Small Nucleolar RNAs in breast cancer

Project report

Lorenzo Bocchi, Valentino Frasnelli, Erich Robbi, Annalisa Xamin

Laboratory of Biological Data Mining
Department of Information Engineering and Computer Science
University of Trento, Italy
{lorenzo.bocchi, valentino.frasnelli, erich.robbi, annalisa.xamin} @studenti.unitn.it

***Abstract*—The canonical roles of small nucleolar RNAs (snoRNAs), which include ribosome biogenesis and RNA modification, are well recognized. snoRNAs function as endogenous sponges that control the expression of miRNAs. Therefore, accurate snoRNA expression is essential for adjusting miRNA expression. Similarly to miRNAs, snoRNAs that have been converted into miRNA-like sequences are essential to control the expression of protein-coding genes. SnoRNA dysregulation is related to breast cancer (BC), according to recent findings. By allowing breast cells to develop cancer hallmarks, the inappropriate expression of snoRNA leads to the pathogenesis of breast cancer. This study tries to validate any causal association between snoRNA and breast cancer using quantitative methodologies such as machine learning-based approaches, statistical testing, and descriptive statistics. Furthermore, a study of the network of gene expansions conducted on FANTOM5, a dataset that is used as a reference for physiological conditions for the human body, will be conducted.**

***Index Terms*—snoRNA, small nucleolar RNA, human tissues, RNA-Seq, snoRNA/host gene relationship, transcriptome, ribosome, regulation.**

I. Introduction

SMALL nucleolar RNAs (snoRNAs) are a conserved type of noncoding RNA, best characterized for their role in ribosome biogenesis[1]. The three main categories of classic snoRNAs are as follows: first, the C/D box snoRNAs (**SNORDs**), which are typically 60 to 90 nucleotides long and contain C/D box motifs (C: RUGAUGA and D: CUGA). Next, the H/ACA box snoRNAs (**SNORAs**) have H/ACA box motifs (H: ANANNA and ACA: ACA) and range in length from 120 to 140 nt. Third, the size of small Cajal body-specific RNAs (**SCARNAs**) varies substantially, and they contain different combinations of the C/D and H/ACA motifs. SCARNAs are localized to Cajal bodies, where it is hypothesized that they contribute to the modification of U1 to U6, while the SNORDs and SNORAs normally reside in the nucleolus where they are incorporated into RNPs that change rRNA. [2]

Because of recombination and retrotransposition from an ancestral snoRNA, many snoRNAs exist in mammalian genomes in multiple copies. They are encoded either in the intronic region of the larger host gene or independently outside the gene and are highly distributed in the nucleolus of eukaryotic cells. In particular, Box C/D snoRNA family members are frequently encoded in the same host gene, whereas box H/ACA family members typically occur in many hosts. Moreover, the regulation of snoRNAs encoded in the same host gene can differ depending on the tissue, as does the quantity of some snoRNAs within the same family [3]. These non-coding RNAs play a key roles in the RNA modification process: they function as guide RNAs for the site-specific modification of target RNAs such as rRNAs and snRNAs [4] (Figure 1 reports a detailed description of all their functions).

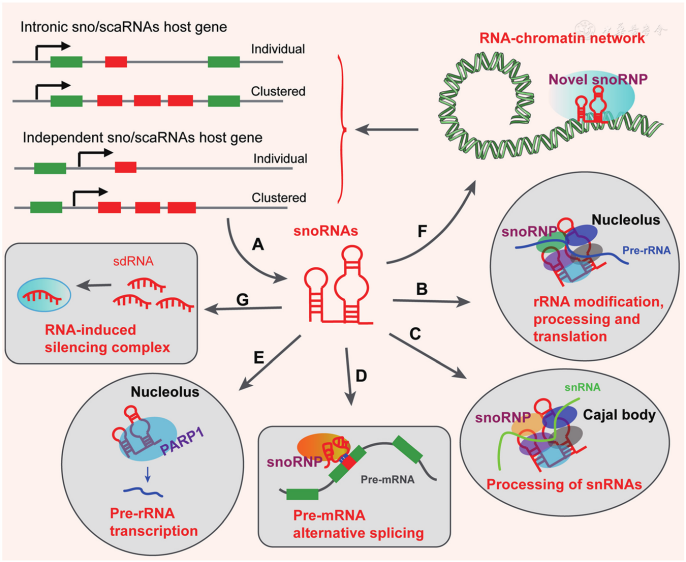

Figure 1. **Function of snoRNAs**. **A**. snoRNAs are derived from the snoRNA genes with independent promoters, or intronic snoRNA genes encode them. **B**. The canonical function of snoRNAs is realized by guiding 2'-O-methylation and pseudouridylation of rRNAs in the nucleolus. **C**. scaRNAs also guide 2'-O-ribosemethylated nucleotides and pseudouridines on snRNAs in cajal bodies. **D**. The "orphan" snoRNAs play an important role in alternative splicing. **E**. snoRNAs bind to PARP1 and stimulate its catalytic activity to promote rDNA transcription. **F**. snoRNAs are also enriched in chromatin, which suggests a chromatin-associated role. **G**. snoRNAs can be processed into sno-derived RNA (sdRNA), which has been shown to perform a regulatory function similar to microRNAs. snoRNA= small nuclear RNA.
Reprinted from *Small nucleolar RNA is potential as a novel player in leukemogenesis and clinical application*, by L. M. Lin, Q. Pan, Y. M. Sun, and W. T. Wang, Blood Science, vol. 3, no. 4, pp. 122–131, Oct. 2021 [5].

Previous research has shown that snoRNAs, which are primarily thought to be housekeeping genes, influence rRNA synthesis. However, this assumption has recently been questioned. Through signaling pathways and cell cycles, snoRNAs may have an impact on the control of tumor cell growth and death. Gene amplification and gene inhibition may dramatically alter a person's genetic makeup

by encouraging cancer development. Therefore, understanding the mechanisms underlying genes whose expression is altered or enhanced will help us to understand how cancers arise and progress. In particular, snoRNA's expression appears to be dysregulated in a peculiar way, indicating that the incidence of cancer is intimately associated with these dysregulations. [1]

For this reason, the issues that this project tries to address are whether there is a difference in the level of expression of snoRNAs in our genes of interest in the presence of breast cancer compared to healthy individuals.
Recent research highlighted the role of some host genes in the presence of cancer, where changes in snoRNA expression were analyzed and genes such as *GAS5* and *ZFAS1* were found to have important functions in cancer development, acting, respectively, as regulators of cell death and proliferation and tumour suppressive ncRNA[6].

The key point of this project is to compile a list of potential genes that could be relevant for breast cancer and to analyze their expression levels in cancer patients compared to the control group, which in our case is made up of healthy patients. In addition, our objective is to find if there are causal relationships between snoRNAs and genes linked to breast cancer and if there are differences in the expression levels of those genes between the control and treatment groups.

II. Materials and methods

The pipeline of the project, as illustrated in Figure 2, consists of applying the RMA normalization to a Gene Expression Omnibus data set. After that, a subset of the gene of interest is gained and on that, a feature selection is applied. Then, we continued with the differential expression analysis and model tuning and fitting. We also performed network analysis using gene network expansions from the FANTOM5 dataset.

A. Data preprocessing

The very first step is to collect data. For what concerns, the gene expansion and network analysis data from the FANTOM5 project[7] have been used. The Fantom dataset has been created with the CAGE technology, which identifies all the transcriptional starting sites (TSS) for a gene.
The snoDB 2.0 database[8] has also been used to match information. After collecting the datasets, the relevant genes are extracted from FANTOM5 and joined with those from snoDB.

The resulting dataset is then filtered once again to only keep snoRNAs and host genes that, according to research, play a role in the development of breast cancer. In particular, the snoRNAs kept are:

SNORA1	SNORA12	SNORA14B
SNORA16A	SNORA21	SNORA23
SNORA24	SNORA32	SNORA38
SNORA44	SNORA48	SNORA49
SNORA52	SNORA53	SNORA57
SNORA61	SNORA63	SNORA64
SNORA65	SNORA70	SNORA71C
SNORA73A	SNORA73B	SNORA75
SNORA78	SNORA8	SNORD10
SNORD102	SNORD104	SNORD105
SNORD105B	SNORD108	SNORD110
SNORD111B	SNORD113-3	SNORD113-4
SNORD114-1	SNORD114-13	SNORD114-14
SNORD114-19	SNORD114-20	SNORD114-21
SNORD115-23	SNORD115-32	SNORD116-13
SNORD119	SNORD12	SNORD12B
SNORD13	SNORD14A	SNORD14C
SNORD14D	SNORD15B	SNORD16
SNORD17	SNORD18A	SNORD1B
SNORD20	SNORD22	SNORD26
SNORD28	SNORD29	SNORD32A
SNORD33	SNORD34	SNORD35A
SNORD36A	SNORD36B	SNORD38A
SNORD3A	SNORD3D	SNORD41
SNORD42A	SNORD42B	SNORD44
SNORD45A	SNORD47	SNORD49B
SNORD4B	SNORD5	SNORD50A
SNORD52	SNORD53	SNORD54
SNORD55	SNORD56	SNORD57
SNORD58A	SNORD58C	SNORD59A
SNORD60	SNORD63	SNORD64
SNORD65	SNORD68	SNORD69
SNORD71	SNORD74	SNORD76
SNORD8	SNORD81	SNORD82
SNORD84	SNORD86	SNORD87
SNORD89	SNORD94	SNORD96A
SNORD97	SNORD99	

Table I
LIST OF SNORNAS INVOLVED IN THE DEVELOPMENT OF BREAST CANCER (ACCORDING TO LITERATURE).

