Biological Data Analysis

Andreas Hadjiprocopis

29/11/2017

Building and exploring Protein Interaction Networks from Mass Spectrometry / Microarray data using public databases

Typical output from MS

Accession	Description	Heavy/Ligh
Thymidine kinase, cytosolic (EC 2.7.1.21) [Source:UniProtKB/Swiss-Prot;Acc:P04184]	Tk1| ENSMUST00000026661| ENSMUSG00000025574 .	6.261
Ribonuclease P protein subunit p30 (RNaseP protein p30)(EC 3.1.26.5)(RNase P subunit 2) [Source:UniProtKB/Swiss-Prot;Acc:O88796]	Rpp30| ENSMUST00000025714| ENSMUSG00000024800	4.898
Secretory carrier-associated membrane protein 3 (Secretory carrier membrane protein 3) [Source:UniProtKB/Swiss-Prot;Acc:O35609]	| ENSMUST00000098941| ENSMUSG00000028049	4.673
Protein Hook homolog 3 [Source:UniProtKB/Swiss-Prot;Acc:Q8BUK6]	Hook3| ENSMUST00000037182| ENSMUSG00000037234	4.492
60S ribosomal protein L23 [Source:UniProtKB/Swiss-Prot;Acc:P62830]	Rpl23| ENSMUST00000103146| ENSMUSG00000071415 . .	4.310
Ubiquitin-conjugating enzyme E2 G1 (EC 6.3.2.19)(Ubiquitin-protein ligase G1)(Ubiquitin carrier protein G1)(E217K)(UBC7) [Source:UniP	Ube2g1| ENSMUST00000021148| ENSMUSG00000020794	0.556
Phosphoglycerate mutase 1 (EC 5.4.2.1)(EC 5.4.2.4)(EC 3.1.3.13)(Phosphoglycerate mutase isozyme B)(PGAM-B)(BPG-dependent PGAM	Pgam1| ENSMUST00000011896| ENSMUSG00000011752	0.555
Casein kinase II subunit alpha' (CK II)(EC 2.7.11.1) [Source:UniProtKB/Swiss-Prot;Acc:O54833]	Csnk2a2| ENSMUST00000056919| ENSMUSG00000046707 .	0.552
Protein LAP4 (Protein scribble homolog) [Source:UniProtKB/Swiss-Prot;Acc:Q80U72]	| ENSMUST00000063747| ENSMUSG00000022568	0.548
Glutathione reductase, mitochondrial Precursor (GRase)(GR)(EC 1.8.1.7) [Source:UniProtKB/Swiss-Prot;Acc:P47791]	Gsr| ENSMUST00000033992| ENSMUSG00000031584 .	0.545
EH domain-containing protein 2 [Source:UniProtKB/Swiss-Prot;Acc:Q8BH64]	Ehd2| ENSMUST00000098799| ENSMUSG00000074364	3.454
182 kDa tankyrase-1-binding protein [Source:UniProtKB/Swiss-Prot;Acc:P58871]	Tnks1bp1| ENSMUST00000111605| ENSMUSG00000033955	3.413

Typical spreadsheet output from MS

Multiple data sources - work at Erler Lab

* It is possible to have more than one data sources yielding a multi-dimensional node properties.
* For example, mass spectrometre, micro-arrays, MALDI.

Journal of proteome research

Article

pubs.acs.org/jpr

Identification of Hypoxia-Regulated Proteins Using MALDI-Mass Spectrometry Imaging Combined with Quantitative Proteomics

Marie-Claude Djidja,[†,#,◆] Joan Chang,[†,∥,◆,⊥] Andreas Hadjiprocopis,[†] Fabian Schmich,[†,▽] John Sinclair,[‡] Martina Mršnik,[†,○] Erwin M. Schoof,[§] Holly E. Barker,[†] Rune Linding,[§] Claus Jørgensen,[‡] and Janine T. Erler*[,†,∥,⊥]

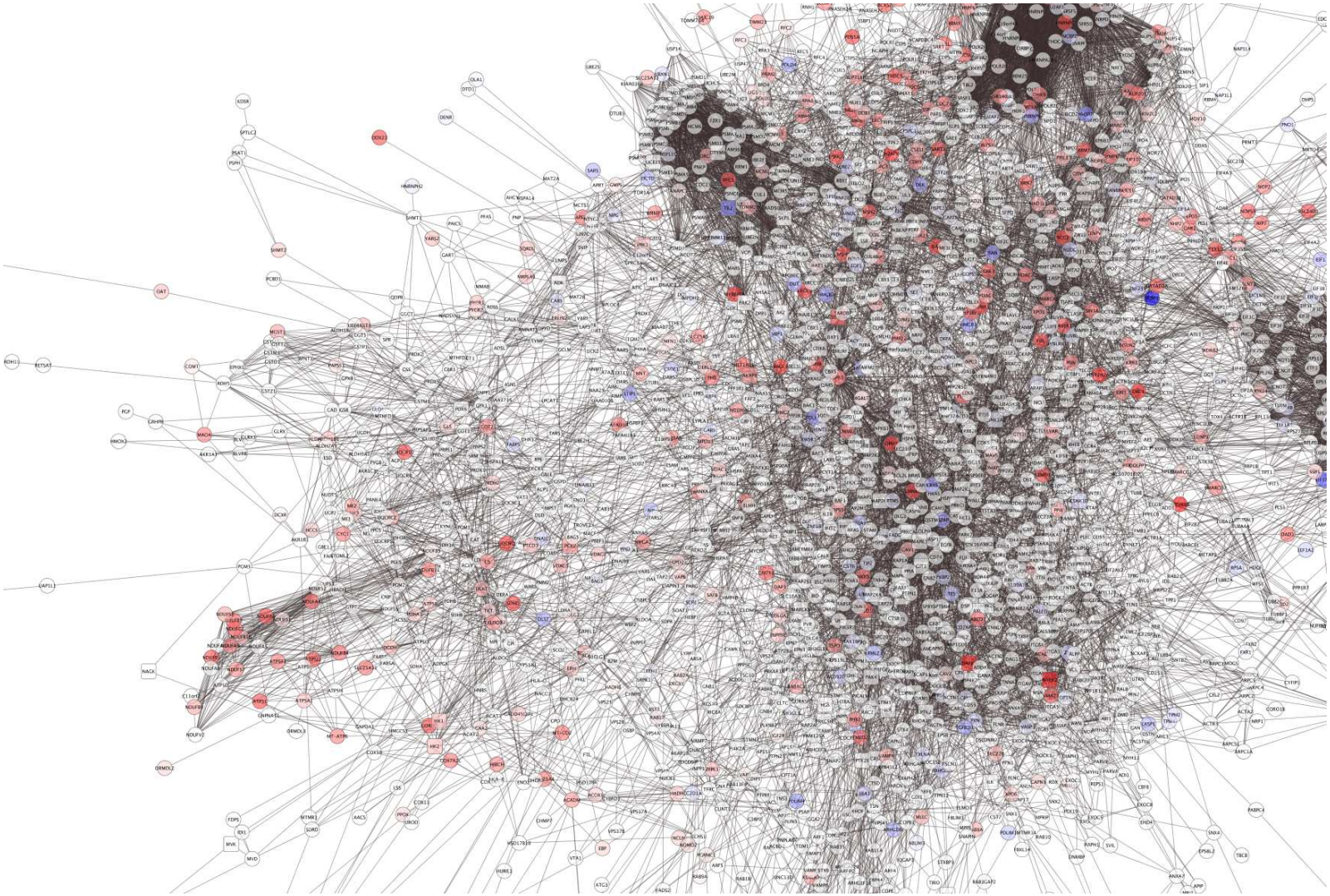
[†]Hypoxia and Metastasis Team and [‡]Cell Communications Team, Cancer Research U.K. Tumour Cell Signalling Unit, Division of Cancer Biology, The Institute of Cancer Research, London, United Kingdom

[§]Cellular Signal Integration Group (C-SIG), Centre for Biological Sequence Analysis, Department of Systems Biology, Technical University of Denmark, DK-2800 Lyngby, Denmark

[⊥]Biotech Research and Innovation Centre (BRIC), University of Copenhagen, Ole Maaløes Vej 5, Copenhagen 2200, Denmark

Aims

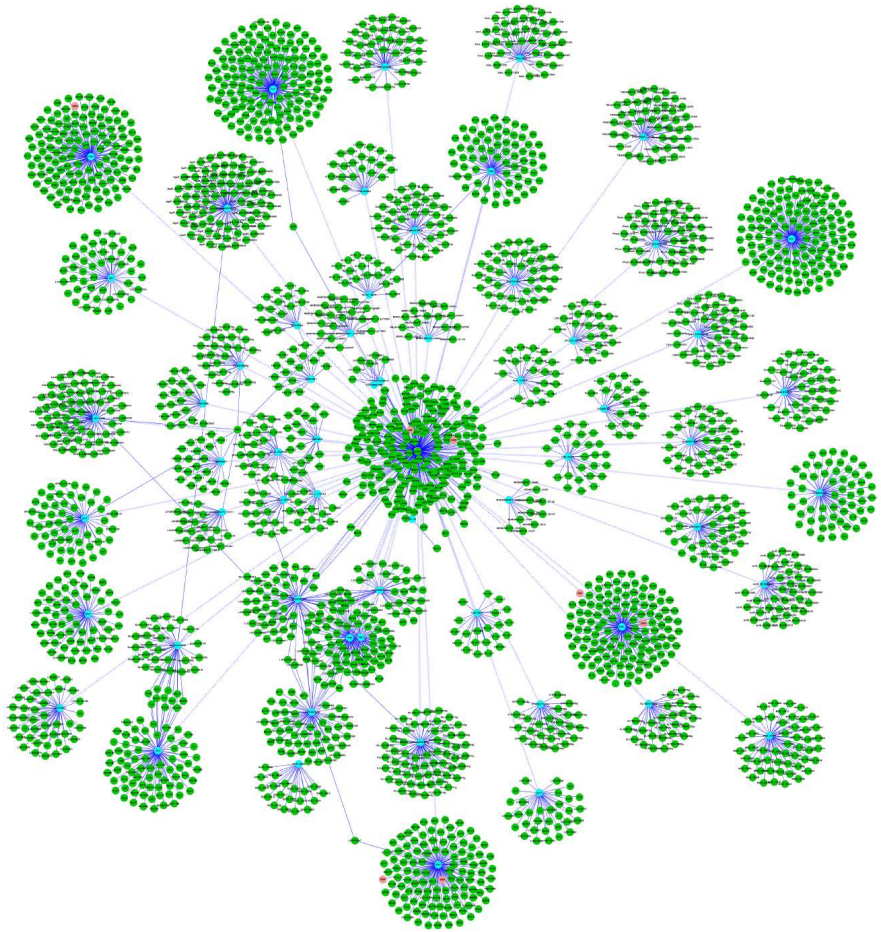
1. To build a network of interactions between detected proteins.



