometimes called Anderson's Iris data set because Edgar Anderson collected the data to quantify the geographic variation of Iris flowers in the Gaspé Peninsula.
The dataset consists of 50 samples from each of three species of Iris flowers (*Iris setosa*, *Iris virginica* and *Iris versicolor*). Four features were measured from each sample, they are the length and the width of sepal and petal, in centimeters. Based on the combination of the four features, Fisher developed a linear discriminant model to distinguish the species from each other.

Iris setosa

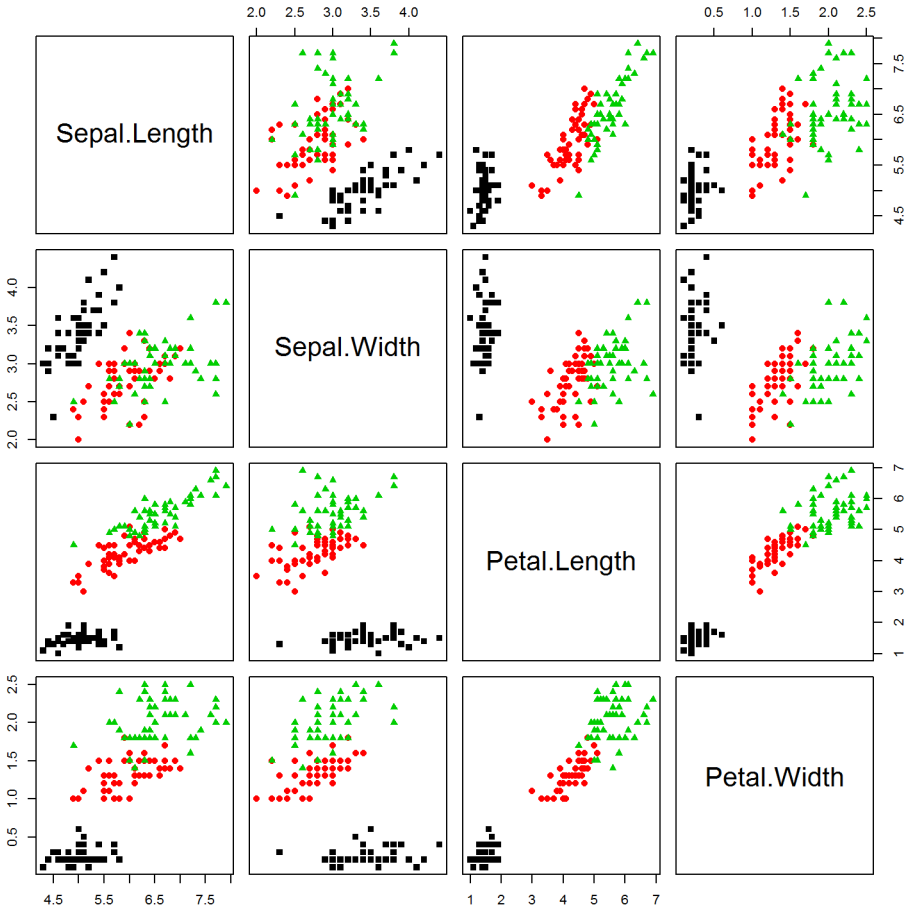
Iris versicolor

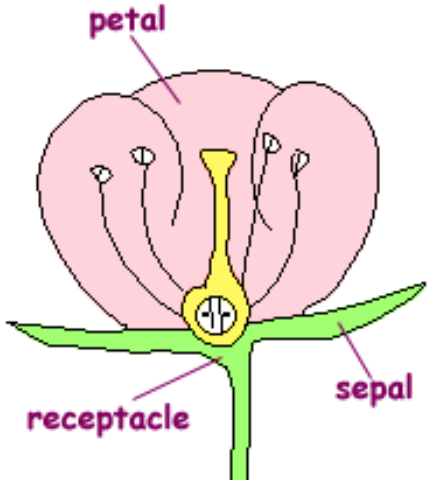
Iris virginica

Dataset 1

Iris Dataset Presentation

```
print(iris)
str(iris)
plot(iris[,-5],
     col=as.integer(iris[,5]),
     pch=19)
```





http://urbanext.illinois.edu/gpe/case4/c4facts1a.html

How could we possibly represent these data on a single plot?

Dataset 1

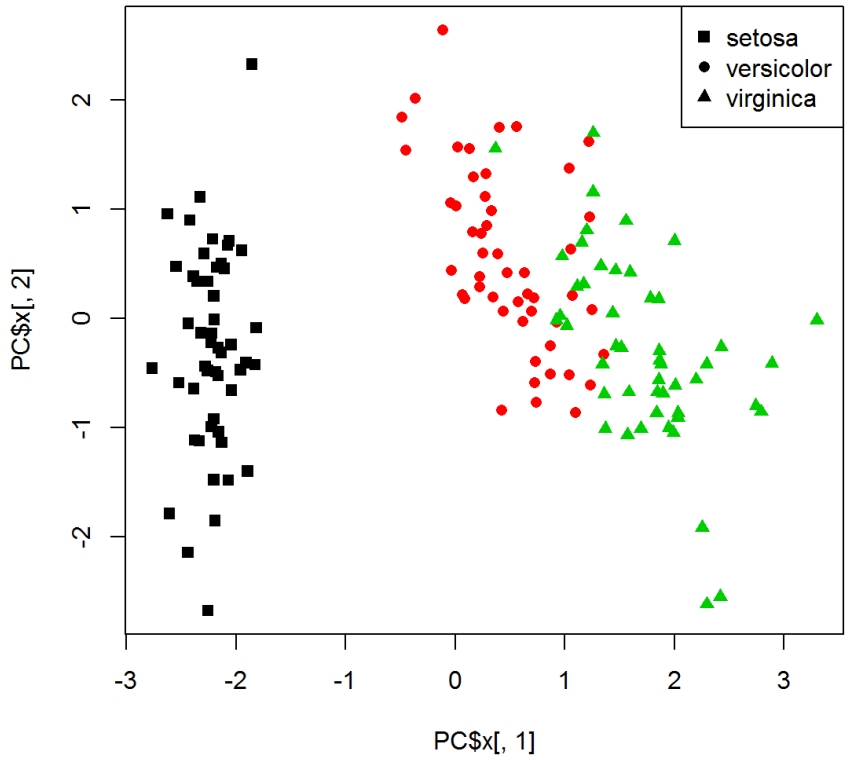
PCA of iris dataset

Iris setosa (1)

Iris versicolor (2)