And their host genes:

TMX1	ZFAS1	CDKN2B-AS1
CWF19L1	EIF4A1	EP400
HIF1A-AS2	MEG8	NOP56
RACK1	RPL13A	RPS13
SCARNA12	SNHG12	SNHG20
SNHG5	SNHG6	SNHG7
SNHG8	AP1G1	CFDP1
CHD8	DDX39B	EEF2
EIF4A2	EIF4G2	GAS5
GNL3	HSPA8	HSPA9
IPO7	MYRIP	NAN
NCL	NFATC3	PCAT4
PPAN	PRKAA1	PRRC2A
PTCD3	RABGGTB	RNF149
RPL10	RPL12	RPL13
RPL17	RPL21	RPL23
RPL23A	RPL4	RPL7A
RPLP2	RPS11	RPS2
RPS20	RPS3	RPS8
SF3B3	SLC25A3	SNHG1
SNHG3	SNHG9	SNORD1C
SNORD35B	SNORD37	SNRPB
SNX5	TAF1D	TNPO2
TOMM20	WDR43	

Table II
LIST OF HOST GENES

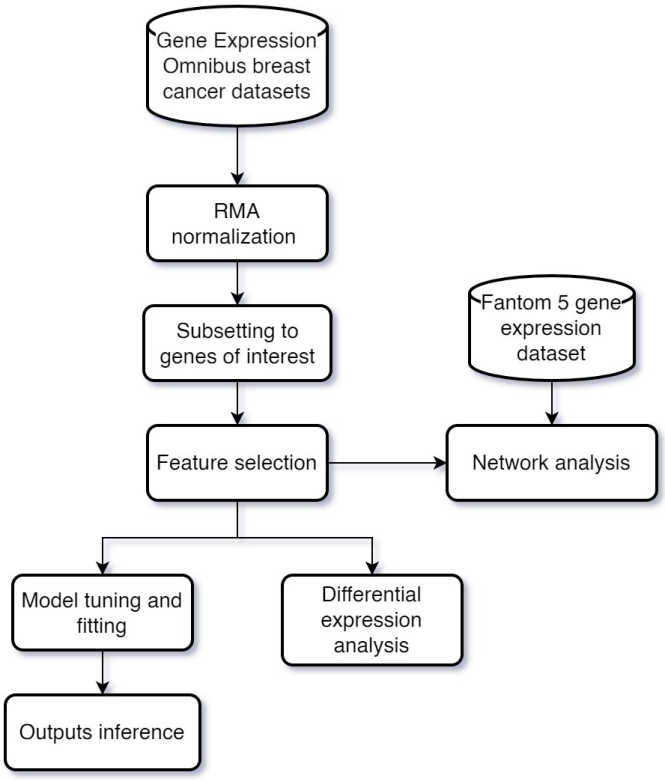

Figure 2. Project pipeline

For the rest of the analysis, two datasets from Gene Expression Omnibus (GEO) have been used. In particular, firstly, we tried our pipeline by merging the following datasets from GEO: GSE10810[9], GSE29431[10], GSE42568[11] and GSE61304[12]. After a first try, we decided to continue our work with a different dataset, but still about breast cancer patients: GSE45827[13].
The first GEO dataset used contains both control and test samples: in particular, it contains expression data of the genes both in presence of cancer and in healthy patients. It was obtained from a combination of different data sets, all related to breast cancer: GSE10810 (58 samples), GSE29431 (66 samples), GSE42568 (121 samples), and GSE61304 (62 samples). They all contain the same genes and their combination amounts to a total of 307 samples, 60 of which are control samples with no breast cancer. The data was created using an Affymetrix Human Genome U133 Plus 2.0 Array. Later, prompted by some issues with the data which will be discussed in the results section, we decided to use the data coming from the Gene Expression Omnibus with id GSE45827[1]. This dataset contains 178 samples of "Expression data from Breast cancer subtypes" coming from 155 different patients. This data was also created with the same Affymetrix Human Genome U133 Plus 2.0 Array platform.

To prepare the data for analysis, we follow the following steps:

[1]https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE45827

1) We loaded the data with the *affy* library[14] in R.
2) We proceeded with a normalization step to remove batch effects from the Affymetrix arrays data. On the raw micro-array data we used the RMA (Robust Multiarray Analysis)[15] normalization. In this way, probes that mapped to the same gene have been collapsed to one computing average for the gene expression value.
3) The logarithmic value of all gene expression levels was calculated and used downstream in the analysis.
4) Transpose the matrix and add columns useful for ML prediction (such as the binary predictor variable for the absence or presence of cancer in the first dataset or the multiclass predictor variable in the second dataset for the cancer subtypes).

Why did we need to perform a normalization?

Affymetrix GeneChips are made up of a number of probes, each designed to measure the expression levels of a particular genomic sequence. Each probe consists of hundreds of short 25-mer oligonucleotide strands that match the target mRNA sequence exactly. RNA samples are transformed into cRNA (complimentary RNA) using an in vitro transcription method that. cRNA is fragmented, biotin labels are attached, and the fragments are washed over the chip. cRNA fragments hybridise with their respective oliginucleotide sequence, resulting in an increase in abundance of biotinylated cRNA molecules on the probe. A fluorescent dye is washed over the chip and binds to the biotin labels. A fluorescent scan of the chip gives fluorescence values for each probe, which are directly related to the abundance of cRNA fragments hybridised to the probe. These fluorescence levels can be used to infer the relative abundance of specific mRNA sequences in a sample, giving a "snapshot" of the transcriptome.

Affymetrix Genechips are designed in such a way that each gene is matched to 11-20 such probes evenly distributed throughout the chip. These probes make up a Probe Set on the chip. The presence of noise is a large problem in large scale microarray studies, and can come from a number of sources. Noise due to non-specific binding of cRNA fragments to probes is one such source, and is eleviated through the use of probes on the chip designed to measure for non-specific binding. Each probe described above, termed a Perfect Match probe (PM) is matched to a second probe, termed a Mis-Match probe (MM). MM probes are identical to PM probes but for the middle (13th) nucleotide in the sequence, which is replaced with it's complement. By subtracting the signal for the MM probe from the signal for the PM probe, a value for the true signal is reached.

However, it has been shown that the signal strength for the MM probes can often be larger than that of the PM probes, implying that the MM probe detects both the true signal and background signal. This can result in non-sensical negative expression values. RMA is a normalisation procedure for microarrays that background corrects, normalises and summarises the probe level information without the use of the information obtained in the MM probes [15], [16], [17].

B. Exploratory Data Analysis

EDA is a way of analyzing data sets to summarize their essential properties, frequently utilizing statistical graphics and other data visualization approaches. There are several techniques available for EDA; in our work, we used Principal Component Analysis.

PCA consists in projecting orthogonally a dataset $X = x_1, \ldots, x_n$ of n p-dimensional points into a r-dimensional space with $r = min(n-1, p)$, so that in the new coordinates the projected points' variables are uncorrelated. The new coordinates are called principal components, and each component is defined by the rules:

- being orthogonal to the previous components
- having highest possible variance

The first principal component corresponds to a line that passes through the multidimensional mean and minimizes the sum of squares of the distances of the points from the line. Each eigenvalue is proportional to the portion of the "variance" (more correctly of the sum of the squared distances of the points from their multidimensional mean) that is associated with each eigenvector. The sum of all the eigenvalues is equal to the sum of the squared distances of the points from their multidimensional mean The explained variance of k_{th} principal component is is $\lambda_k / \sum_i \lambda_i$. PCA essentially rotates the set of points around their mean in order to align them with the principal components. This moves as much of the variance as possible (using an orthogonal transformation) into the first few dimensions.

C. Differential gene expression analysis

After having removed the batch effect, we proceeded to find differentially expressed genes or (DEGs). It's important to note that this procedure was done only on the second dataset.

Differential Gene Expression analysis refers to the analysis and interpretation of differences in abundance of gene transcripts within a transcriptome [18]. Practically, it consists of normalizing the read count data and performing a hypothesis test for the means, to observe quantifiable differences in expression levels between the groups.

DGE analysis in this case is performed in order to assess the difference between control and treatment. More specifically, the difference between the control and the different cancer subtypes. To perform this analysis, we used the *limma* package of R [19].

D. Feature selection

The best way to proceed would be by carrying out a feature selection on our data, firstly, by removing all the genes that are not relevant to the project, afterwards, we further reduce the number of features via an algorithmic selection.

To do so, we fit a regularized algorithm, called LASSO, which applies a penalty to the coefficients in order to introduce bias in the model. This penalty makes it so that the coefficients can decrease to zero, performing feature selection in an embedded way. By tuning the penalty through cross-validation, the variables can be subset to an acceptable number, allowing us to fit the models without worrying about the $p >> n$ problem. Furthermore, the coefficients with the optimal penalty can provide us with further insight into how correlation in gene expression is related to the presence of breast cancer.

E. Classification models

Fitting a classification model with our gene expression data will allow us to infer useful information and assess whether it is possible to perform a classification that predicts our cancer subtype groups with the transcripts of the genes we have chosen.

The models we are fitting and testing on our data are:

- Logistic regression
- Lasso
- Random forest
- Support vector machine (more specifically kernel machines)

The choice of such models was justified by some of their embedded properties. Logistic regression, like LASSO, outputs coefficients and significance (p-value) for each variable, which once again can give us insight into how gene expression correlates with the presence or absence of breast cancer. On the other hand, random forest computes variable importance, which can be useful to extract more insight from our data. Support vector machines were chosen for being largely considered the best performing ML classifier, and in this project, they are used to understand just how good the classification performance can get.

Now, a brief theoretical introduction to the various classification methods.

1) Logistic regression: Logistic regression models the probability of an event taking place by having the log-odds of such event be the linear combination of the covariates:

$$ln(\frac{p}{1-p}) = \beta_0 + \beta_1 \times X_1 + \ldots + \beta_k \times K_k$$

The logistic function then converts such log-odds to a value between 0 and 1 through the logistic function, so that one has the probability of a sample to belong to a certain class. The coefficients of the logistic regression are calculated by maxmimum-likelihood estimation, finding the best β_k values by optimizing the fit on the log-odds. These coefficients are important for analysis, since they tell how a variable varies when the dependent variable (response) varies.