A typical network

Aims

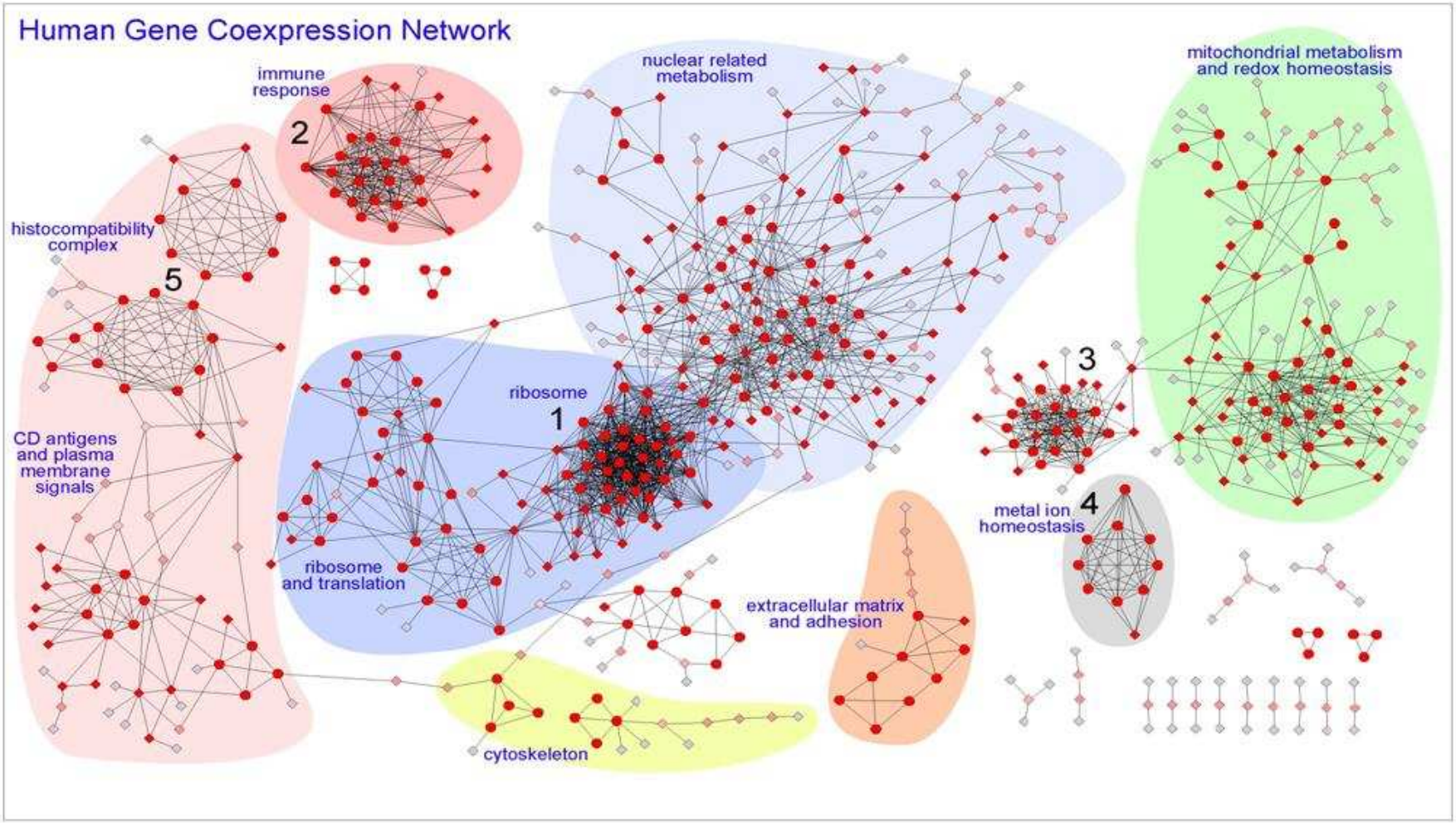
2. To group nodes together according to various criteria – e.g. similar expression.



nodes with similar expression are clustered together

Aims

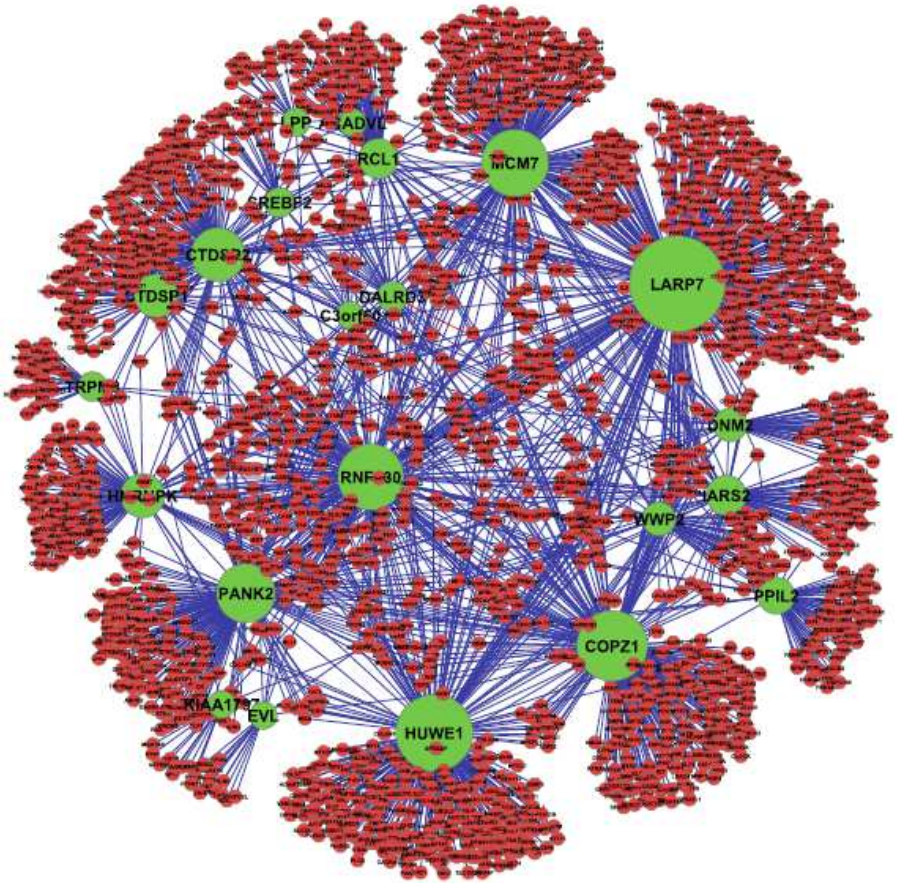
3. ... or to group based on similarity of protein function / ontology.