Iris virginica (3)

```
## prepare features
X = as.matrix(iris[,1:4])
X[,] = scale(X)
color = as.integer(iris$Species)
## perform PCA
PC = prcomp(X)
str(PC)
## plot PC1 and PC2 only
plot(PC$x[,1],
     PC$x[,2],
     col=color,
     pch=point)

## alternative
source("http://sablab.net/scripts/
plotPCA.r")
plotPCA(t(X),col=color,pch=point)
```

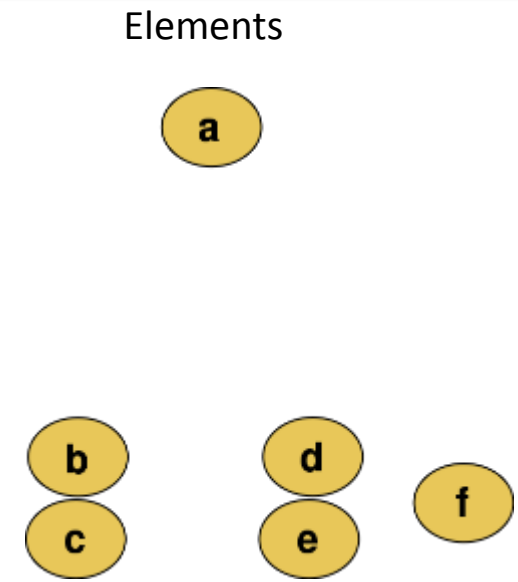


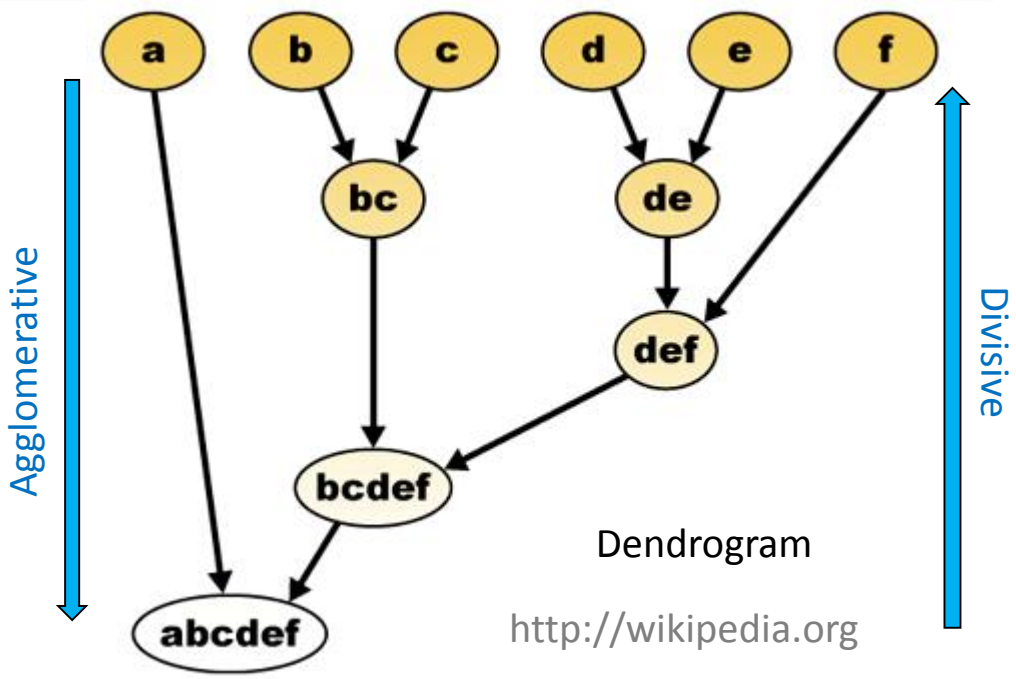
Hierarchical Clustering

Basic Hierarchical Clustering

Hierarchical Clustering

Hierarchical clustering creates a hierarchy of clusters which may be represented in a tree structure called a dendrogram. The root of the tree consists of a single cluster containing all observations, and the leaves correspond to individual observations.
Algorithms for hierarchical clustering are generally either agglomerative, in which one starts at the leaves and successively merges clusters together; or divisive, in which one starts at the root and recursively splits the clusters.



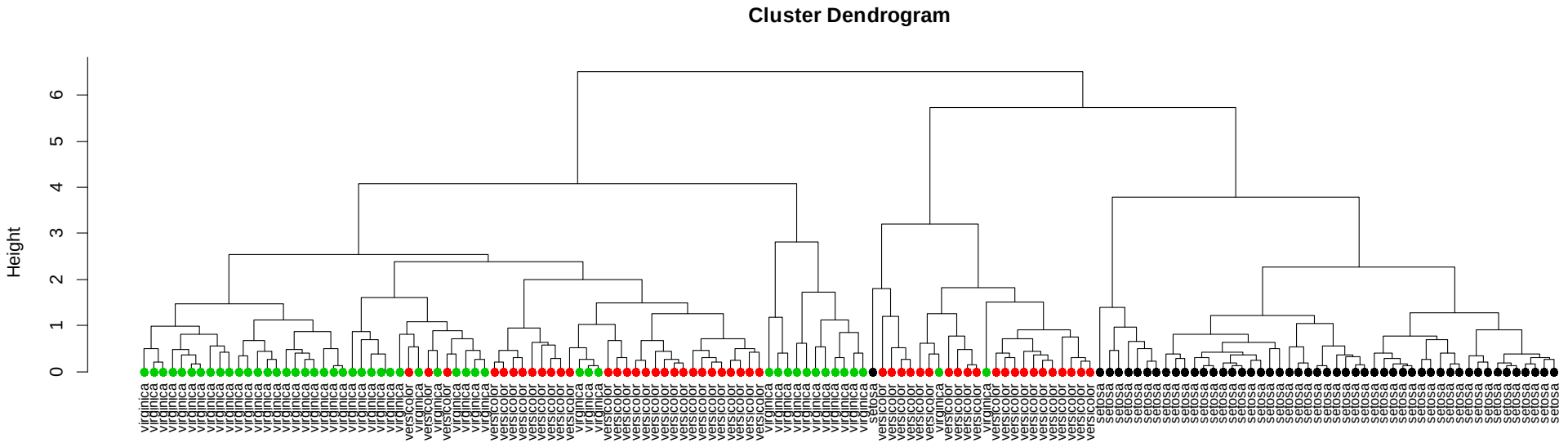


Hierarchical Clustering

Iris dataset

```
H = hclust(dist(X))
plot(H,
     labels=iris$Species,
     hang=-1,
     cex=0.75)
```