2) Lasso: Lasso (Least absolute shrinkage and selection operator) is a regression method that performs variable selection through regularization to improve the prediction accuracy and interpretability of the resulting statistical model. The aforementioned regularization occurs in the form of shrinkage, implemented through a penalty applied to the coefficients, which, in the particular case of Lasso, shrinks coefficients in order to introduce bias and decrease the variance in the best fit model, restricting the influence of the predictors. Not only does such penalty shrink coefficients towards zero, but it also makes them zero, making the model sparse (some of the variables do not influence the outcome) and therefore simpler. This can be seen as a form of variable selection, ideal for the domain of the project, where most likely we will have to deal

with large amounts of variables and a selection of them will be necessary. Of course, coefficients, which have not shrunk to zero are useful for interpretability of the model, giving more insight into the roles of the variables.
Now, Lasso penalty is usually used in linear regression models, which are not suited for our purpose, since the project will most likely deal with classification problems, not regression ones. Fortunately, the Lasso regularization technique is easily extended to so-called Generalized Linear Models, a more general and flexible formulation of linear regression, which allows to use shrinkage in a classification context, and more specifically in a logistic regression model.

3) Random forest: Random forest is an ensemble Machine Learning algorithm. For classification, it works by creating N classification trees and taking the most recurring response class predicted by the trees. It uses the bootstrap method, meaning that for each tree it creates a training set composed of randomly selected samples to avoid always fitting the same data on all the trees. Also, when building these decision trees, each time a split in a tree is considered, a random sample of m predictors is chosen as split candidates from the full set of p predictors. The split is allowed to use only one of those m predictors. A fresh sample of m predictors is taken at each split, and typically we choose $m \approx \sqrt{p}$. The reason for this is that in the dataset there might be a very strong predictor. This means that most if not all of the trees in the ensemble would use this predictor, making all the trees very similar and possibly highly correlated. Averaging many correlated trees have a smaller reduction in variance than averaging uncorrelated trees; therefore, we force each split to only consider a random subset of the possible predictors.

4) Support vector machines: Support Vector Machines, for short SVMs, are another supervised ML algorithm that can be used for classification. More specifically, they are divided in three subclasses:

- Hard Margin Classifiers
- Soft Margin SVMs
- Non-linear SVMs

The first subclass, Hard Margin Classifiers (also called Maximal Margin Classifiers), represents the simplest form of SVM. The idea is that, given a set of n labelled and linearly separable samples, there is a hyperplane that can separate the examples. But this does not give a single solution since, therefore we choose the one that maximizes the margin between the two classes. The margin is 2 * the smallest distance of any point to the hyperplane. If such a solution exists, the algorithm is guaranteed to find the globally optimal solution. The samples that "sit" on the margin are called Support Vectors, and they are the only ones that matter to the final boundary. Finding the maximal margin requires solving a constrained optimization problem via quadratic programming. The major issue with this subclass of SVMs is that it only works with linearly separable problems with no chance of misclassification. This means that even if the data are generally linearly separable, but there is a single misclassified example, there will not be a solution. To overcome this, we move to the second subclass, Soft Margin SVMs.

The key difference between Soft Margin SVMs and Hard Margin SVMs is that here we introduce the concept of slack variables ε. A slack variable ε_i measures the penalty for example x_i for not satisfying the margin constraint. In particular, based on its value we can assert where the sample is located with respect to the hyperplane:

- $\varepsilon_i = 0 \rightarrow$ the sample is on the correct side of the hyperplane and does not violate the margin
- $0 < \varepsilon_i < 1 \rightarrow$ the sample is on the correct side of the hyperplane but violated the margin
- $\varepsilon_i = 1 \rightarrow$ the sample is on the hyperplane and cannot be classified
- $\varepsilon_i > 1 \rightarrow$ the sample is on the wrong side of the hyperplane and it's misclassified

The sum of all slack variables is minimized in the objective of the SVM together to the inverse margin and it's multiplied to a parameter C. This parameter allows a trade off between data fitting and misclassifications. The higher its value, the more misclassifications are allowed. If it's 0, we simply have a Hard Margin Classifier.

What if we have a non-linearly separable problem? In that case we can resort to the third subclass of SVMs, Non Linear SVMs. The idea is that we can define a higher dimensional space with respect to the one the data is in with p* > p dimensions (p is the number of dimensions in the original space) and then perform linear classification in this higher dimensional space. We can do so by mapping the features in the higher dimensional feature space. An example of mapping is the polynomial mapping which maps features to all possible conjunctions of features of a certain degree d (in case of the homogeneous mapping). One issue with this approach is that computation can be highly expensive in very high dimensions. We can use a trick called "kernel trick" which uses a kernel function that acts as a dot product in some p* dimensional space for some mapping. The most common kernel functions are:

- linear kernel $= K(x, z) = x \cdot z = \sum_{j=1}^{p} x_j z_j$ non linear SVMs in this case coincide with Soft Margin SVMs
- polynomial kernel $= K(x, z) = (1 + \sum_{j=1}^{p} x_j z_j)^m$ adds a constant to the linear kernel and gets elevated to a certain degree (with m = 2 one obtains a second degree kernel for example)
- radial kernel $= K(x, z) = exp(-\gamma \sum_{j=1}^{p} (x_j - z_j)^2)$ with $\gamma > 0$ here the kernel tends to 0 the further x gets from z

Once again, we need to cross validate to understand which kernel to use and which parameters are best.

To be sure that the models actually yield relevant results, not only cross validation is applied but also a fit on 'housekeeping genes' and low variable genes is performed and compared with the fit on our genes of interest. The selected housekeeping genes are described in Table III.

Housekeeping genes are genes that are involved in basic cell maintenance and, therefore, are expected to maintain constant expression levels in all cells and conditions. Their identification facilitates exposure to the underlying cellular infrastructure and increases understanding of various structural genomic

Gene symbol	Gene description
C1orf43	Chromosome 1 open reading frame 43
CHMP2A	Charged multivesicular body protein 2A
EMC7	ER membrane protein complex subunit 7
GPI	Glucose-6-phosphate isomerase
PSMB2	Proteasome subunit, beta type, 2
PSMB4	Proteasome subunit, beta type, 4
RAB7A	Member RAS oncogene family
REEP5	Receptor accessory protein 5
SNRPD3	Small nuclear ribonucleoprotein D3
VCP	Valosin containing protein
VPS29	Vacuolar protein sorting 29 homolog
ACTG1	Cytoskeletal Gamma-Actin
RPS18	Ribosomal Protein S18
POM121C	POM121 Transmembrane Nucleoporin C
MRPL18	Mitochondrial ribosomal protein L18
TOMM5	Translocase of outer mitochondrial membrane 5
YTHDF1	YTH N6-methyladenosine RNA binding protein 1
TPT1	Tumor protein, translationally-controlled 1
RPS27	Ribosomal protein S27

Table III
LIST OF HOUSEKEEPING GENES

LINC00189	SCEL	CDH7
FAM9B	NAALAD2	GRTP1-AS1
DCAF4L2	LINC02059	LINC00408
UST-AS1	PRKCA-AS1	PHLDB2
LINC00301	LINC01020	TYR
FMR1NB	C4orf33	DNAH6
C11orf87	GATM	SLC9C2
PCDHGB7	HISLA	CNOT6L
CFAP97	LHFPL3-AS1	ARPP21
MPP4	CDC14C	LINC01990
UBDP1	BCL2L14	LINC01692
PPP4R3C	LINC02297	HNF4A-AS1
CAGE1	LARGE-AS1	PKIA-AS1
MYLK3	LIFR-AS1	SAMMSON
LINC02395	BDNF-AS	LINC00102
SLC17A1	IGF2BP2-AS1	TRAV36DV7
BRPF3-AS1	LMOD3	LINC02424
GRK4	MIPOL1	F11
TRAF3IP2-AS1	LINC01088	EAF2
ZNF605	GSK3B-DT	WARS2-AS1
ALDOB	DAZ4	DAZ3
DAZ1	DAZ2	GYPA
PDE6C	DHFRP3	SLC13A1
MCFD2P1	KRTAP9-9	WNT16
ZFY	DEFB126	CDH26
CABS1	LINC02555	LINC02619
LINC00644	ASB5	MMAA
TECRL	VWC2	ADAM30

Table IV
LIST OF LOW VARIABLE GENES

features. In addition, they are instrumental for calibration in many biotechnological applications and genomic studies. So, we have chosen the gene listed above as control as suggested from the literature[20], [21] because they are expected to be expressed in all cells of an organism under normal conditions, irrespective of tissue type, developmental stage, cell cycle state, or external signal. Subsequentally, Table IV illustrates genes with few alterations in their expression levels in the dataset.

F. Network analysis

Finally, we perform a network analysis on the FANTOM5 dataset using only the genes that we are interested in. This allows us to understand all possible interactions among genes related to a particular network. Therefore, we are able to find causal relationships among our genes of interest using OneGene, a method that uses PC-algorithm. We run this method on the BOINC platform in order to expand our list of genes, thanks to the computational power provided by gene@home volunteers. To represent the interactions that resulted from the expansions, we decided to build a network representing the connections among the genes. To do this, a fork of the library Tools_Expand_Genes_Network[2] was created and made available on Github[3]. The reason for this was that the original library does not plot the nodes with the name of the gene, but with a number. This is not useful for the purpose of this project and, therefore, the library was edited to plot the gene names instead.

[2]https://github.com/gabrieletome/Tools_Expand_Genes_Network
[3]https://github.com/bocchilorenzo/Tools_Expand_Genes_Network

The input to the library is a zip file containing all the various gene isoforms downloaded from gene@home. These isoforms are themselves zip files that contain a .expansion and a .interaction file. The library then proceeds to build a complete graph of interaction and find an LGN of the genes. The local gene network is discovered using an adapted version of the Charikar algorithm [22].

Before explaining the results obtained with each method, we briefly introduce a few key concepts.