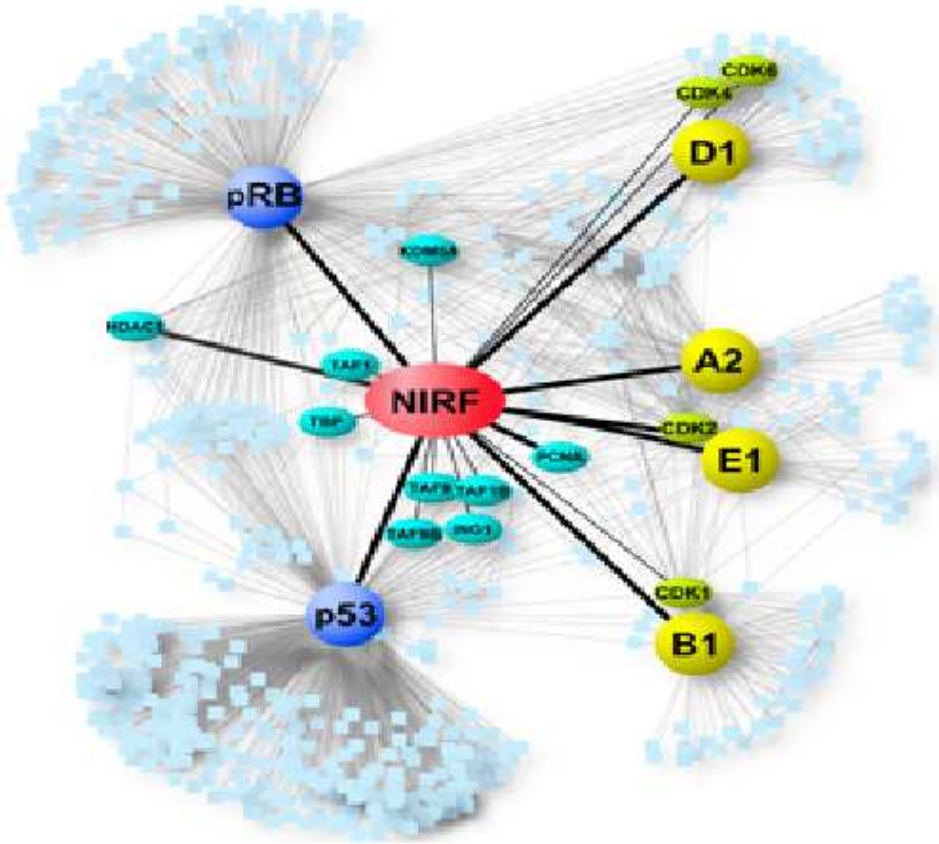


nodes with similar <u>function</u> are clustered together

Aims

4. To identify the hubs – the nodes linking groups together.
5. To see the interactions between and within the groups.





to see hubs and interactions between/within groups

Data integration with information from public databases

* There are numerous public databases with information about proteins and their interactions.
* Integrating the experimental data (e.g. differential expression) with such information yields a powerful method for gaining an insight, for example by knowing the function of each protein or the type of interactions between pairs.
  - STRING-db: `www.string-db.org`
  - Gene Ontology, GO: `www.geneontology.org`
  - KEGG : `www.genome.jp/kegg`
  - PANTHER-db : `www.pantherdb.org`
* Reliability varies.
* Especially when text-mining (from publications) is the source of evidence.

The STRING-db database

* STRING is a database of protein associations.
* For a given protein, STRING returns a list of other proteins associated with it.
* Evidence for these associations are provided by:
  - fusion (detected fusion between two genes indicates that their protein products may physically interact or be involved in the same pathway),
  - systematic co-expression analysis makes use of all microarray gene expression experiments deposited at NCBI.
  - knowledge from other databases, gene ontology and high-throughput experiments,
  - co-occurence in literature / text-mining.

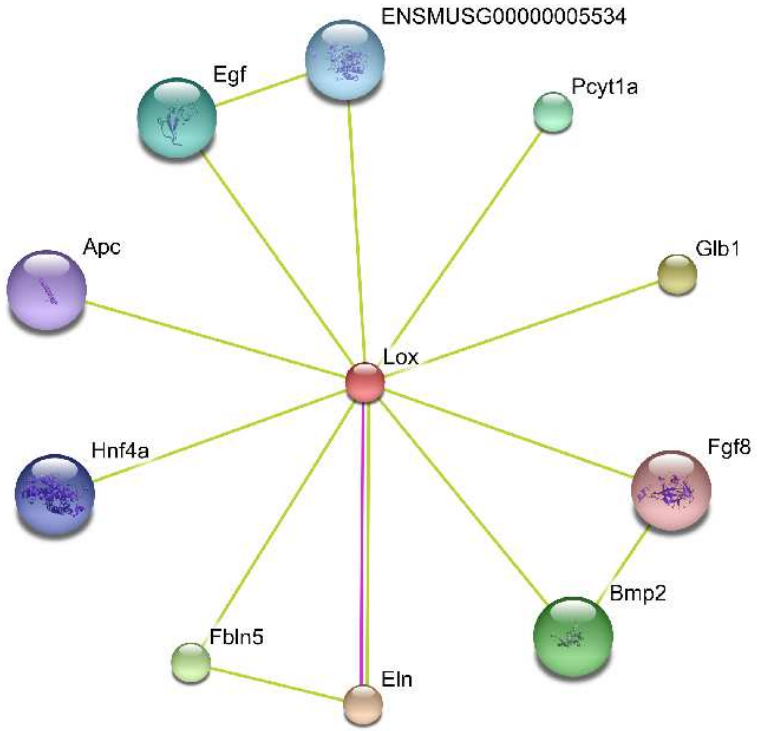
  and are accompanied by some confidence score.
* Szklarczyk D, Morris JH, Cook H, et al. The STRING database in 2017: quality-controlled protein–protein association networks, made broadly accessible. Nucleic Acids Research. 2017

The STRING-db database : network based on evidence

Your Input:

Lox

Predicted Function	Neighborhood	Gene Fusion	Cooccurrence	Coexpression	Experiments	Databases	Textmining	[Homology]	Score
Eln							•		0.966
Glb1							•		0.940
Fbln5							•		0.929
Bmp2							•		0.918
Pcyt1a							•		0.904
Egf							•		0.900
ENSMUSG00000005534							•		0.887
Hnf4a							•		0.882
Apc							•		0.881
Fgf8							•		0.877

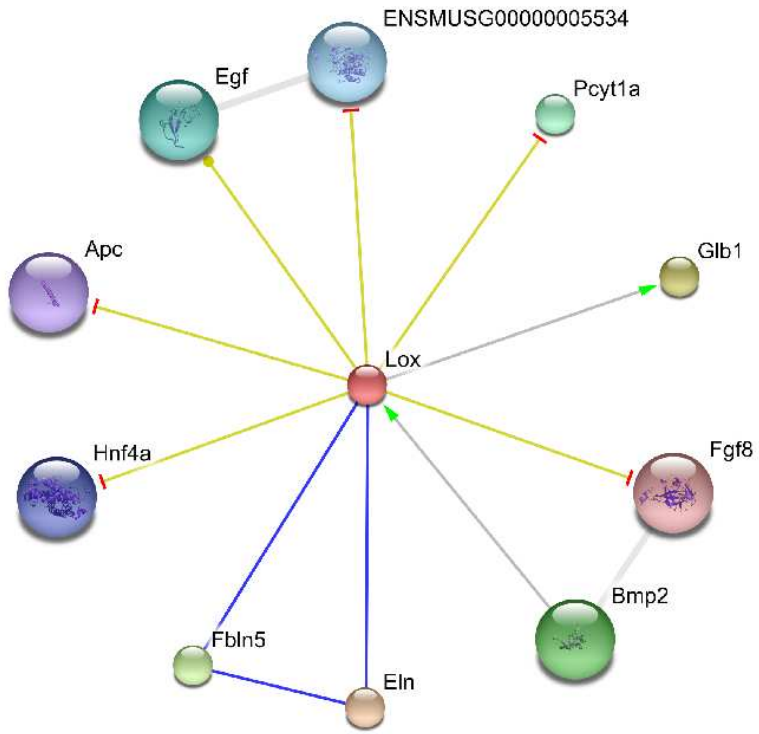


The STRING-db database : network based on actions probably with direction and causality

Your Input:

Lox

Predicted Functior	Activation	Inhibition	Binding	Phenotype	Catalysis	Post-transl. m	Reaction	Expression	Score
Eln			•						0.966
Glb1	•								0.940
Fbln5			•						0.929
Bmp2	•								0.918
Pcyt1a								•	0.904
Egf								•	0.900
ENSMUSG0000000553								•	0.887
Hnf4a								•	0.882
Apc								•	0.881
Fgf8								•	0.877



Nature and direction (arrows) of protein interactions, e.g.: Activation, Inhibition, Binding, Phenotype, Catalysis, Post-translational modification, Reaction, Expression.

Tools : Computational Analysis

* In the experiment which gave rise to this analysis (Prof Chris Marshall, Faraz Mardakheh), MS detected about 2,500 nodes which yielded about 18,000 direct associations.
* Visualisation of large networks using available software has its limitations. Interactiveness is a plus when exploring but a minus when we actually want to produce something.
* So the process can be slow and often unresponsive or crashes with large networks.
* There are possibly long waiting hours.
* Sometimes we forget what steps we took or what parameters we used. There are different files to load and save manually.
* Cluster computers usually do not allow for graphic interfaces.

Tools : Computational Analysis

* All in all, it is fine to experiment with but difficult to arrive to conclusions.
* In my opinion there are huge benefits in automating this process in what we call *pipelines*.
* In what follows I will outline some methods I have developed for analysing this data in order to answer specific questions and drive further experiments in the lab.

Finding hubs and knock-down scenaria

* Nodes which have a large number of interactions are called hubs. For this reason they are interesting: knocking them down can alter the network dramatically.
* Work-in-progress: follow the paths originating from hubs and see what happens to the network when a hub is switched off.
* Alternatively, find the hub responsible for switching off a part of the network or a single protein. Report side-effects, i.e. what else will be affected.
* It is easy to query the network in this way with a tool I already have for mapping a network to a Graph (the mathematical construct).

Challenges

* The mathematical fields covering this kind of problems are Graphical Models and Graph Theory. And they are quite mature.
* The challenge is to adapt the existing algorithms to the gigantic size of the Graphs and the parallel computational machinery in our disposal.
* There are plenty of algorithms to search a Graph. And in parallel.
* Some algorithms are already implemented for GPU (graphics cards) processing which improves speed quite a lot.

Adding confidence to the Graph

* Enquiries to biological databases (e.g. STRING-db) often yield a probability, a confidence of how good the evidence was, or how contradicting were the published results for that matter.
* This probability should be used in PPI network building.
* For example we already filter out information with less confidence (score) than a cut-off value.
* Or we build a network for each score cut-off value.
* A more dynamic approach is to use Probabilistic Graphical Models which incorporate the score as *edge strength*.
* This is more important that it sounds because it will answer each query with not only with a combined confidence but it will also identify the links with the least confidence. If the result is important but the confidence is low, then we can manually investigate those weakest links even by conducting an experiment ourselves.