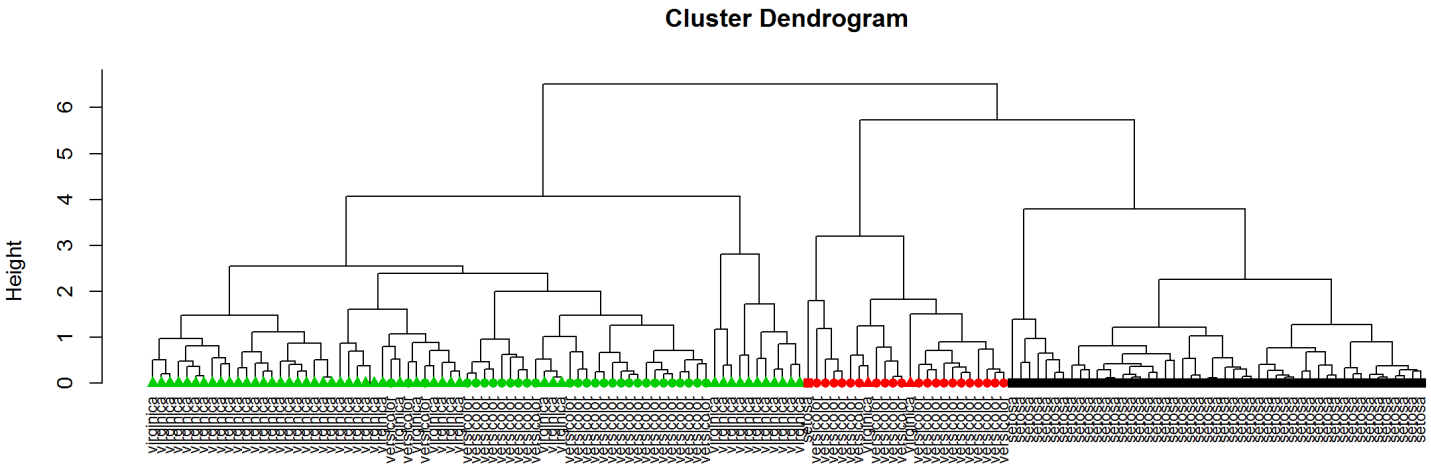
```
## optional - add points
points(x=1:nrow(X),
       y=rep(0,nrow(X)),
       pch=19,
       col=color[H$order])
```



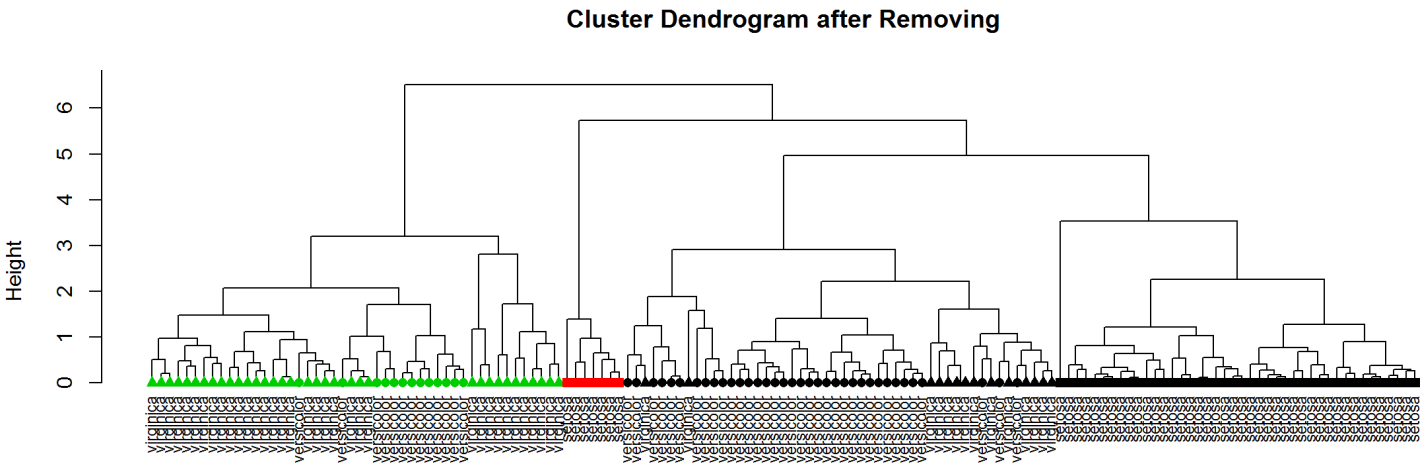
NOTE: In the online materials you will see the different coloring (by cluster, not by species)

Hierarchical Clustering

Problem: low stability



Let's remove 1 flower from each group (-41,-98,-144) and repeat clustering:



Clustering 2

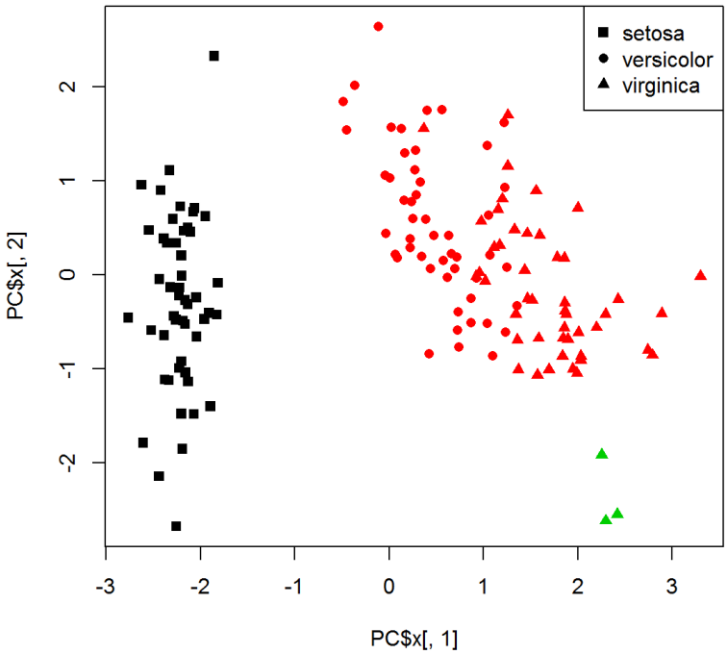
Clustering 1	1	2	3
1	41	7	0
2	24	0	0
3	27	0	48

Hierarchical Clustering

Consensus Clustering

1. Resampling of the original set
2. Clustering
3. Summarizing the results

However, no guarantee that you will get what you expect… 3 group consensus HC:



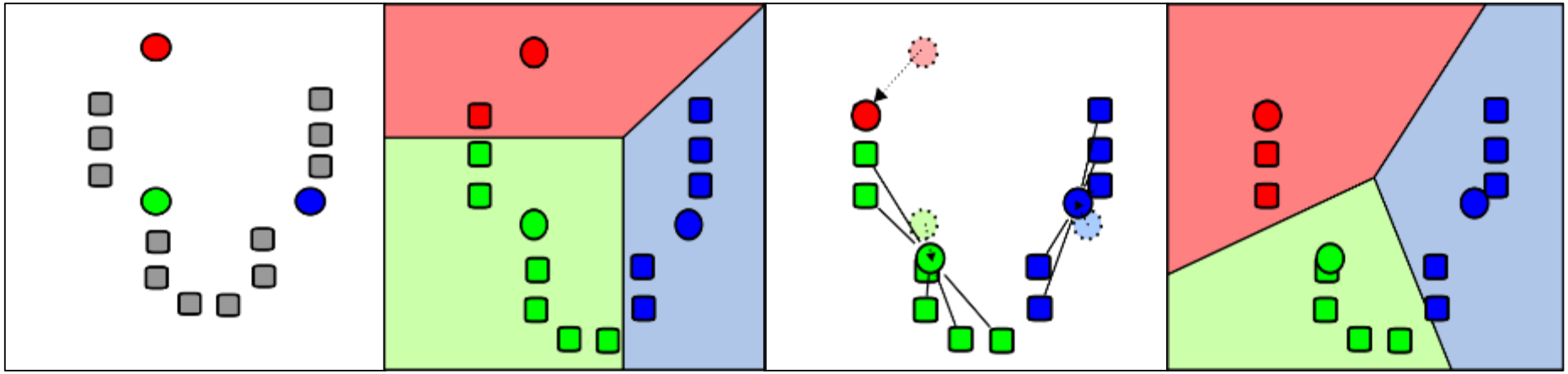

```
library(ConsensusClusterPlus)

results = ConsensusClusterPlus(
    t(X),maxK=6,reps=50,pItem=0.8,pFeature=1,
    title="IRIS ConClust",clusterAlg="hc",
    distance="euclidean",seed=12345,plot="png")

plot(PC$x[,1],PC$x[,2],
     col=results[[3]]$consensusClass,
     pch=point)
```

Non-hierarchical Clustering

k-Means Clustering

k-Means Clustering

k-means clustering is a method of cluster analysis which aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean.



1) *k* initial "means" (in this case k=3) are randomly selected from the data set (shown in color).

2) *k* clusters are created by associating every observation with the nearest mean.

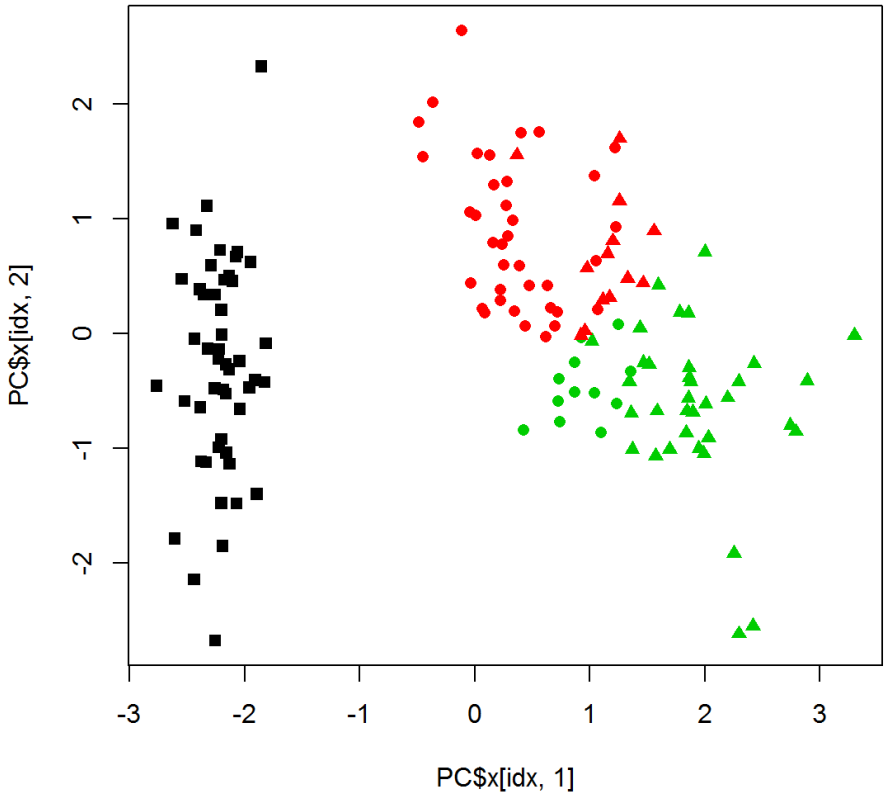
3) *The centroid of each of the k clusters becomes the new means.*

4) Steps 2 and 3 are repeated until convergence has been reached.

http://wikipedia.org

Non-hierarchical Clustering

k-Means Clustering




```
cl = kmeans(X,
            centers=3,
            nstart=10)$cluster

plot(PC$x[,1],PC$x[,2],
     col = cl, pch = point)
```

Let's remove 1 flower from each group (-41,-98,-144) and repeat clustering:

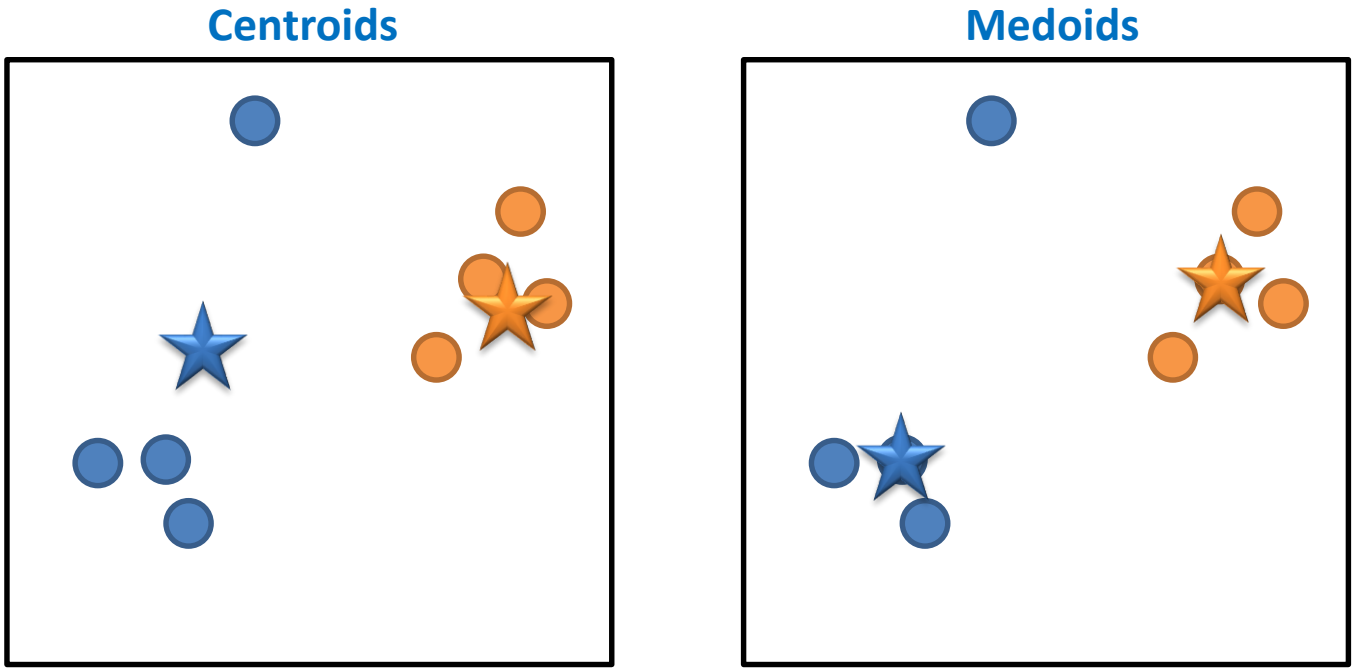
	1	2	3
1	46	0	0
2	1	51	0
3	0	0	49

Much more stable than HC! But still sensitive to outliers

Non-hierarchical Clustering

PAM: Partitioning Around Medoids (k-medoids)

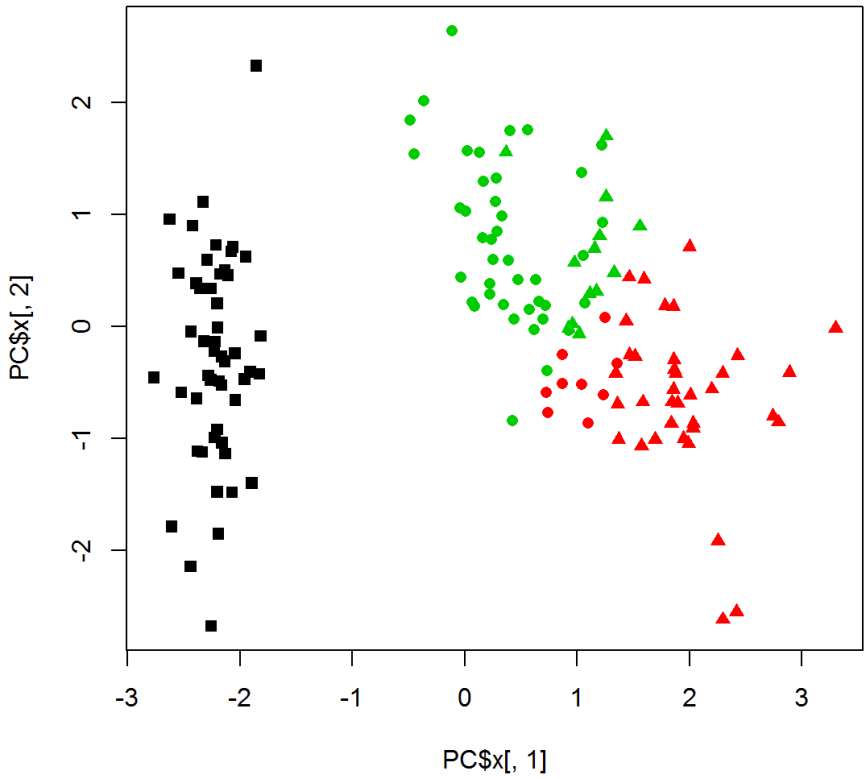
PAM is a version of “k-means” that is more robust to outliers. Instead of calculated centroids, it uses medoids – representative objects of each class.



Non-hierarchical Clustering

PAM: Partitioning Around Medoids (k-medoids)

```
library(cluster)
cl = pam(X,k=3,nstart=10)$cluster
plot(PC$x[,1],PC$x[,2],col = cl, pch=point)
```



Remove 1 flower from each group (-41,-98,-144) and repeat clustering:

	1	2	3
1	49	0	0
2	0	44	0
3	0	11	43

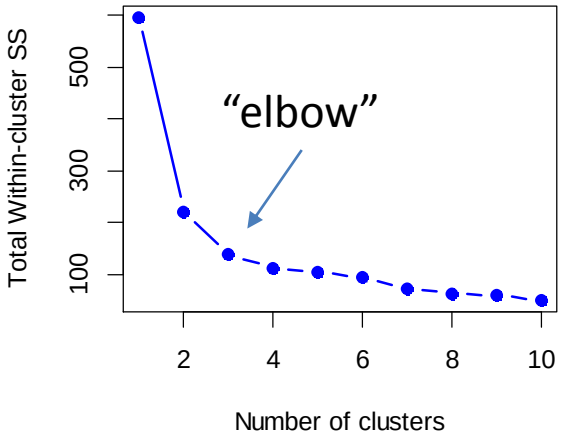
Number of Clusters

There is no universal ("magical") solution, so aim at:

- method that gives most logical clustering (e.g. on "training" set)
- method that you could defend in your paper ☺

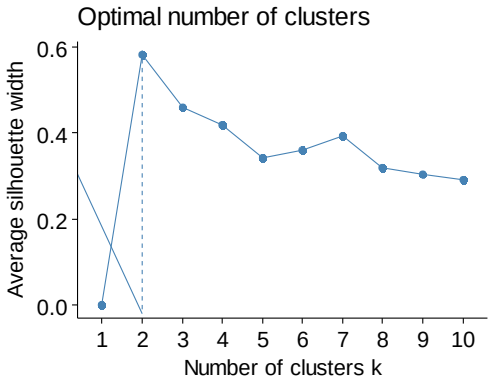
Elbow

Minimizes within-cluster sum of square (WSS)



Silhouette

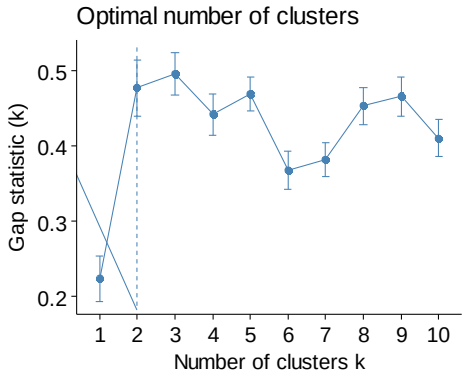
Silhouette – similarity to own cluster members compared to members of other clusters