1) PC-algorithm: The PC-algorithm is a method that allows estimating the equivalence class for a high-dimensional directed acyclic graph (DAG). It starts from a complete, undirected graph and recursively deletes edges based on conditional independence decisions. This yields an undirected graph that can be partially directed and further extended to represent the underlying DAG. This algorithm runs in the worst case in exponential time, but if the true underlying DAG is sparse, it is reduced to a polynomial runtime. This can be achieved by reducing the search space from the individual DAGs to equivalence classes. Therefore, the focus is on estimating the equivalence class and the skeleton of DAGs corresponding to multivariate Gaussian distributions in a high-dimensional context, or where the number of nodes p may be much larger than the sample size n. Generally, the PC-algorithm is divided in two steps:

- creation of the skeleton of the DAG;
- extension of the skeleton to the equivalence class.

There are two main versions of the skeleton creation:

- population version = perfect knowledge about all necessary conditional independence relations is assumed available

- sample version = the nodes correspond to multivariate normal distributed random variables, and thus it uses partial correlations to approximate conditional independencies

Usually, the exact conditional independence relations are not known, therefore it is better to consider the sample version. Partial correlations are computed via regression or other methods; for each result, it is possible to define a two-sided statistical test, where the null hypothesis is "partial correlation of nodes i and j given a set of nodes k is 0", with a significance level α (the only tuning parameter). In case of the gene@home project, linear correlations are computed using the Pearson coefficient:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

The steps to assign direction to some of the edges of the graph are:

- Given any pair of non-adjacent variables i and j with common neighbour k, orient $i - k - j$ into $i \to k \leftarrow j$if k is not in $S(i, j)$ (which is the separation set computed during skeleton construction)
- Orient $j - k$ into $j \to k$ whenever there is an arrow $i \to j$ such that i and k are nonadjacent
- Orient $i - j$ into $i \to j$ whenever there is a chain $i \to k \to j$
- Orient $i - j$ into $i \to j$ whenever there are two chains $i - k - j$ and $i - l - j$ such that k and l are nonadjacent
- Orient $i - j$ into $i \to j$ whenever there are two chains $i - k - l$ and $k - l - j$ such that k and l are nonadjacent

There are some variants to this algorithm, namely:

- PC-Stable Algorithm (PC*): the deletion of an edge takes place after all sizes of the neighborhood have been tested, leading to more conditional independence tests overall, but the result is independent from the order of the operations. This gives the ability to parallelize the computation, using the GPU and speeding up the process
- PC-Iterative Method (PC-IM) = the algorithm is performed iteratively on tiles, subsets of variables each of which contains all the variables already known to belong to some network to expand, plus some other variables chosen at random

OneGene uses the second variant, the PC-Iterative Method.

2) OneGene: OneGene is an algorithm that is used to perform gene network expansion. It was developed as a successor to NES^{2}RA with the objective of reducing the delay between the user query and the response from the system. The inputs to the algorithm are:

- set of transcripts and an expression matrix with the expression levels of each transcript in various conditions;
- tuple of parameters, those being set of alpha values, set of subset size and set of number of iterations;
- local gene network (LGN) to expand.

The OneGene pipeline is divided in two steps:

- **pre-computation**: for each transcript in the set, multiple PC-IM iterations are performed. A single of those iterations corresponds to a NES^{2}RA run with the single transcripts as LGN and the probability usually set to always 1. The result is the candidate expansions list for the transcript with all the parameters. There is no aggregation here and the expansions are stored for later use;
- **real-time interaction**: when all transcripts belonging to the desired LGN have been expanded, OneGene aggregates the results using different ranking aggregators. The aggregated result is given back to the user.

3) Network Validation: The approach to validate the network was to extract from snoDB all the snoRNAs in our selection, extract their target genes when available and check in our network how many of the snoRna - target connections were satisfied. In total, 231 snoRna - target connections were gathered, with 60 snoRNAs that had no recorded targets in snoDB, namely:

SNORD14A	SNORD8	SNORD32A
SNORD14C	SNORD58C	SNORA71C
SNORD36B	SNORD55	SNORA8
SNORD4B	SNORD5	SNORD33
SNORD71	SNORA12	SNORD86
SNORD99	SNORA64	SNORD68
SNORD18A	SNORD45A	SNORD37
SNORD97	SNORA53	SNORA57
SNORD58A	SNORD35B	SNORA48
SNORD36A	SNORD111B	SNORA49
SNORD14D	SNORD26	SNORA52
SNORD42B	SNORA61	SNORD65
SNORA24	SNORA23	SNORD102
SNORD57	SNORD110	SNORD87
SNORD96A	SNORD69	SNORD54
SNORD17	SNORA38	SNORA1
SNORD3A	SNORD74	SNORD1C
SNORD44	SNORD29	SNORA63
SNORD49B	SNORD36C	SNORA16A
SNORD58B	SNORD34	SNORA32

Table V. List of snoRNAs with no target on snoDB

III. Results and Discussion

A. First part: initial dataset and suspicious results

Analyzing the data raises some issues. In particular, plotting X and Y the results are unexpected and seemingly wrong. As we can see in figures 8,9 and 10 the plots have strange patterns that should not appear. This sort of strange issue is confirmed by trying to fit the models.

Trying this technique with a Random Forest yields suspicious results. Fit on genes of interest:

		Truth	
		No Cancer	Cancer
Predicted	No Cancer	12	0
	Cancer	3	62

Table VI
Random forest fit on our genes of interest (old data)

	Accuracy	Precision	Recall	F1
No Cancer	0.9615	1	0.80	0.89
Cancer	0.9615	0.95	1	0.98

Table VII
RANDOM FOREST PERFORMANCE MEASURES ON OUR GENES OF INTEREST (OLD DATA)

		Truth	
		No Cancer	Cancer
Predicted	No Cancer	9	0
	Cancer	6	62

Table VIII
RANDOM FOREST FIT ON HOUSEKEEPING GENES (OLD DATA)

	Accuracy	Precision	Recall	F1
No Cancer	0.9221	1	0.60	0.75
Cancer	0.9221	0.91	1	0.95

Table IX
RANDOM FOREST PERFORMANCE MEASURES ON HOUSEKEEPING GENES (OLD DATA)

The fit on housekeeping genes can be seen in Table VIII and its performance measures in Table IX.

This issue with regards to the dataset drove the group to try and find another set of data to use, possibly more accurate.

B. Second part: new dataset

The decision was to use the data from the Gene Expression Omnibus with id GSE45827[4]. This dataset contains 178 samples of expression data from Breast cancer subtypes coming from 155 different patients. This data was also created with the same Affymetrix Human Genome U133 Plus 2.0 Array platform. Additionally, this new dataset is stratified differently with respect to the previous one, as the variable containing the condition of the patient is not binary as before, but instead is divided in 5 levels that correspond to different breast cancer subtypes:

- 0 Normal (no cancer)
- 1 Basal
- 2 Her2
- 3 Luminal A
- 4 Luminal B

The analysis of the new data yields significantly better results. The first model to fit was a Random Forest on the genes of interest, with the following performances during cross validation (best to worst):

mtry*	**Measure**	**Mean**	**N. folds**	**Std. error**
10	F1	0.7872726	5	0.03857438
22	F1	0.7630541	5	0.03922816
16	F1	0.7593441	5	0.04674310
28	F1	0.7388160	5	0.04477342
58	F1	0.7301482	5	0.04794219

*mtry is the number of predictors used for the splits

[4]https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE45827

Testing the model against the test set yields the following confusion matrix:

		Truth				
		Basal	HER	L. A	L. B	Normal
Predicted	Basal	9	2	0	1	0
	HER	2	5	0	1	0
	L. A	0	1	6	2	0
	L. B	0	0	1	4	0
	Normal	0	0	0	0	1

From the matrix we can extract an overall accuracy of 71.43% and the following measures:

	Accuracy	Precision	Recall	F1
Basal	0.8571	0.75	0.82	0.78
HER	0.8286	0.63	0.63	0.63
L. A	0.8857	0.67	0.86	0.75
L. B	0.8833	0.60	0.75	0.67
Normal	1	1	1	1

When extracting the variable importance from the model, seen in Figure 11, it is interesting to notice that SNORD50B and its host gene SNHG5 are in the top 10 most important genes for classification.
Fitting the model with only the lesser variant genes, the performance drops, as can be seen from the cross validation data:

mtry*	**Measure**	**Mean**	**N. folds**	**Std. error**
178	F1	0.5038095	5	0.05853361
112	F1	0.4933707	5	0.05767972
154	F1	0.4882505	5	0.07213041
118	F1	0.4865629	5	0.06589169
142	F1	0.4790185	5	0.08137526

*mtry is the number of predictors used for the splits

Given the test set data, this latter model outputs the confusion matrix in Table X with its performance measures in Table XI.

		Truth				
		Basal	HER	L. A	L. B	Normal
Predicted	Basal	7	6	2	2	0
	HER	2	0	1	1	0
	L. A	1	1	1	1	0
	L. B	1	1	3	4	1
	Normal	0	0	0	0	0

Table X
RANDOM FOREST CONFUSION MATRIX ON LESSER VARIANT GENES

	Accuracy	Precision	Recall	F1
Basal	0.6	0.41	0.64	0.5
HER	0.65	0	0	0
L. A	0.74	0.24	0.14	0.18
L. B	0.71	0.40	0.5	0.44
Normal	0	0	0	0

Table XI
RANDOM FOREST PERFORMANCE METRICS ON LESSER VARIANT GENES

Next, an SVM was fit to the genes of interest. The hyperparameter tuning performed with a cross validation found the best cost for the model to be 0.005332322.

The results obtained are slightly better than the Random Forest, namely:

cost	Measure	Mean	N. folds	Std. error
0.005332322	F1	0.8063554	5	0.02904139
0.006592777	F1	0.8063554	5	0.02904139
0.004312850	F1	0.7981699	5	0.02556259
0.008151179	F1	0.7978359	5	0.02487277
0.010077956	F1	0.7864484	5	0.01755194

The final model obtained the values in Table XII, with the confusion matrix in Table XIII and the performance measures in Table XIV.