Clustering

* Clustering assigns a node to a group depending on its attributes, e.g. expression, its interaction with other nodes, functional group, etc.
* Our aim is to layer the complexity of the network by iterative clustering.
* i.e. cluster the clusters of the clusters ...
* At each layer there will be a number of clusters, a number of hubs and their interactions.
* For example, the network can be reduced to 324 clusters, 20 hubs and only 4,168 connections between them at the first layer.
* It is a way to reduce complexity, to abstract the network but also be able to dive in when needed.
* Hopefully, at some of these layers, the clusters and hubs will coincide with Biology.

Assessment of Clustering

* Is the grouping of nodes statistically significant? In other words, do the nodes within a given cluster have significant differences compared to the whole network? Basically, find the *p-value* of the cluster.
* Differences in mean expression for example. But this is not as simple as it sounds because we can have multi-dimensional expression data.
* Also note that clustering often depends on initial conditions and alternative clusterings usually exist.
* Conventional statistical tests (e.g. t-test) assume Normal distributions. For this reason I prefer to use Bootstrapping / Cross Validation / permutation tests in order to assess statistical significance.

Assessment of Clustering : Bootstrapping

* For the given cluster of N nodes to assess, we repeatedly draw N random nodes from the whole network, i.e. we form a random cluster of the same size.
* Total mean expression in the network is compared to that of the random cluster.
* The more these differences approach that of the cluster we test, the more its *p-value* increases.

Assessment of Clustering

fold change vs statistical significance for cluster and network

The missing links

* In experimental setups such as MS, MALDI-MSI, microarray assays, a list of detected proteins is obtained.
* Integrating with functional association data, such as from STRING-DB, yields relationships between detected proteins A and Z, of the form $A \xrightarrow{0.7} Z$.
* These relationships are useful in **linking protein clusters** (via any of their member proteins) or hubs.
* Simply, enquire the Graph about functional relationships between all possible pairs of proteins of the two clusters and find the bridges.
* Or just look at the network for these bridges.

The missing links

* **Does it mean that clusters or hubs without any direct protein bridge are not linked?**
* No. On the one hand, bridges can be *undetected* proteins, therefore not contained in the network even after integrating with STRING.
* On the other hand, there may be bridges which are chains of more than one protein.
* An algorithm for finding those will be better than doing it manually: read data from STRING, convert it into a Graph and then query it.

The missing links

* Enter the power of computer programming and the importance of affording tailor-made **algorithmic** and **computational** solutions.
* We have created such a solution which searches protein relationships one- and two-steps away, possibly more – mileage varies with resources.
* For example, in cases when protein A does not connect to Z, the algorithm tries to find protein B such that: $A \xrightarrow{0.7} B \xrightarrow{0.9} Z$ (one-step link) or B and C such that: $A \xrightarrow{0.7} B \xrightarrow{0.8} C \xrightarrow{0.9} Z$ (two-step link).
* **In this way, possible indirect protein cluster relationships can be found via detected or undetected proteins and with specific number of steps/hops.**

Example

* Here the question was: we are interested in LOX but it has not been detected. Report all the 2-step chains starting and ending with a detected protein which contain LOX.

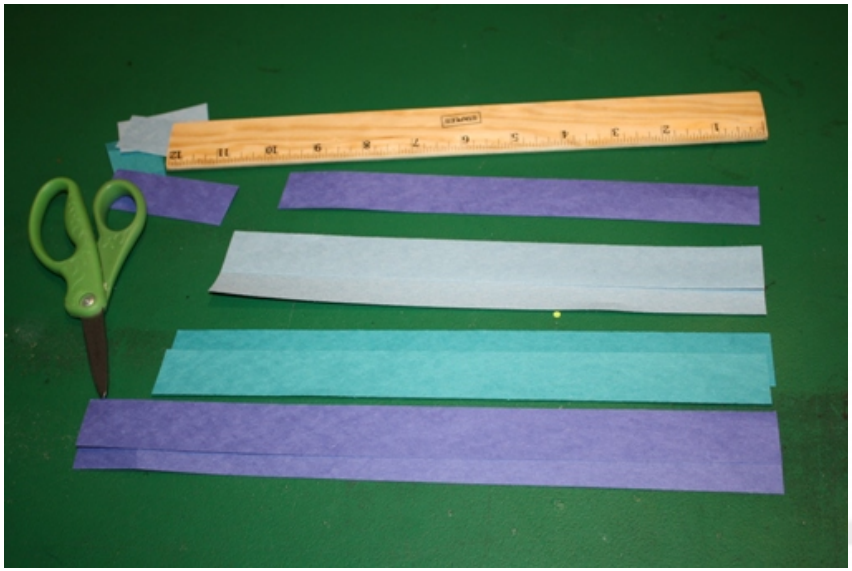
```
CFLAR(1.66) --(0 0 0 0 0 0 162 162)--> NOTCH2() --(0 0 0 0 0 0 377 377)--> LOX() --(0 0 0 0 0 0 223 223)--> BCL10(1.55)
CFLAR(1.66) --(0 0 0 0 0 0 165 165)--> FGFR3() --(0 0 0 0 0 0 300 300)--> LOX() --(0 0 0 0 0 0 223 223)--> BCL10(1.55)
CFLAR(1.66) --(0 0 0 0 0 0 165 165)--> PTEN() --(0 0 0 0 0 0 798 798)--> LOX() --(0 0 0 0 0 0 223 223)--> BCL10(1.55)
CFLAR(1.66) --(0 0 0 0 0 0 167 167)--> CDKN1A() --(0 0 0 0 0 0 275 275)--> LOX() --(0 0 0 0 0 0 223 223)--> BCL10(1.55)
CFLAR(1.66) --(0 0 0 0 0 0 171 171)--> PTK2() --(0 0 0 0 0 0 600 600)--> LOX() --(0 0 0 0 0 0 223 223)--> BCL10(1.55)
UGP2(1.57) --(0 0 0 199 0 0 0 199)--> EFHA1() --(0 0 0 0 0 0 302 302)--> LOX() --(0 0 0 0 0 0 223 223)--> BCL10(1.55)
UGP2(1.57) --(68 0 0 132 0 0 118 187)--> PKLR() --(0 0 0 0 0 0 230 230)--> LOX() --(0 0 0 0 0 0 224 224)--> CFLAR(1.66)
ZNF132(1.78) --(0 0 0 0 0 0 409 409)--> BAX() --(0 0 0 0 0 0 215 215)--> LOX() --(0 0 0 0 0 0 223 223)--> BCL10(1.55)
ZNF132(1.78) --(0 0 0 0 0 0 409 409)--> BAX() --(0 0 0 0 0 0 215 215)--> LOX() --(0 0 0 0 0 0 224 224)--> CFLAR(1.66)
ZNF132(1.78) --(0 0 0 0 0 0 567 567)--> CDKN1C() --(0 0 0 0 0 0 287 287)--> LOX() --(0 0 0 0 0 0 224 224)--> CFLAR(1.66)
```

* Another question can be to list all indirect links between CFLAR and BCL10 which contain only detected proteins.
* Future work: allow for specifying constraints on confidence of interactions or arbitrarily fixing some of the missing links via something like: LOX must exist in in any chain of 2 steps and the other must be a detected protein.

Analysis of live cell images acquired by high-throughput imaging

A quiz first

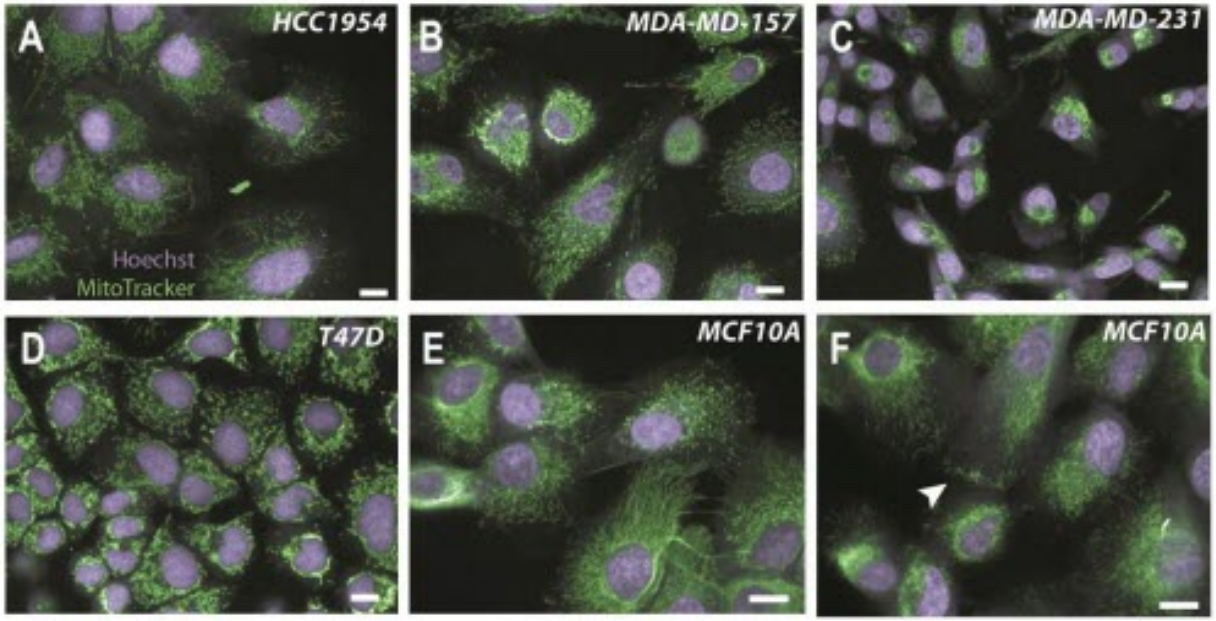
* Thousands of children are given each a strip of paper of random length and asked to make a rectangular box from it with (uniformly) random dimensions between, say, 10cm and 20cm.
* What is the distribution of the boxes volume?