Gap statistics

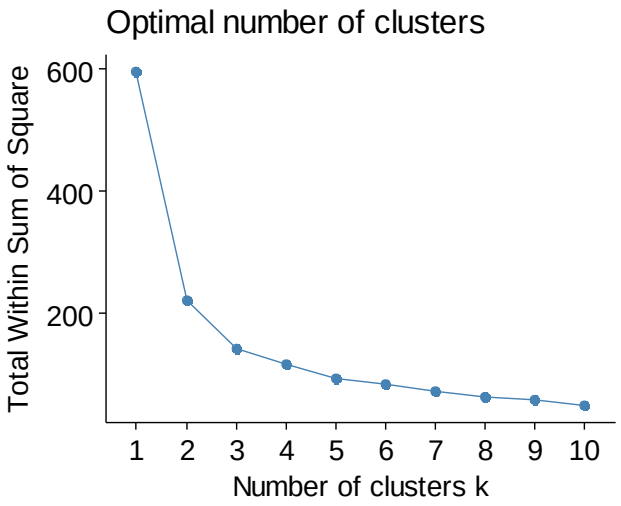
http://web.stanford.edu/~hastie/Papers/gap.pdf

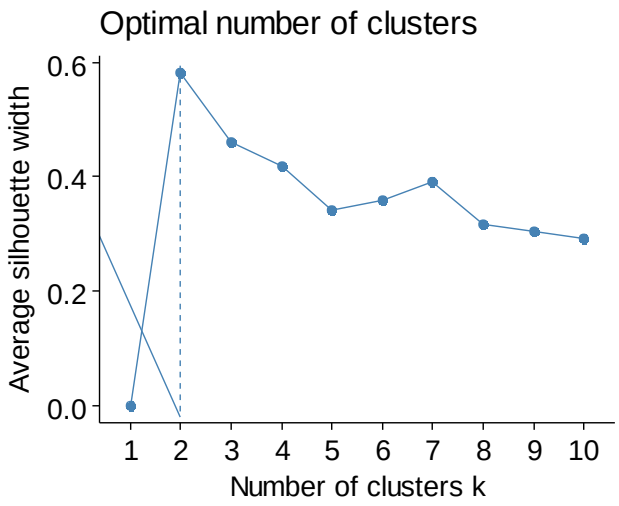
Comparing intra-cluster variation to variation in random case

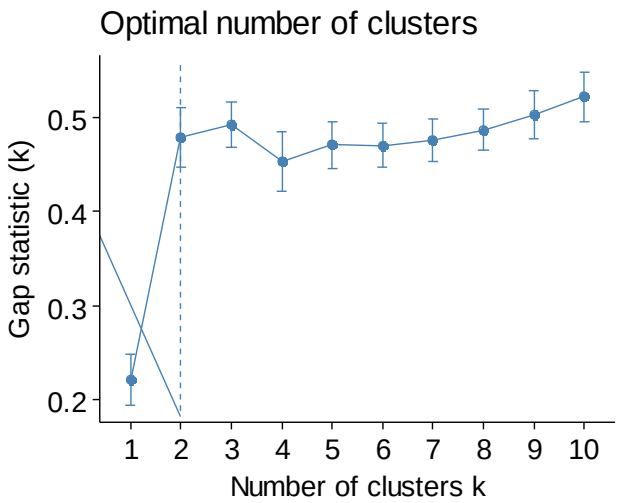


Number of Clusters

```
library(cluster)
library(factoextra)
library(NbClust)
fviz_nbclust(X, pam, method = "wss")
fviz_nbclust(X, pam, method = "silhouette")
fviz_nbclust(X, pam, method = "gap_stat")
```







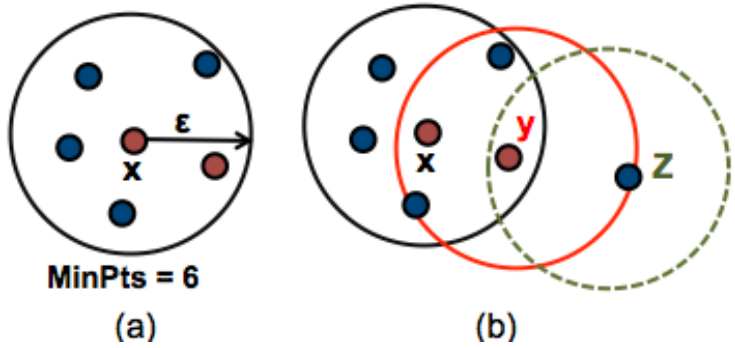
Density-based Clustering

DBSCAN

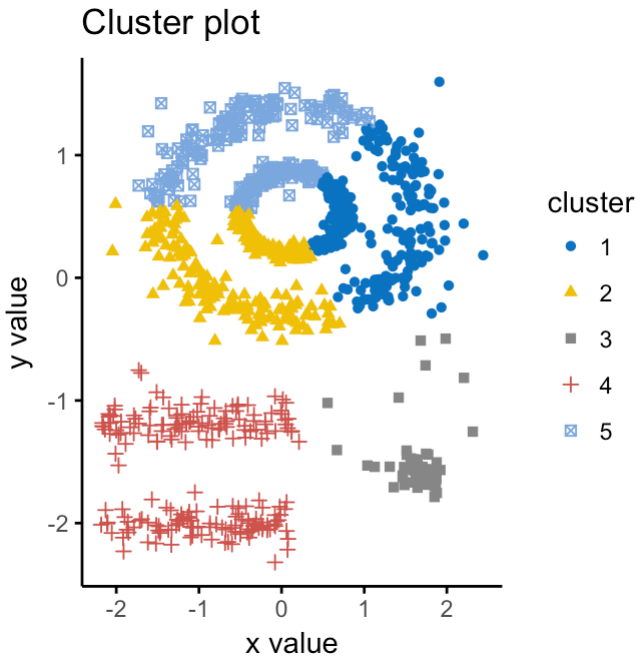
Important parameters:

- Epsilon (ε)
- Minimal number of points (MinPts)

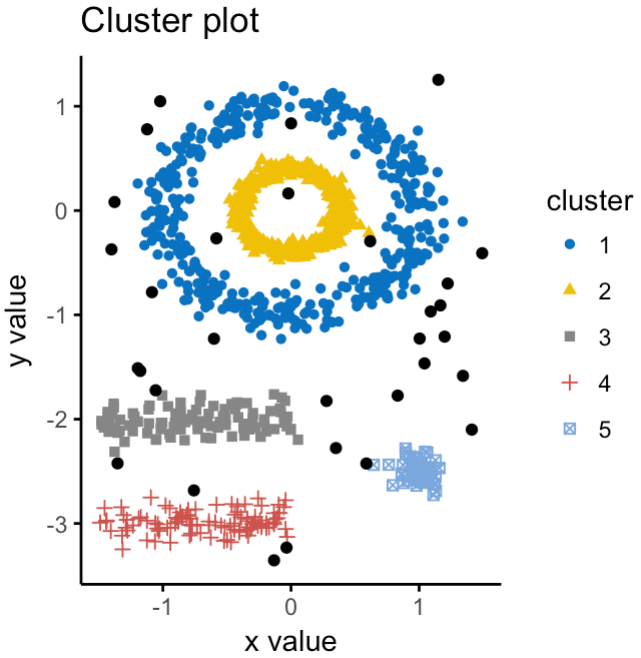
No need to define number of clusters!