Property	Value
Cost	0.005332322
Support Vectors	92/102
Training Error	0.019608

Table XII
FINAL SVM MODEL

		Truth				
		Basal	HER	L. A	L. B	Normal
Predicted	Basal	10	2	0	1	0
	HER	1	6	0	1	0
	L. A	0	0	7	2	0
	L. B	0	0	0	4	0
	Normal	0	0	0	0	1

Table XIII
FINAL SVM MODEL CONFUSION MATRIX

	Accuracy	Precision	Recall	F1
Basal	0.8857	0.77	0.91	0.83
HER	0.8857	0.75	0.75	0.75
L. A	0.9429	0.78	1	0.88
L. B	0.8857	1	0.5	0.67
Normal	1	1	1	1

Table XIV
FINAL SVM MODEL PERFORMANCE MEASURES

We decided to try a logistic regression on all coefficients, but introducing a penalty so to not have a model with high variance.

The model fit for this phase was the logistic regression with a L1 penalty. The performance is noticeably good, as seen in Table XV, Table XVI and Table XVII:

penalty	Measure	Mean	N. folds	Std. error
6.250552e-01	F1	0.8195279	5	0.04381708
1.264855e-05	F1	0.8137864	5	0.03345629
1.206793e-06	F1	0.8099060	5	0.04936689
1.757511e-08	F1	0.8039975	5	0.04661561
6.866488e-09	F1	0.8030488	5	0.04050916

Table XV
LOGISTIC REGRESSION WITH L1 PENALTY CROSSVALIDATION RESULTS

Through this model, the most important and least important variables can also be extracted. Specific results can be seen in Table XVIII and Table XIX.

This result suggests that the classification is in fact significant and that the expression levels of the selected snoRNAs with their host genes can, in fact, be used as a predictor for different cancer subtypes.

		Truth				
		Basal	HER	L. A	L. B	Normal
Predicted	Basal	9	1	0	1	0
	HER	2	5	0	0	0
	L. A	0	1	6	3	0
	L. B	0	1	1	4	0
	Normal	0	0	0	0	1

Table XVI
LOGISTIC REGRESSION WITH L1 PENALTY CONFUSION MATRIX

	Accuracy	Precision	Recall	F1
Basal	0.8857	0.82	0.82	0.82
HER	0.8571	0.71	0.63	0.67
L. A	0.8571	0.60	0.86	0.71
L. B	0.8286	.67	0.5	0.57
Normal	1	1	1	1

Table XVII
LOGISTIC REGRESSION WITH L1 PENALTY PERFORMANCE MEASURES

Coefficient (Probe id)	Estimate	Gene Symbols
214744_s_at	-5.807828e-01	RPL23
228971_at	-5.005921e-01	SLC25A3
232355_at	-4.876120e-01	MEG8, SNORD114-3
235102_x_at	4.474731e-01	SNORD3B-1, SNORD3B-2, SNORD3D, SNORD3A, SNORD3C
223666_at	4.409927e-01	SNX5
231096_at	-4.260707e-01	PCAT4
242856_at	-4.194403e-01	MEG8, SNORD114-9, SNORD114-10
221621_at	3.693238e-01	SNHG20
228879_at	-3.637258e-01	SNORD104, SNORA50C
241448_at	2.737583e-01	TOMM20

Table XVIII
MOST IMPORTANT VARIABLES

Coefficient (Probe id)	Estimate	Gene Symbols
1555177_at	0	PRKAA1
215011_at	0	SNHG3
229038_at	0	CWF19L1
226428_at	0	TNPO2
240083_at	0	MEG8
200716_x_at	-8.590805e-05	RPL13AP5, RPL13A
200031_s_at	-1.930631e-04	RPS11
224741_x_at	-2.245289e-04	GAS5
200692_s_at	2.554860e-04	HSPA9
209476_at	-2.689458e-04	TMX1

Table XIX
LEAST IMPORTANT VARIABLES

C. Network analysis

It should be noted that, unfortunately, not all genes could be used. At the time of writing (08-12-2022), the genes in Table XX had not yet been expanded. In total, for the remaining genes, 396 isoforms were gathered from gene@home.

PCAT4	SNORD114-14	RPL23
SNORD114-1	RPS3	SNORD114-19
RACK1	SNORD113-4	NAN
SNORD114-13	RPS11	RPS2
SNORD114-20	SNHG20	SNORD115-23
SNORD116-13	CDKN2B-AS1	SNORD113-3
SNORD115-32	SLC25A3	ZFAS1
HIF1A-AS2	SNORD114-21	

Table XX
LIST OF GENES NOT YET EXPANDED BY GENE@HOME

Two versions of the network plot were created: one with ignored edges between isoforms of the same gene and the other with complete edges. To make the analysis meaningful, only connections with a relative frequency of at least 0.85 were initially kept. The complete networks can be seen in Figure 16 and Figure 17.

From both networks, it is noticeable the presence of a few genes that are only connected to themselves. These are:

- TMX1
- MYRIP
- PRKAA1
- SNORD1B

Already from the complete network, it looks like the snoRNAs are very connected among themselves. From the circular network with ignored edges (Figure 18), it is clear that areas with many snoRNAs are densely connected, while areas with many host genes are less densely connected.

By plotting only the core network with the most connections (Figure 20) it is noticeable how the vast majority of genes are snoRNAs, with only a few host genes. This could suggest that snoRNAs tend to be connected with eachother or that OneGene is more able to capture relations between snoRNAs.

Trying to validate the network only yields only 11/231 connections validated. This poor result led to a series of tests with consequently lower threshold for the relative frequency of the connections.

With both a 0.90 and a 0.85 threshold the results are still the same, so we went for a more drastic cut with a 0.5 threshold. This gave better results, with 22/231 connections found. Also, in the network plot seen in Figure 19 there are no more genes connected to themselves, but there are many connections with a Pearson correlation < 0.

Lowering yet again the threshold at 0.3 gives the same result as 0.5, so we went for a drastic measure and reduced the threshold to 0.1.

This only brings up the validated connections to 26. Such a low threshold does not make much sense because it could introduce many connections which might not be correct in the real world due to the algorithm picking up noise in the data or simply establishing random connections.

We can further analyze the network with a 0.85 threshold. In particular, we can check which are the connections that are validated with the help of snoDB. These are:

- SNORD41 $\rightarrow$ SNORD60
- SNORA73B $\rightarrow$ SNORA73A
- SNORA78 $\rightarrow$ SNORD28
- SNORD50A $\rightarrow$ SNORD59A
- SNORD20 $\rightarrow$ SNORA75
- SNORD108 $\rightarrow$ SNORD64
- SNORD12 $\rightarrow$ SNORD12B
- SNORD47 $\rightarrow$ SNORD59A
- SNORD47 $\rightarrow$ SNORD60
- SNORA70 $\rightarrow$ SNORD52
- SNORD94 $\rightarrow$ SNORD22

This small number of connections validated could also be due to the fact that traditional techniques for RNA-Seq, like CAGE that is used to generate data in FANTOM, seem to underestimate snoRNA expression level. This is because snoRNAs are structured RNAs and polymerase cannot correctly sequence them [23].

snoRNA	**Host Gene**
SNORD18A	RPL4
SNORD20	NCL
SNORA73A	SNHG3
SNORA14B	TOMM20
SNORA38	PRRC2A
SNORD82	NCL
SNORA78	SNHG9
SNORA48	NFATC3
SNORD68	RPL13
SNORD97	EIF4G2
SNORA75	NCL
SNORD37	EEF2

Table XXI
LIST OF CONNECTIONS BETWEEN SNORNAS AND THEIR REPECTIVE HOST GENES FOUND IN THE NETWORK AND SNODB

D. Differential gene expression

The first step for this analysis was to shrink the data down to only our genes of interest, both snoRNAs and host genes. This brought the number of rows (genes) of the dataset from 29873 to 161. The data were then transformed into log2. The reason for that is that when using log-transformed expression (or concentration) values, the modeled changes are proportional rather than additive. This is typically biologically more relevant [24]. In addition, errors are usually proportional to the values. This kind of mean-variance relationship is usually absent on the logarithmic scale [24].

The next step was to assign the samples to their respective group. The groups have been defined by following the cancer subtype, namely: The cell lines samples have been excluded

Group number	**Cancer subtype**
0	Normal (no cancer)
1	Basal
2	Her2
3	Luminal A
4	Luminal B

because we focused our research only on healthy and cancer patients.

After preparing the data, we can now begin the proper analysis. Initially, the check was to see the significance of the genes in terms of p-value. The normal test assumption is that most genes are not differentially expressed.

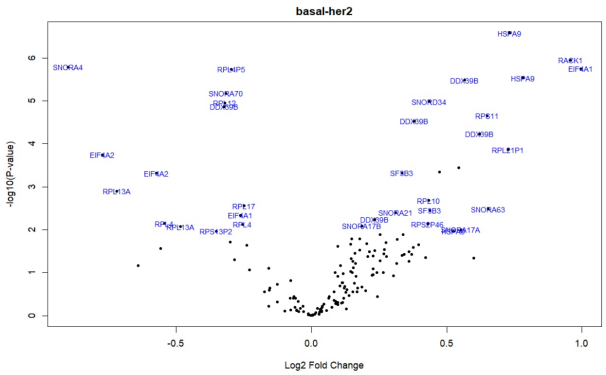

Figure 3. Volcano plot Basal VS Her2

We also obtained the volcano plots, useful for identifying events that differ significantly between two groups of experimental subjects.
In Figure 3 is reported the comparison between the Basal group and the HER2 group. Each point on the graph represents a gene. The log_2-fold differences between the groups are plotted on the x-axis and the $-log_{10}$ p-value differences are plotted on the y-axis. We can see that the genes whose expression is decreased from the basal condition to the HER2 condition are located to the left zero on the x-axis. The genes whose expression is increased are illustrated to the right of zero. Closer to zero indicates fewer changes, while moving away from zero in either direction indicates more changes. The most statistically significant genes are reported in blue, along with their gene name. The volcano plot for all the other conditions are reported in Appendix (Figure 23, 24, 25 and 26).

With the Venn diagram in Figure 4, we look at the conditions that share differentially expressed genes. The number in each circle represents the amount of differentially expressed genes between the different comparisons. The overlapping number stands for the mutual differentially expressed genes between the different comparisons and the non overlapping numbers specify the genes unique to each condition.