Perkin-Elmer Opera

* The microscope is able to track thousands of cells living in a petri dish.
* Images of each cell are taken at regular intervals.
* 3rd-party software extracts morphological and texture features from the cell images.

Cell morphology and texture features data



* Data to be analysed consists of a matrix (spreadsheet) where rows correspond to different cells and columns to cell features.
* Some features in the data: *nucleus area*, *cytoplasm area*, *cell width-to-length ratio*, *cell skeleton body area*, *nuclear roundness*.
* Tagging technology allows for the quantification of NFκB expression in nucleus and cytoplasm.
* Overall, 60 to 100 features (more in future releases) that describe the geometry, texture and NFκB expression of each cell over time.

Cell morphology and texture features data

* For some particular experiment (Chris Bakal, Julia Sero), we had data available corresponding to
  - 20 different cell lines,
  - 10 different treatment conditions, e.g. EGF, hydrocortisone, TNF-α,
  - 1- and 5-hour treatment duration.
* For another experiment (Janine Erler, Joan Chang) we had two cell lines (with knockdowns making them 4) in Hypoxia or Normoxia for 24, 48 and 72 hours, with and without DHE.

Aims

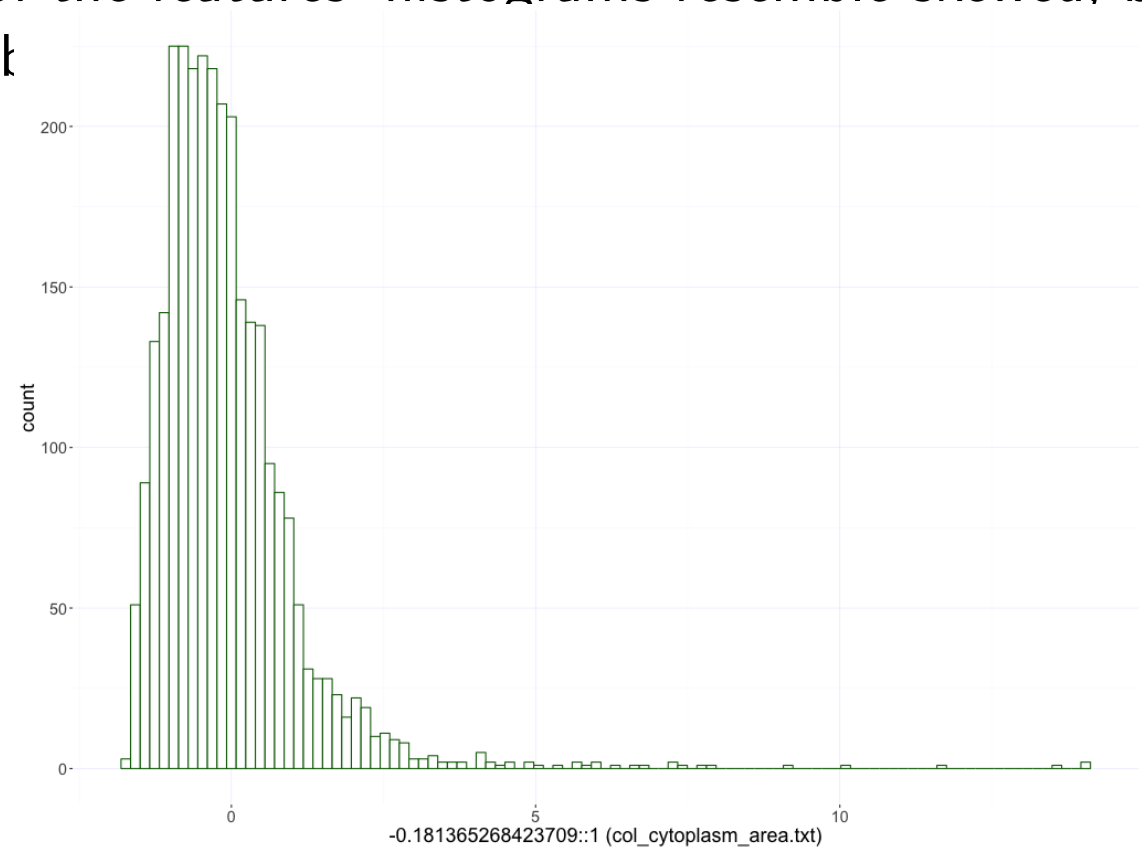
The aim of this experiment with the then newly acquired microscope was to develop techniques and then implement them as "pipelines" in order to be able to perform the following tasks:

* Estimate the statistical significance of the difference between features of two groups (e.g. do two specific cell lines differ wrt feature *nucleus area*?).
* Are two features of cells within a single group totally independent or are there certain feature values or ranges of values that frequently occur together?
* Estimate the separability of the available features between two different groups.
* Can feature values be used to *predict cell line* and/or *treatment* and/or *NFκB expression*?

Statistical significance of the difference between features

Bootstrapping was used in this because we did not want to make any assumptions about the generating distribution and because computational resources allowed so.

In fact, most of the features' histograms resemble skewed, bell-shape curves and so k



Histogram of *cytoplasm area*

Statistical significance of the difference between features

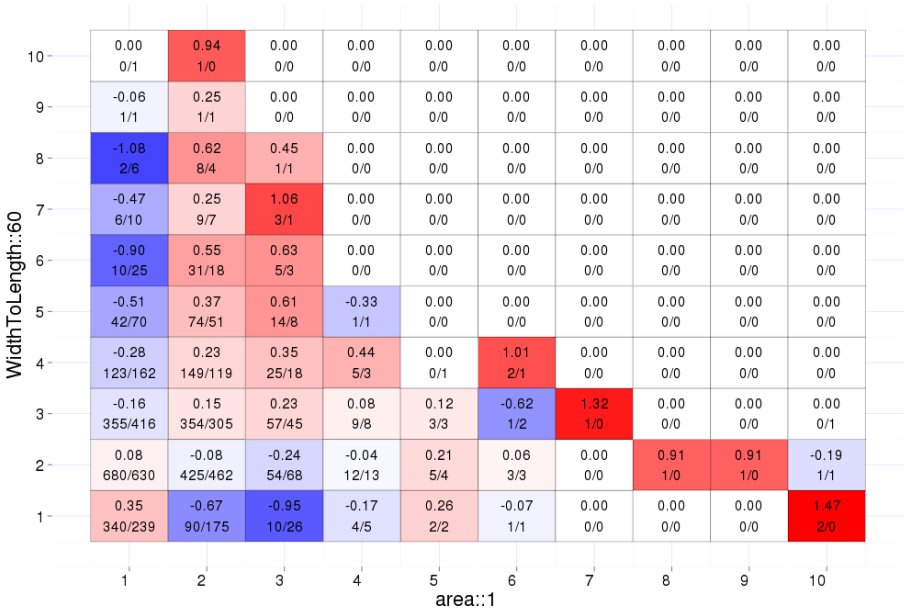
* Bootstrapping involves sampling with replacement from the two groups, and recording the differences in the means of the samples each time.
* Confidence intervals can be estimated by plotting the histogram of the differences of the means and recording the cuounts at the centiles of interest (e.g. at 5 and 95 %).

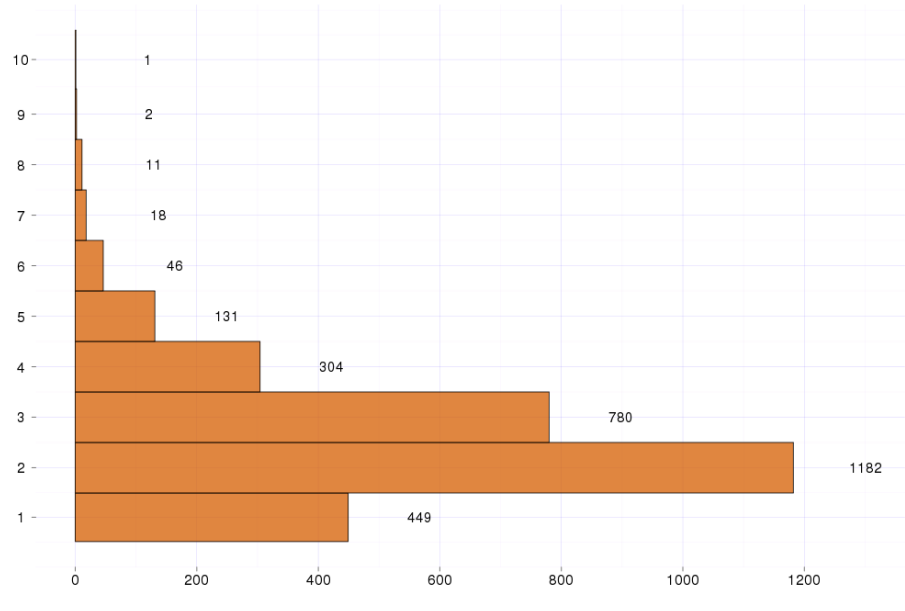
Feature (in)dependence

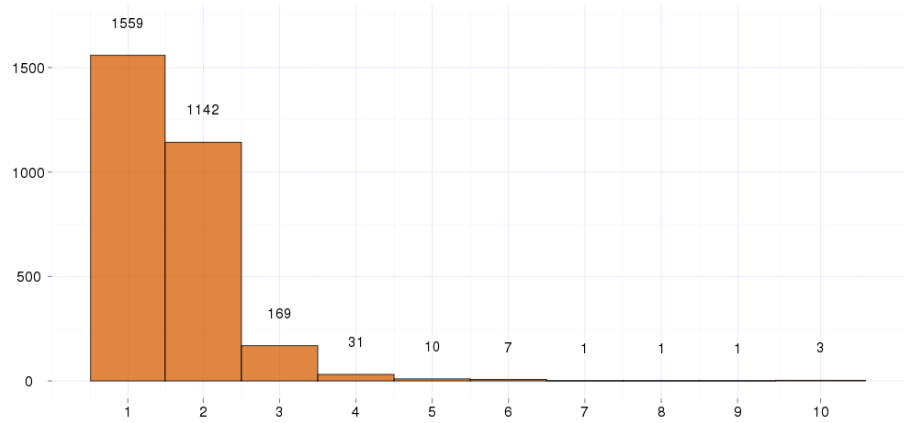
* When two random events are independent the probability of occuring together equals the product of their probabilities occuring, $P(A \cap B) = P(A)\dot{P}(B)$
* Firstly, we construct a histogram for the values of each feature with appropriate breaks.
* Then we construct a 2-dimensional histogram of the two features. That is we bin pairs of feature values.
* Each normalised bin count in the 2D histogram represents the actual probability of co-occurence. Whereas the corresponding product of probabilities in the two 1D histograms the expected probability in the case of independence.
* The less the discrepancy between the two values the more independent the two features are.
* Caveat: a large number of data rows required to populate the 2D histogram.