kmeans



DBSCAN



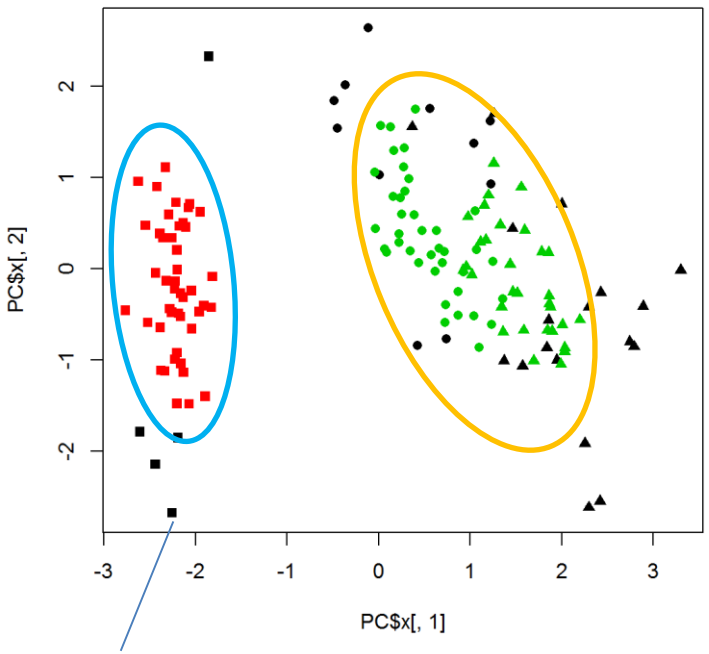
http://www.sthda.com/english/articles/30-advanced-clustering/105-dbscan-density-based-clustering-essentials/

Density-based Clustering

DBSCAN

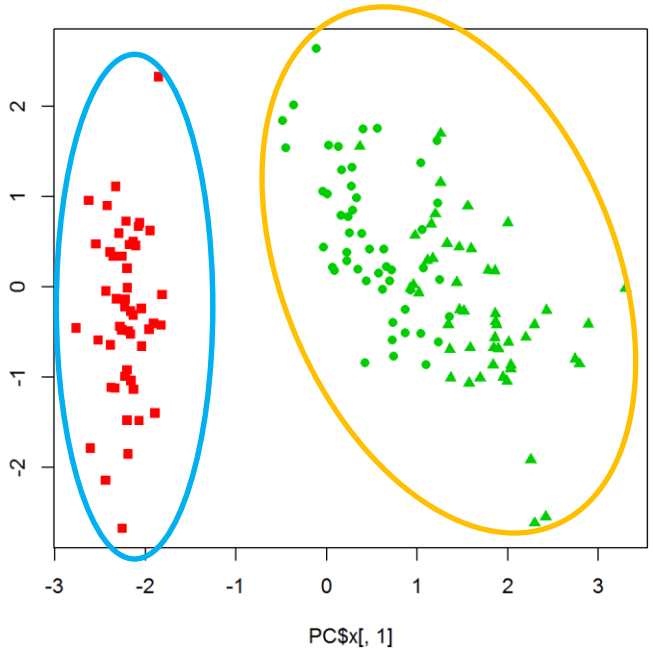
```
library(dbscan)
res = dbscan(X,eps=0.5,minPts=5)
res
plot(PC$x[,1],PC$x[,2],col = 1+res$cluster,pch=point)
```