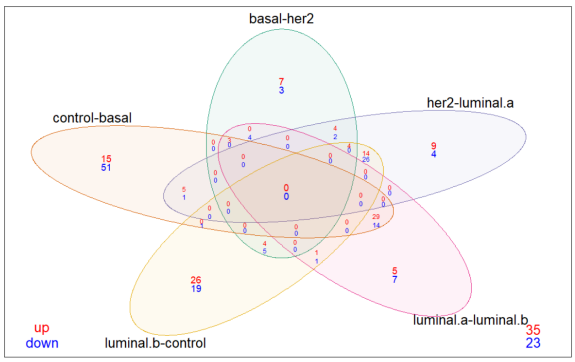

Figure 4. Venn diagram showing both upregulated (in red) and downregulated (in blue) genes.

The Venn diagrams showing only the up- and down-regulated genes are reported in the Appendix (Figure 28 and 29).

With a Q-Q plot for the t-statistic (Figure 5), we can assess whether the sample quantiles are comparable to the theoretical quantiles of the t-distribution. In general, the Q-Q plot compares two probability distributions by plotting their quantiles against each other.

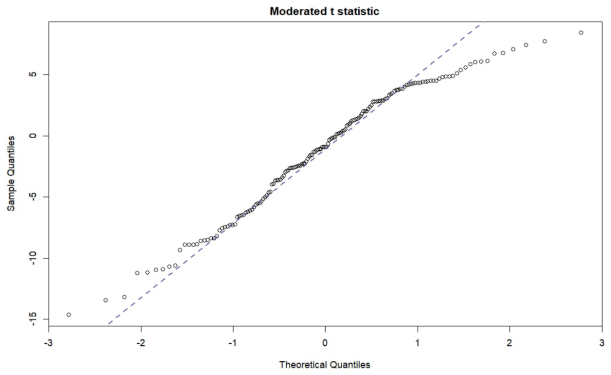

Figure 5. QQplot for t-statistic

In this case, it appears that the samples follows a t-distribution, with the exception of the tails.

Applying a PCA dimensionality reduction (Figure 6), we can notice that the cancer subtypes and the control samples can be easily separated. Among the cancer subtypes, the luminal A and luminal B are the one that are mostly mixed together. This is another indicator that the samples come indeed from different cancer subtypes and the control samples can be safely used for their control role.

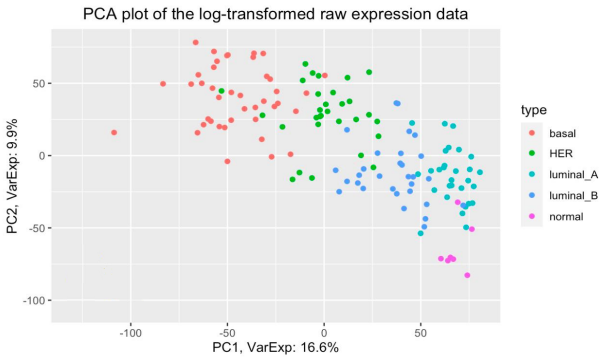

Figure 6. PCA plot of the log-transformed raw expression data.

Finally, we analyze the mean variance of the samples. In Figure 7, means (x-axis) and variances (y-axis) of each gene are plotted to show the dependence between the two. In particular, the plot on created using the function *plotSA*, which plots log_2 residual standard deviations against mean log-CPM values. The average log_2 residual standard deviation is marked by a horizontal blue line.

IV. Conclusions

After performing the analyses described in the previous sections, all techniques employed gave us at least some interesting

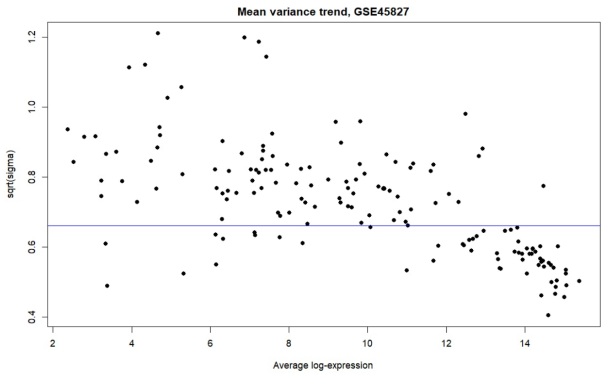

Figure 7. Mean Variance trend

insight into our genes of interest and correlation amongst themselves. Firstly, as far as anomalies in the expression of our genes of interest in the presence of breast cancer, the Random Forest model gave the most insightful results, placing SLC25A3, PTCD3 and SNX5 as the top 3 variables as far as importance goes. Although they are less significant in this context, since coefficients in multinomial Lasso are calculated with respect to a reference variable, such coefficients seem to agree with the Random Forest results, having both SLC25A3 and SNX5 in the top 5 as far as their influence in the Lasso model. Such results can be seen in Figure 11 and Table XVIII. Furthermore, results with the previously mentioned models, as well as the other ones used in the analysis (SVM above all), revealed that the expression of our genes of interest is able to discriminate the various subtypes of cancer, as well as the control group with relatively high performance. Making this conclusion even more robust is the significantly worse performance of the same algorithms performed on a list of housekeeping genes.

The DEG analysis also confirmed the ability of the gene expression to discriminate the subtypes of cancer.

The expansion of our genes of interest allowed us to look further into the correlation of our gene network, revealing strong connections among the snoRNAs, but only biologically validating few connections.

Additionally, an interesting result is that the gene TOMM20 has a discrete importance for the Random Forest model (Figure 11 and it is associated with one of the most important coefficients in the lasso model (Table XVIII). Also, there seems to be a connection between TOMM20 and the associated snoRNA, SNORA-14B. Unfortunately, as SNORA-14B is missing in the our breast cancer dataset, we couldn't perform an additional validation on the relationship between TOMM20 and SNORA-14B.

In the future, the relationship between TOMM20 and SNORA-14B could be further investigated to understand better whether there is any meaningful link between the two or not.

REFERENCES

[1] D. Zhang, J. Zhou, J. Gao, R.-Y. Wu, Y.-L. Huang, Q.-W. Jin, J.-S. Chen, W.-Z. Tang, and L.-H. Yan, "Targeting snornas as an emerging method of therapeutic development for cancer," *American journal of cancer research*, vol. 9, no. 8, p. 1504, 2019.

[2] J. van der Werf, C. Chin, and N. Fleming, "SnoRNA in cancer progression, metastasis and immunotherapy response," *Biology*, vol. 10, no. 8, p. 809, Aug. 2021. [Online]. Available: https://doi.org/10.3390/biology10080809

[3] D. Bergeron, C. Laforest, S. Carpentier, A. Calvé, É. Fafard-Couture, G. Deschamps-Francoeur, and M. S. Scott, "SnoRNA copy regulation affects family size, genomic location and family abundance levels," *BMC Genomics*, vol. 22, no. 1, Jun. 2021. [Online]. Available: https://doi.org/10.1186/s12864-021-07757-1

[4] M. Yoshihama, A. Nakao, and N. Kenmochi, "snOPY: a small nucleolar RNA orthological gene database," *BMC Research Notes*, vol. 6, no. 1, Oct. 2013. [Online]. Available: https://doi.org/10.1186/1756-0500-6-426

[5] L.-M. Lin, Q. Pan, Y.-M. Sun, and W.-T. Wang, "Small nucleolar RNA is potential as a novel player in leukemogenesis and clinical application," *Blood Science*, vol. 3, no. 4, pp. 122–131, Oct. 2021. [Online]. Available: https://doi.org/10.1097/bs9.0000000000000091

[6] G. T. Williams and F. Farzaneh, "Are snoRNAs and snoRNA host genes new players in cancer?" *Nature Reviews Cancer*, vol. 12, no. 2, pp. 84–88, Jan. 2012. [Online]. Available: https://doi.org/10.1038/nrc3195

[7] M. Lizio, , J. Harshbarger, H. Shimoji, J. Severin, T. Kasukawa, S. Sahin, I. Abugessaisa, S. Fukuda, F. Hori, S. Ishikawa-Kato, C. J. Mungall, E. Arner, J. K. Baillie, N. Bertin, H. Bono, M. de Hoon, A. D. Diehl, E. Dimont, T. C. Freeman, K. Fujieda, W. Hide, R. Kaliyaperumal, T. Katayama, T. Lassmann, T. F. Meehan, K. Nishikata, H. Ono, M. Rehli, A. Sandelin, E. A. Schultes, P. A. 't Hoen, Z. Tatum, M. Thompson, T. Toyoda, D. W. Wright, C. O. Daub, M. Itoh, P. Carninci, Y. Hayashizaki, A. R. Forrest, and H. Kawaji, "Gateways to the FANTOM5 promoter level mammalian expression atlas," *Genome Biology*, vol. 16, no. 1, Jan. 2015. [Online]. Available: https://doi.org/10.1186/s13059-014-0560-6

[8] D. Bergeron, H. Paraqindes, É. Fafard-Couture, G. Deschamps-Francoeur, L. Faucher-Giguère, P. Bouchard-Bourelle, S. A. Elela, F. Catez, V. Marcel, and M. S. Scott, "snoDB 2.0: an enhanced interactive database, specializing in human snoRNAs," *Nucleic Acids Research*, Sep. 2022. [Online]. Available: https://doi.org/10.1093/nar/gkac835

[9] V. Pedraza, J. A. Gomez-Capilla, G. Escaramis, C. Gomez, P. Torné, J. M. Rivera, A. Gil, P. Araque, N. Olea, X. Estivill *et al.*, "Gene expression signatures in breast cancer distinguish phenotype characteristics, histologic subtypes, and tumor invasiveness," *Cancer: Interdisciplinary International Journal of the American Cancer Society*, vol. 116, no. 2, pp. 486–496, 2010.

[10] F.-J. Lopez, M. Cuadros, C. Cano, A. Concha, and A. Blanco, "Biomedical application of fuzzy association rules for identifying breast cancer biomarkers," *Medical & biological engineering & computing*, vol. 50, no. 9, pp. 981–990, 2012.