Feature (in)dependence: Area vs WidthToLength

./plots/data/MDA-MB-231.72h.hypoxia without DHE.txt : area (1) :: WidthToLength (60) (probs)
output=(./plots/plot_feature_correlations_using2dhistograms/MDA-MB-231.72h.hypoxia without DHE/out)





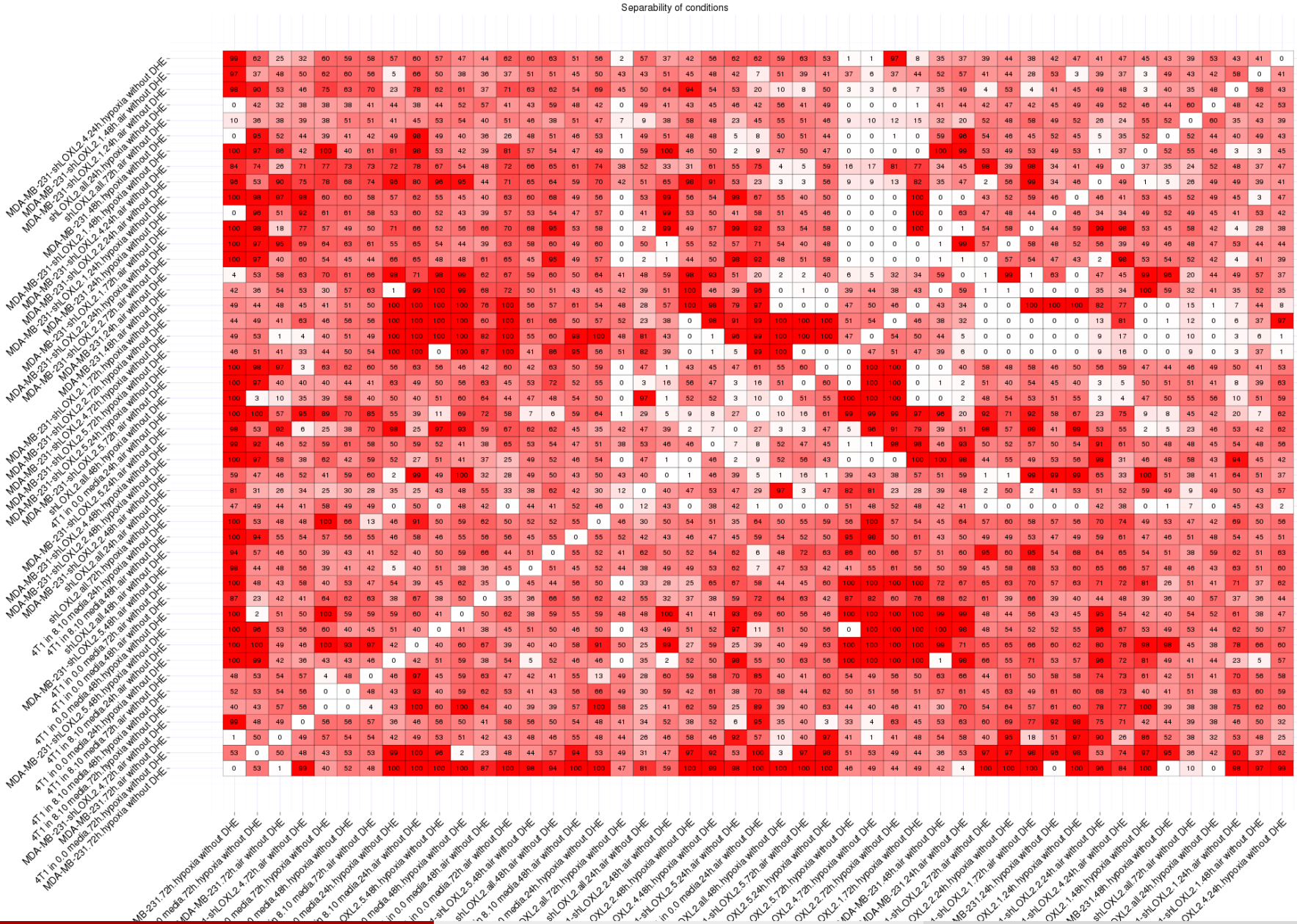


MI:0.129905600684895
COV:0.0173853469637359
Entropy: 1.06
(t-test)pv (sparse/unsparse): 1.00 / 0.99
(levene)pv (sp/unsp): 1.00 / 1.00
box legend: fold change (observed / expected counts)

Separability of features between groups

* Our intention here is to be able to do two things, firstly to select those features which offer discriminatory power and throw away those which do not. But this remains work-in-progress.
* Our second intention was to assess the effect of the different treatments etc. solely on the morphology of the cells. And whether a machine learning classifier could quickly distinguish between the different groups by just looking at them.
* To this end we have used a very quick-to-train machine learning classifier, Support Vector Machines. to train on 60% of the data. Its performance on the uknown 40% data would be an indication of separability.