eps=0.5



unclustered outliers

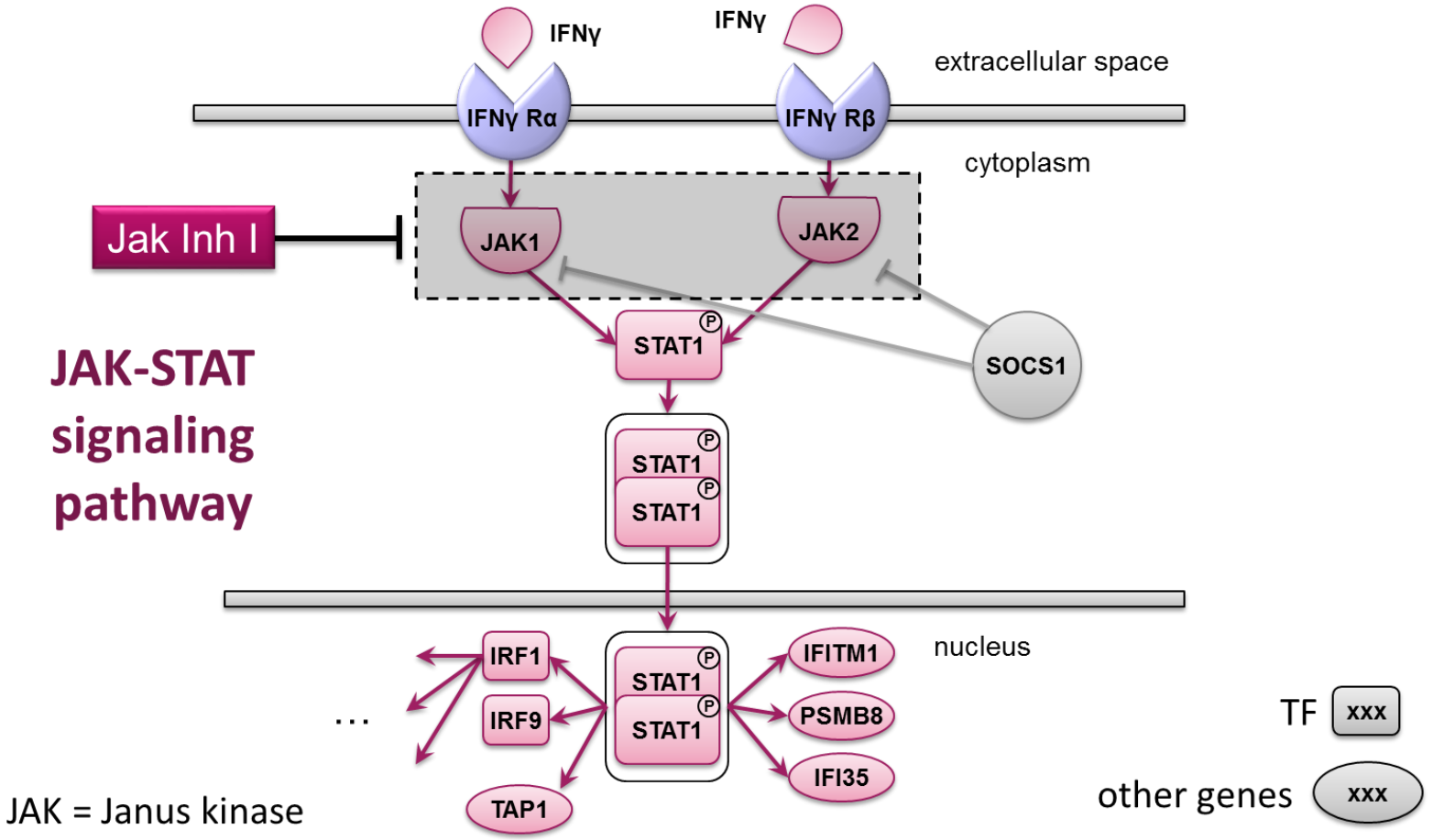
eps=1



Issues:

- Number of clusters strongly depends on eps
- Method assumes similar density of points in clusters

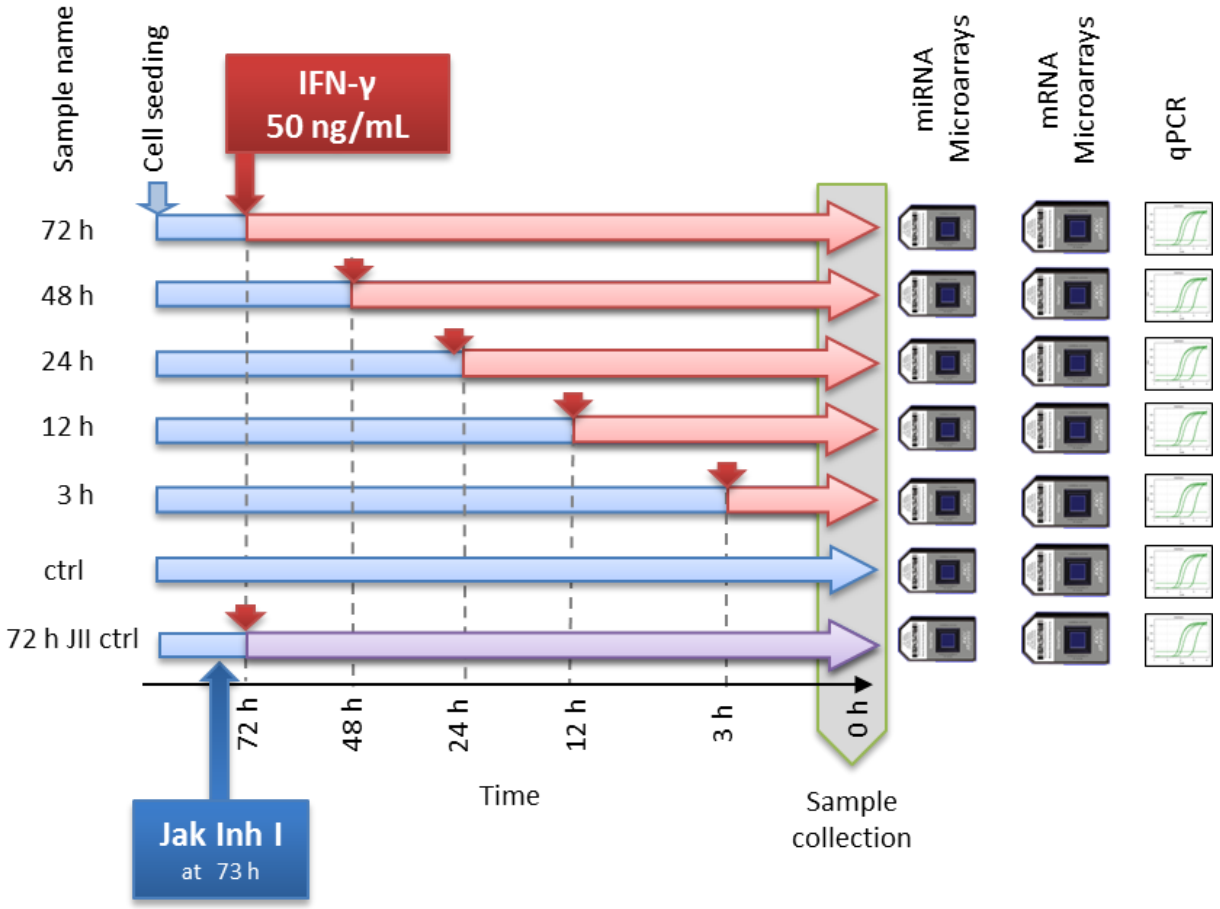
Experiment: IFNg stimulation of melanoma cells



Nazarov, Reinsbach et al (2013) Nucleic Acid Research

Clustering Tutorial

Human melanoma **A375** cells were seeded together and cultured until sample collection. Cells were IFNγ stimulated at different time points.



Nazarov, Reinsbach et al (2013) Nucleic Acid Research

Practical Clustering

LUXEMBOURG INSTITUTE OF HEALTH
RESEARCH DEDICATED TO LIFE

http://edu.sablab.net/nmbu2018/

Dataset: http://edu.sablab.net/data/txt/mRNAIFNg.txt
It contains annotated genes in rows and samples in columns, values are log2 transformed expressions.
NOTE: Depending on the task you can consider samples as objects (gene expression are features then) or genes as objects (expression in samples are features).

Tasks

1. **Import the data** (see online materials for help)
   a. Prepare **matrix X** with gene expressions removing lowly expressed and non-annotated features (GeneSymbol is "")
   b. Create standardized gene expression **matrix Z**, so that all genes have mean = 0, st.dev. = 1
   c. Perform and plot PCA of samples and genes (use both X and Z)
2. **Cluster the samples** (expected outputs are presented in online materials)
   a. Use `pheatmap()` to make bi-cluster of genes and samples (for X and Z)
   b. Cluster the samples using k-means or PAM and define the reasonable number of clusters. Visualize in PCA plot. Any difference when using X or Z matrices?
3. **Cluster the genes** (expected outputs are presented in online materials)
   a. Use k-means or PAM methods to cluster the genes from standardized Z matrix. Visualize them on corresponding PCA plot (genes as objects, samples as features).