[11] C. Clarke, S. F. Madden, P. Doolan, S. T. Aherne, H. Joyce, L. O'driscoll, W. M. Gallagher, B. T. Hennessy, M. Moriarty, J. Crown *et al.*, "Correlating transcriptional networks to breast cancer survival: a large-scale coexpression analysis," *Carcinogenesis*, vol. 34, no. 10, pp. 2300–2308, 2013.

[12] L. Aswad, S. P. Yenamandra, G. S. Ow, O. Grinchuk, A. V. Ivshina, and V. A. Kuznetsov, "Genome and transcriptome delineation of two major oncogenic pathways governing invasive ductal breast cancer development," *Oncotarget*, vol. 6, no. 34, p. 36652, 2015.

[13] T. Gruosso, V. Mieulet, M. Cardon, B. Bourachot, Y. Kieffer, F. Devun, T. Dubois, M. Dutreix, A. Vincent-Salomon, K. M. Miller, and F. Mechta-Grigoriou, "Chronic oxidative stress promotes h2scpAX/scpprotein degradation and enhances chemosensitivity in breast cancer patients," *EMBO Molecular Medicine*, vol. 8, no. 5, pp. 527–549, Mar. 2016. [Online]. Available: https://doi.org/10.15252/emmm.201505891

[14] L. Gautier, L. Cope, B. M. Bolstad, and R. A. Irizarry, "affy–analysis of affymetrix GeneChip data at the probe level," *Bioinformatics*, vol. 20, no. 3, pp. 307–315, Feb. 2004. [Online]. Available: https://doi.org/10.1093/bioinformatics/btg405

[15] R. A. Irizarry, "Exploration, normalization, and summaries of high density oligonucleotide array probe level data," *Biostatistics*, vol. 4, no. 2, pp. 249–264, Apr. 2003. [Online]. Available: https://doi.org/10.1093/biostatistics/4.2.249

[16] B. M. B. Rafael A. Irizarry, "Summaries of affymetrix GeneChip probe level data," *Nucleic Acids Research*, vol. 31, no. 4, pp. 15e–15, Feb. 2003. [Online]. Available: https://doi.org/10.1093/nar/gng015

[17] B. Bolstad, R. Irizarry, M. Astrand, and T. Speed, "A comparison of normalization methods for high density oligonucleotide array data based on variance and bias," *Bioinformatics*, vol. 19, no. 2, pp. 185–193, Jan. 2003. [Online]. Available: https://doi.org/10.1093/bioinformatics/19.2.185

[18] L. Chen and G. Wong, "Transcriptome informatics," in *Encyclopedia of Bioinformatics and Computational Biology*, S. Ranganathan, M. Gribskov, K. Nakai, and C. Schönbach, Eds. Oxford: Academic Press, 2019, pp. 324–340. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780128096338202045

[19] M. E. Ritchie, B. Phipson, D. Wu, Y. Hu, C. W. Law, W. Shi, and G. K. Smyth, "limma powers differential expression analyses for RNA-sequencing and microarray studies," *Nucleic Acids Research*, vol. 43, no. 7, pp. e47–e47, Jan. 2015. [Online]. Available: https://doi.org/10.1093/nar/gkv007

[20] E. Eisenberg and E. Y. Levanon, "Human housekeeping genes, revisited," *Trends in Genetics*, vol. 29, no. 10, pp. 569–574, Oct. 2013. [Online]. Available: https://doi.org/10.1016/j.tig.2013.05.010

[21] M. Caracausi, A. Piovesan, F. Antonaros, P. Strippoli, L. Vitale, and M. C. Pelleri, "Systematic identification of human housekeeping genes possibly useful as references in gene expression studies," *Molecular Medicine Reports*, vol. 16, no. 3, pp. 2397–2410, Mar. 2017. [Online]. Available: https://doi.org/10.3892/mmr.2017.6944

[22] M. Charikar, "Greedy approximation algorithms for finding dense components in a graph," in *Proceedings of the Third International Workshop on Approximation Algorithms for Combinatorial Optimization*, ser. APPROX '00. Berlin, Heidelberg: Springer-Verlag, 2000, p. 84–95.

[23] Fafard-Couture, D. Bergeron, S. Couture, S. Abou-Elela, and M. S. Scott, "Annotation of snoRNA abundance across human tissues reveals complex snoRNA-host gene relationships," *Genome Biol*, vol. 22, no. 1, p. 172, Jun 2021.

[24] J. Wilhelm, "Why do we usually use log2 when normalizing the expression of genes?" 10 2015.

APPENDIX A
LIST OF ABBREVIATIONS

- BC = Breast Cancer
- GEO = Gene Expression Omnibus
- snoRNAs = Small Nucleolar RNAs
- SNORDs = C/D box snoRNAs
- SNORAs = H/ACA box snoRNAs
- SCARNAs = small Cajal body-specific RNAs
- TSS = Transcriptional starting site
- RMA = Robust Multiarray Averaging
- PM = Perfect Match probe
- MM = Mis-match probe
- DEGs = Differentially Expressed Genes
- DGE = Differential Gene Expression
- PC = Peter-Clark
- DAG = Directed Acyclic Graph
- LGN = Local Gene Network
- L.A = Luminal A
- L.B = Luminal B
- EDA = Exploratory Data Analysis
- PCA = Principal Component Analysis

APPENDIX B
CODE

All the code involved in the various activities is hosted GitHub repository[5].