Separability of features between groups

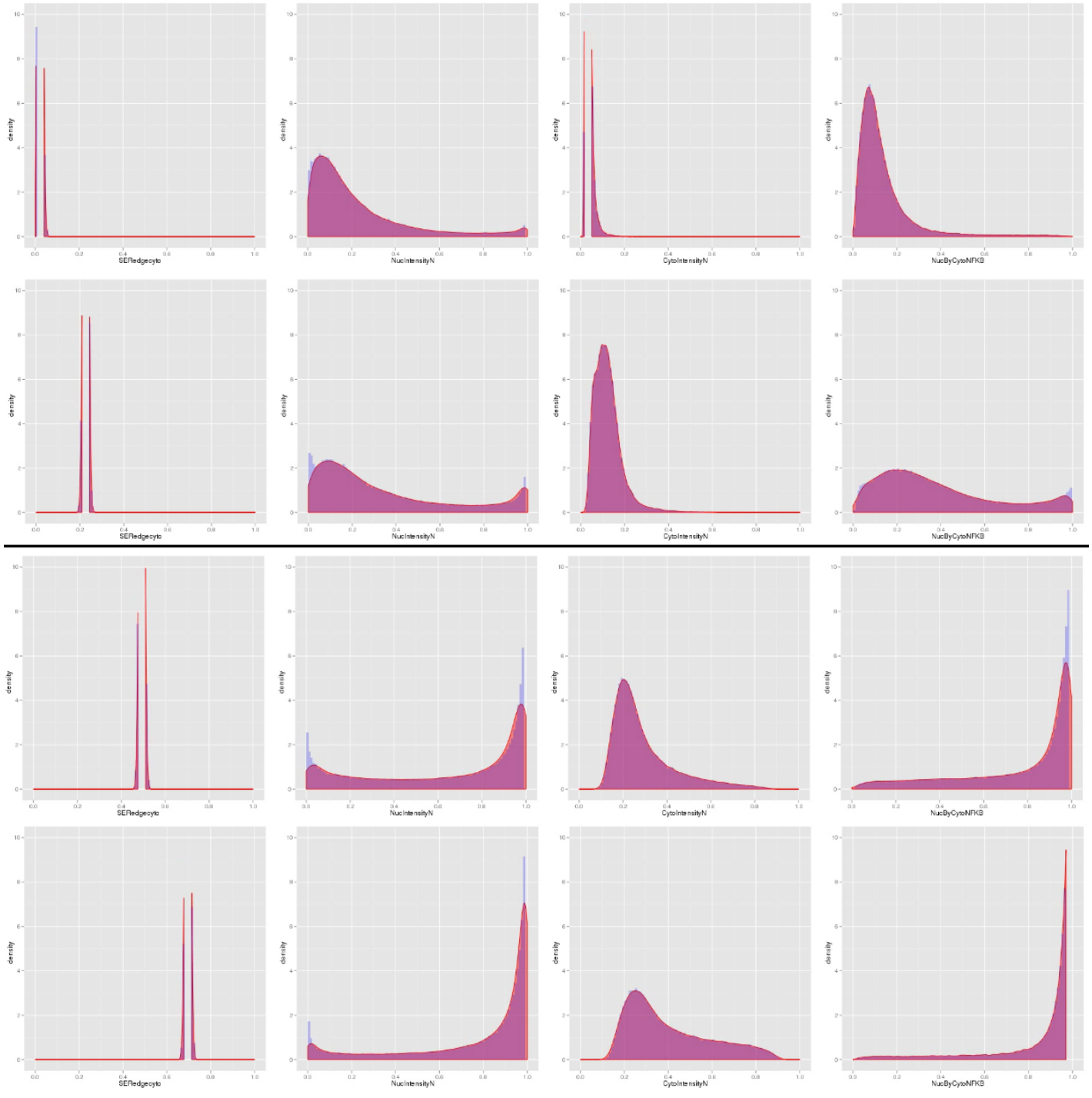


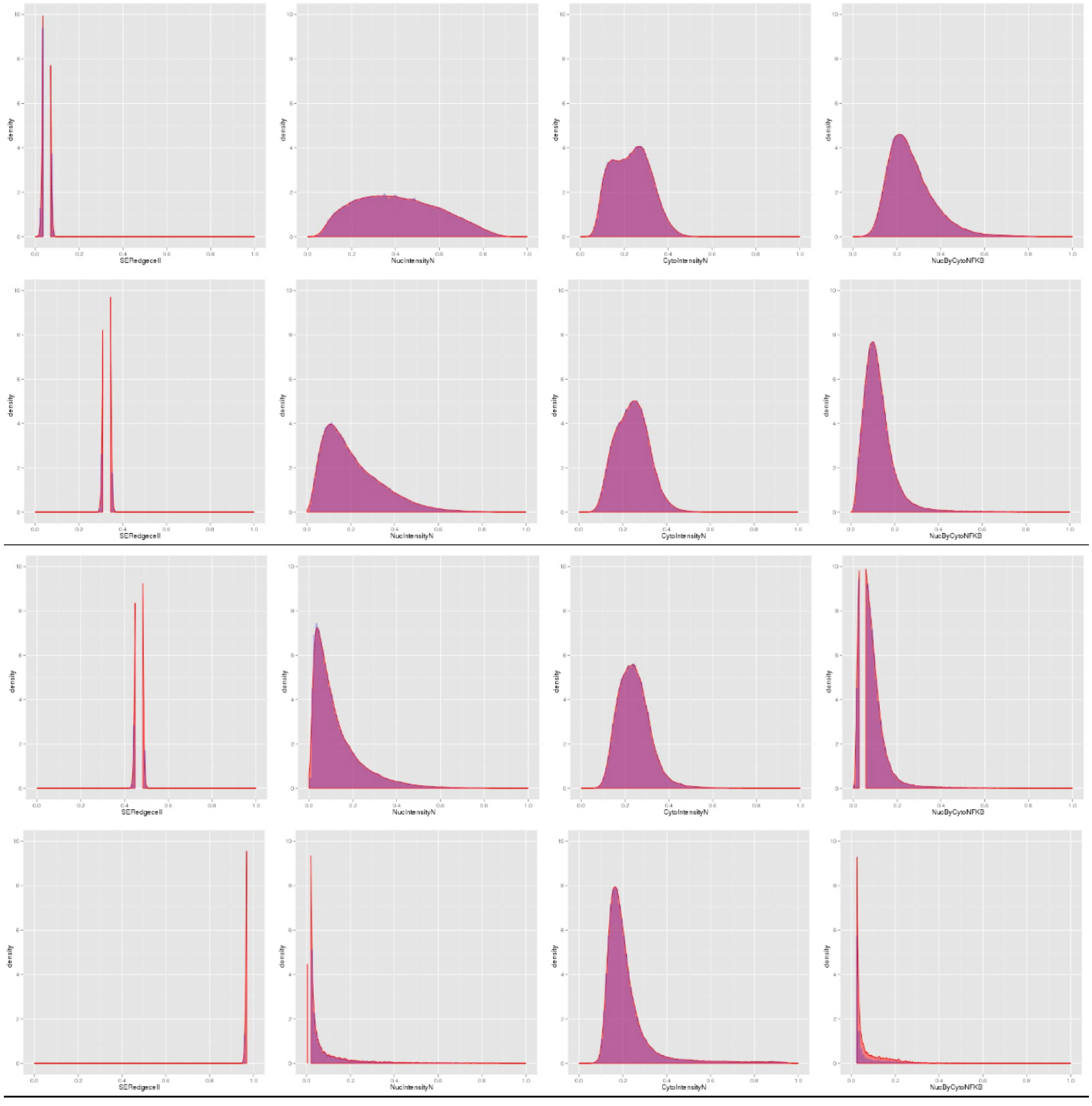
Prediction of cell line given features

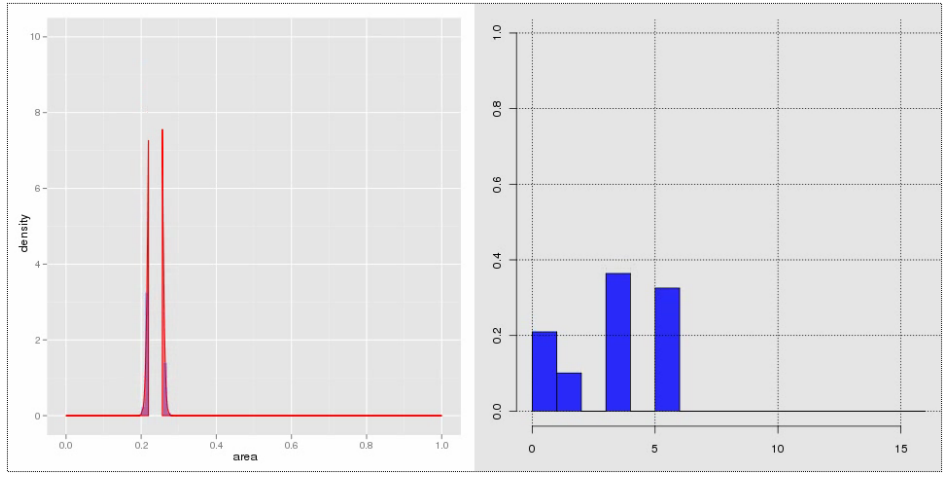
* Usually classification (e.g. cell line or treatment condition) or regression (e.g. expression of NFκB) can be done by training, say, a Feed Forward Neural Network (FFNN) on the set of features.
* FFNN Entities (FFNNE) are composed of many neural networks each seeing only a subset of the input features.
* Furthermore, we can employ many classifiers, constructed using different parameters, some of them with PCAed input. Their combined answer will be obtained by voting.
* Training was done on 60% of the data and testing on the rest yielding a $5-15\%$ error rate.

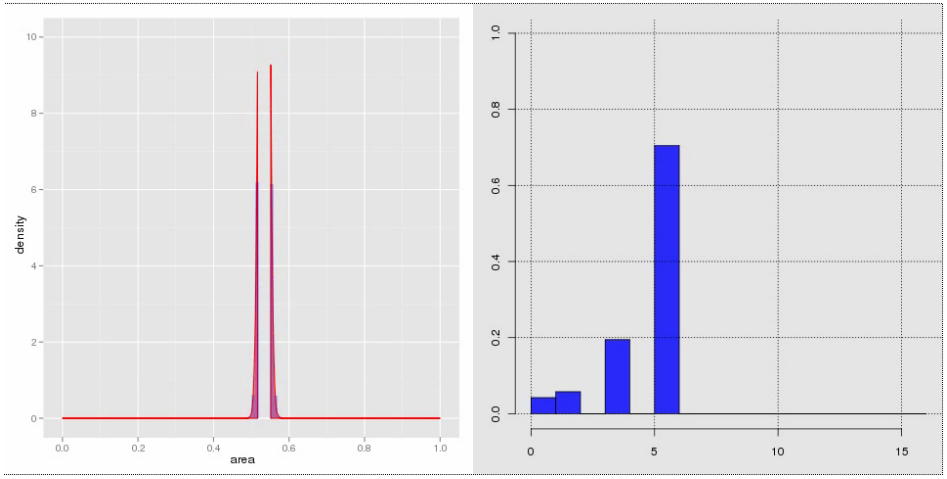
Modelling cell features

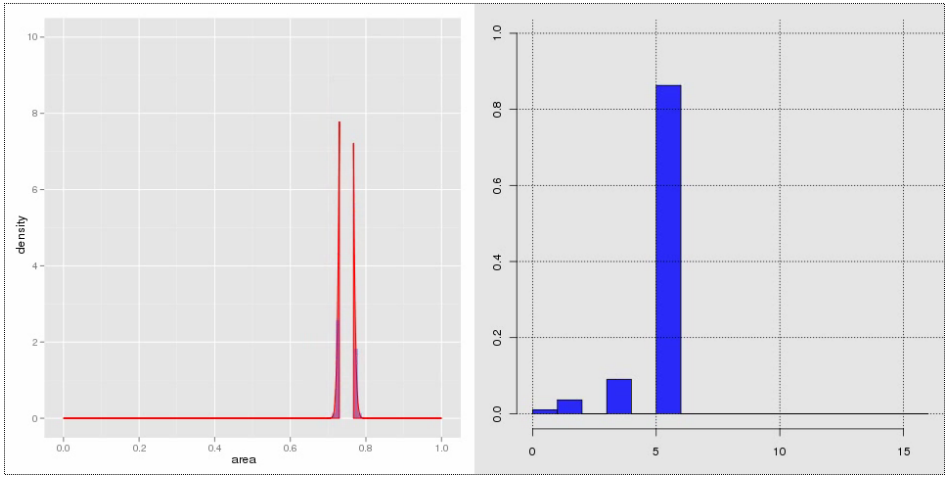
* When the trained classifiers predict which line a cell with specific morphology belongs to, in essense are questioning an internal model of the learned data.
* We can explore this model by systematically varying some of its inputs while keeping the rest fixed.
* Questions like "*what happens to the expression of NFκB when cell area increases?*" can be answered.
* An initial method – as proof-of-concept – was to vary only one feature value over its dynamic range while giving all the other features values generated by their respective statistical distribution.
* Notice that there is a lot of stochasticity in generating the input values and will be stochasticity in the output.
* The answer to our question will not be a unique value but rather a distribution.


density
SERedgecyto
NucIntensityN
CytoIntensityN
NucByCytoNFKB


density
SERedgeceII
NucIntensityN
CytoIntensityN
NucByCytoNFKB


density
area
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8
10
0
5
10
15


density
area
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8
10
0
5
10
15


density
area
0.0
0.2
0.4
0.6
0.8
1.0
0
2
4
6
8
10
0
5
10
15

Modelling cell features

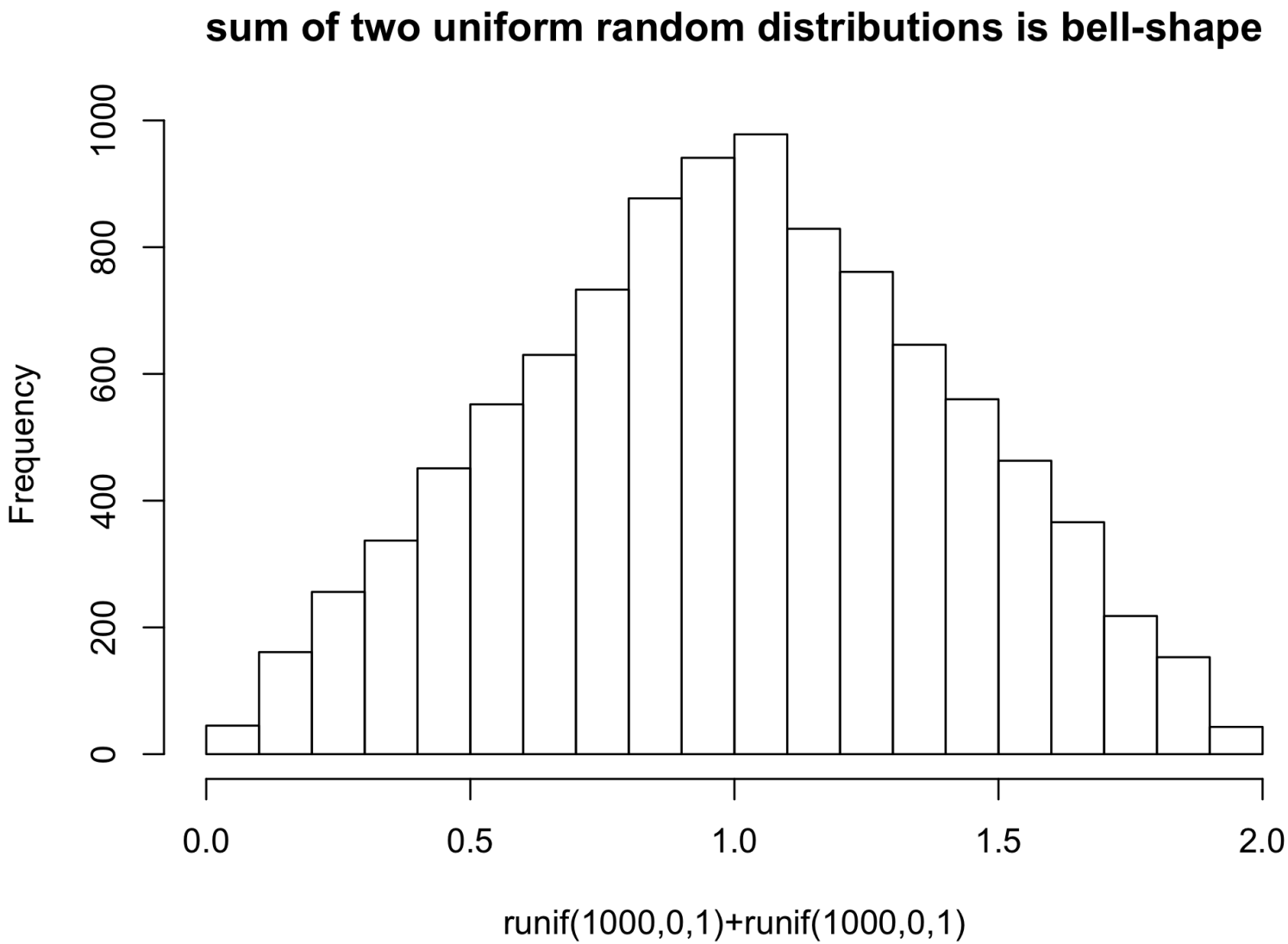
Animations of the changing distributions as the input changes at `http://nfkb.scienceontheweb.net/predict_nfkb_or_cell_line/index.html`

Central Limit Theorem

* One of the least remembered theorems in mathematics states that the distribution of the sums of independent random variables is Normal irrespective of the original distributions.
* In fact they can be uniformly random! But still the result of the interaction will be seen centred around some mean value.
* Other interactions (e.g. a product, a linear or polynomial combination) generally produce skewed bell-shaped distributions.
* The following R-script plots the distribution of the sum of two uniformly random distributions ranging between 0 and 1:

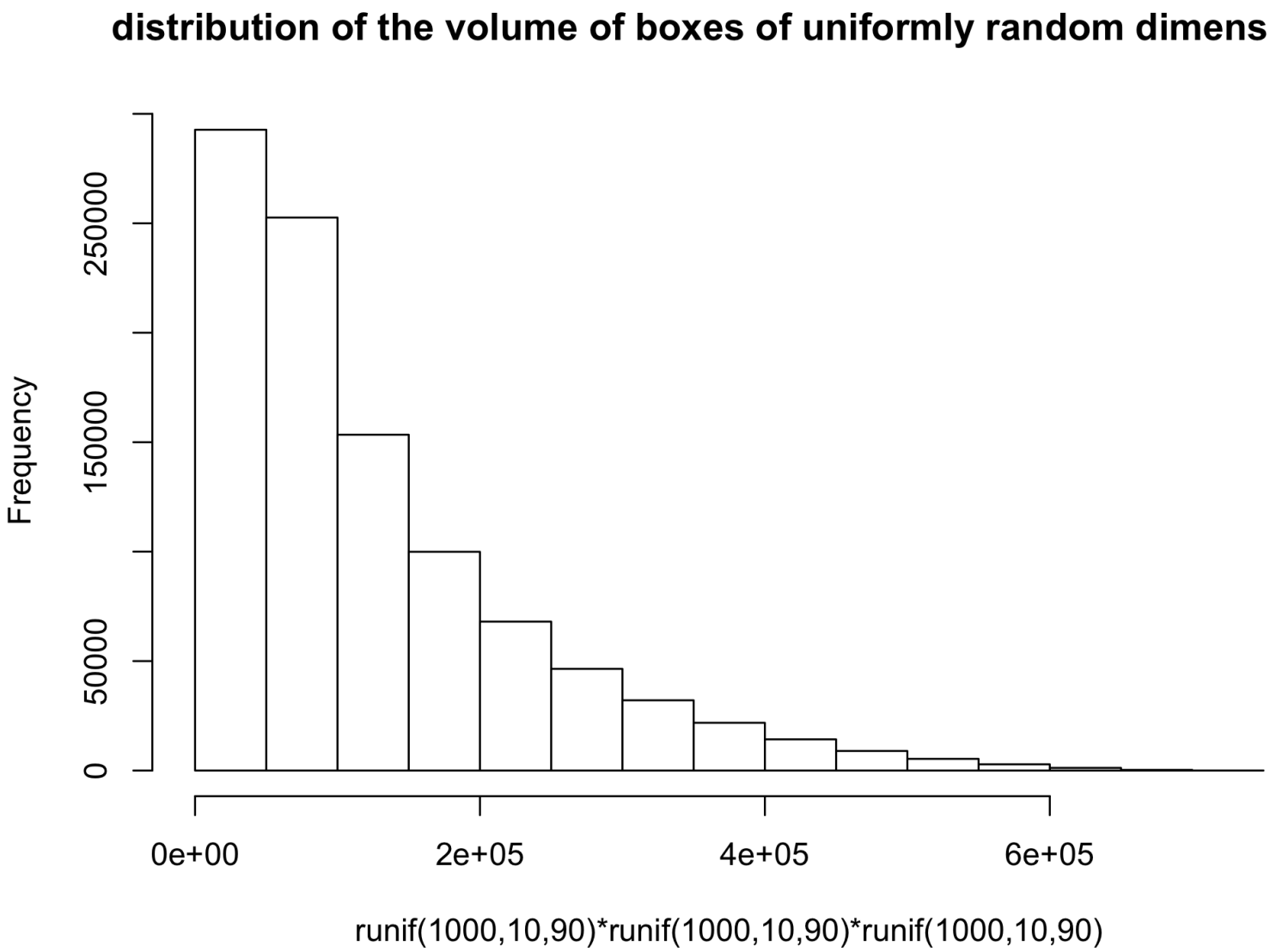
```
r1 <- runif(10000, 0, 1)
r2 <- runif(10000, 0, 1)
plot(hist(r1+r2))
```

Central Limit Theorem



Probability distribution of the sum of two uniformly random distributions (0-1)

Answer to the quiz



Probability distribution of the volume of the random paper boxes

code repository and collaborators

A large portion of the software I created for the analysis discussed in this presentation is publicly available from my code repository:

`https://github.com/hadjiprocopis`

Many thanks to:
• Janine Erler • Rune Linding • Joan Chang • Marie-Claude Djidja • Chris Marshall • Faraz Mardakheh • Wei Xing • Chris Bakal • Julia Sero • Richard Marais •

Thank you for your attention.