[5]https://github.com/annalisaxamin/LBDM

APPENDIX C
FIGURES

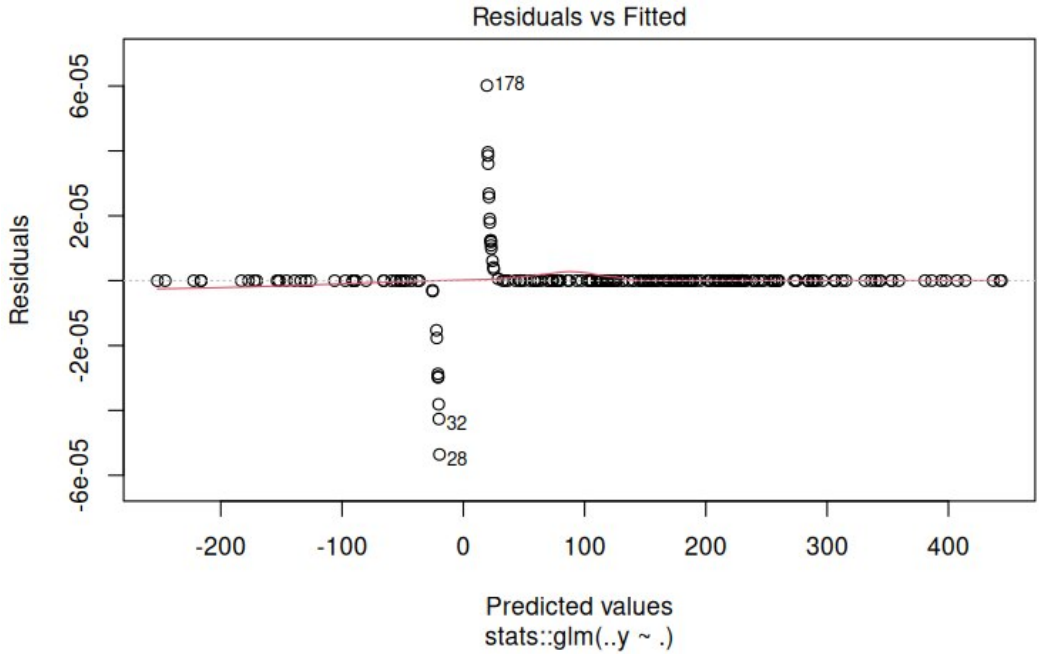

Figure 8. Residuals vs fitted plot on the first dataset

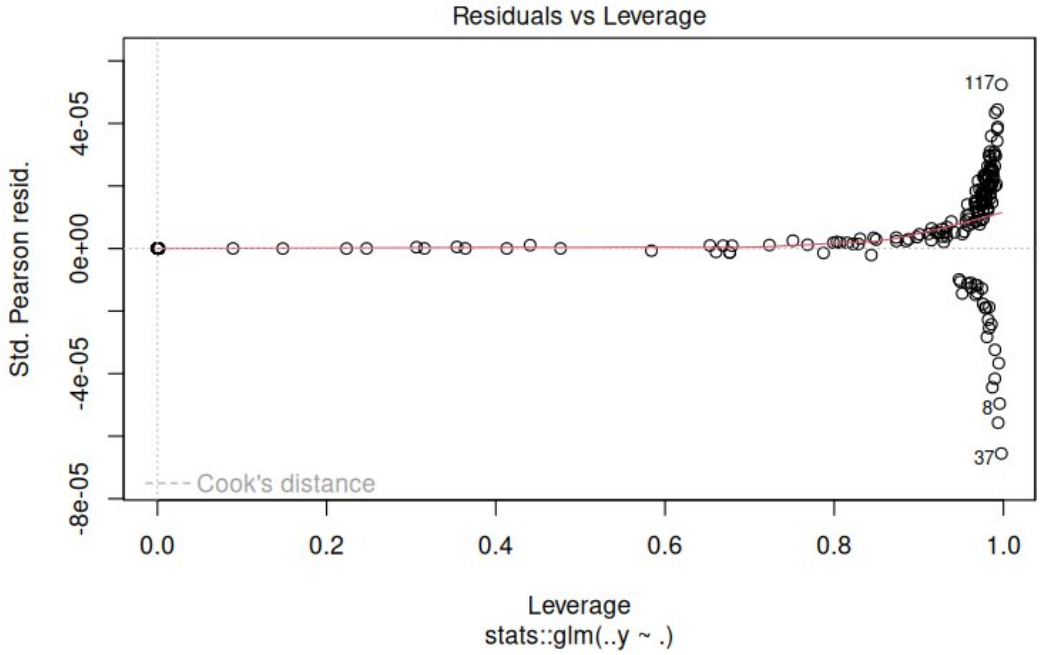

Figure 9. Residuals vs leverage plot on the first dataset

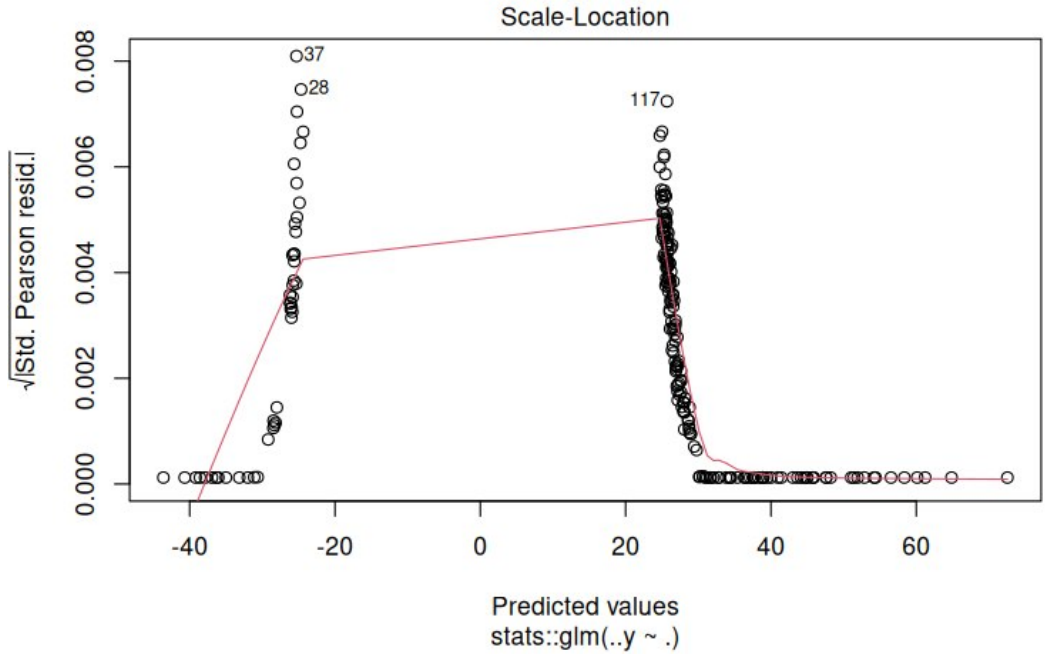

Figure 10. Scale vs location plot on the first dataset

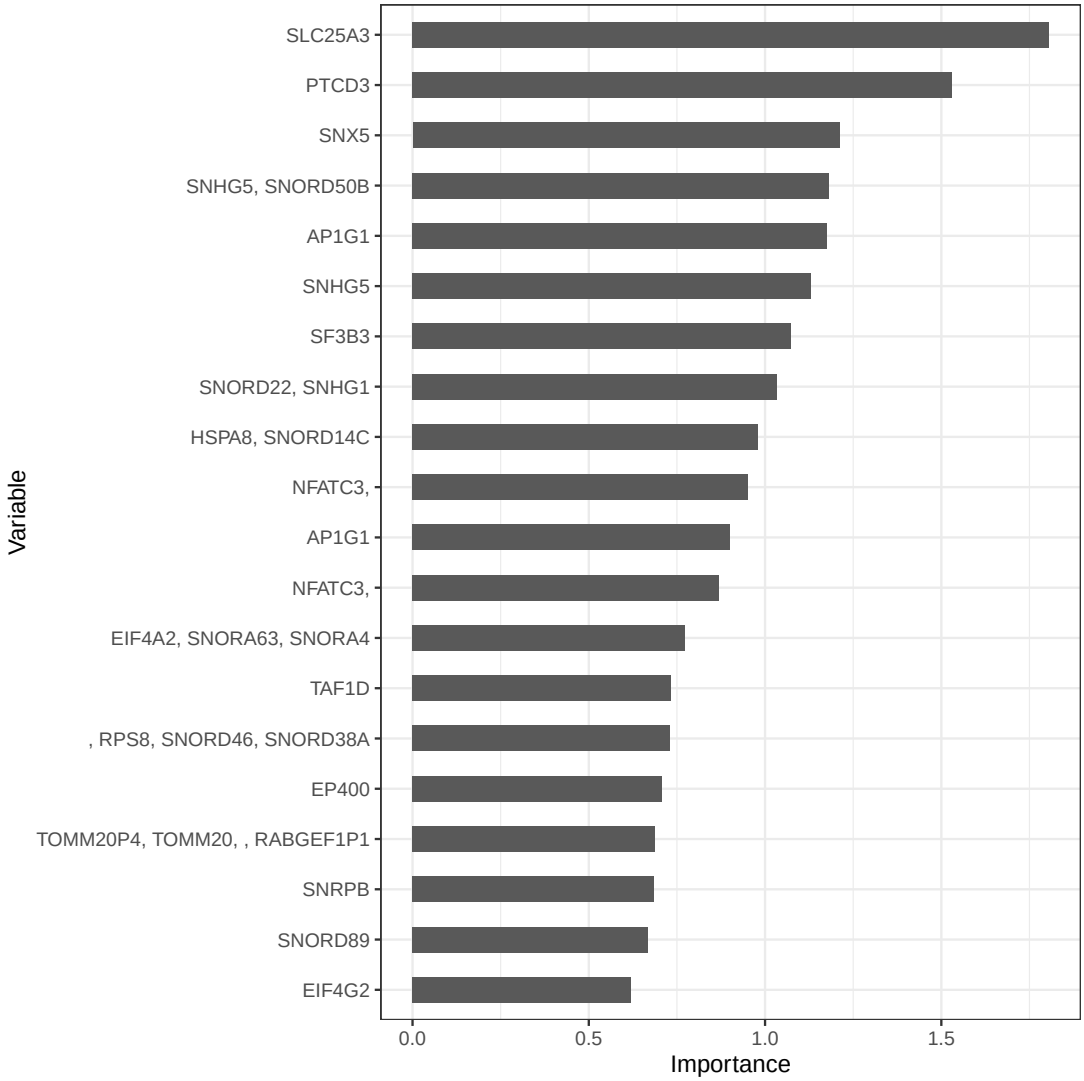

Figure 11. Variable importance extracted from the Random Forest fit on the genes of interest

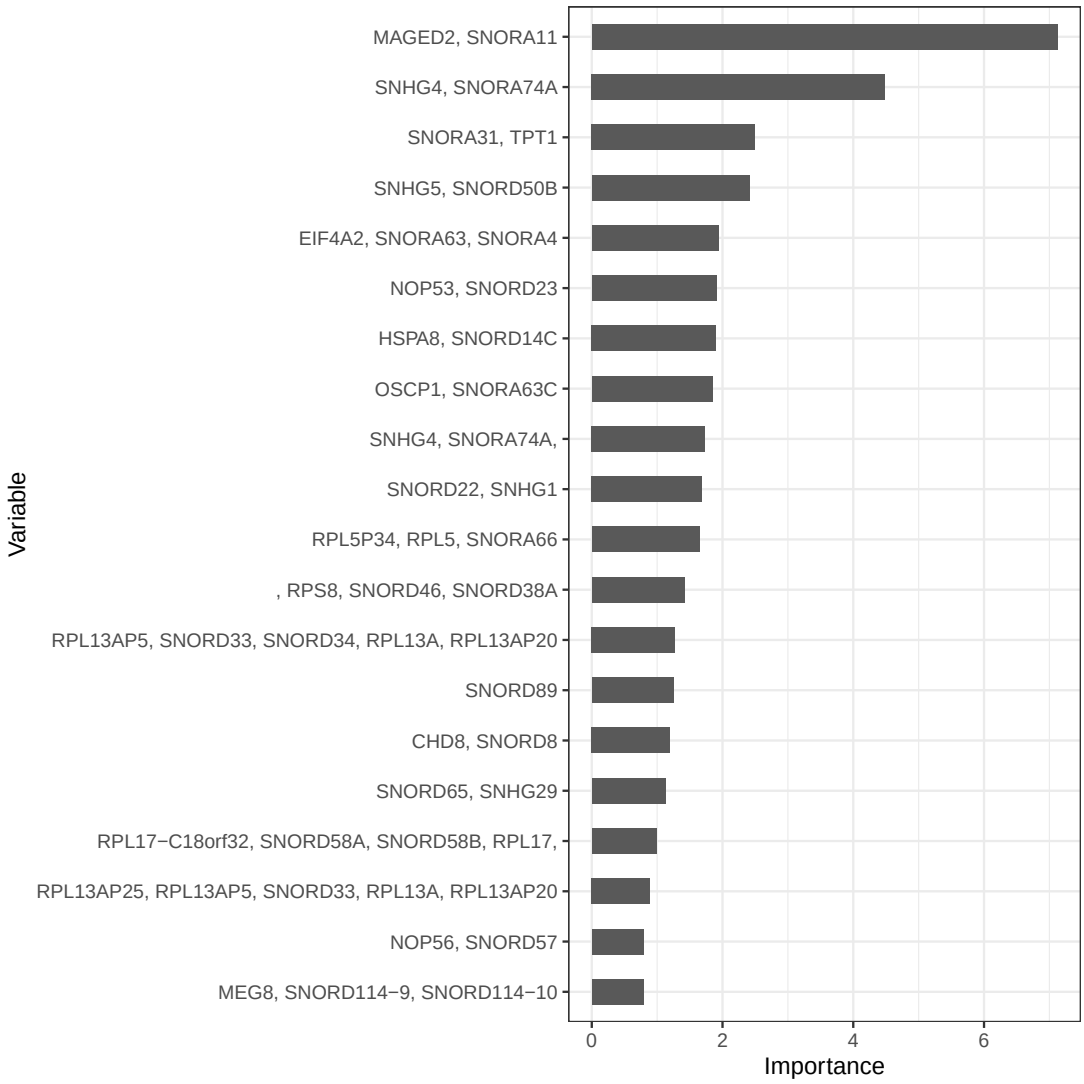

Figure 12. Variable importance extracted from the Random Forest fit on the probes related to snoRNAs

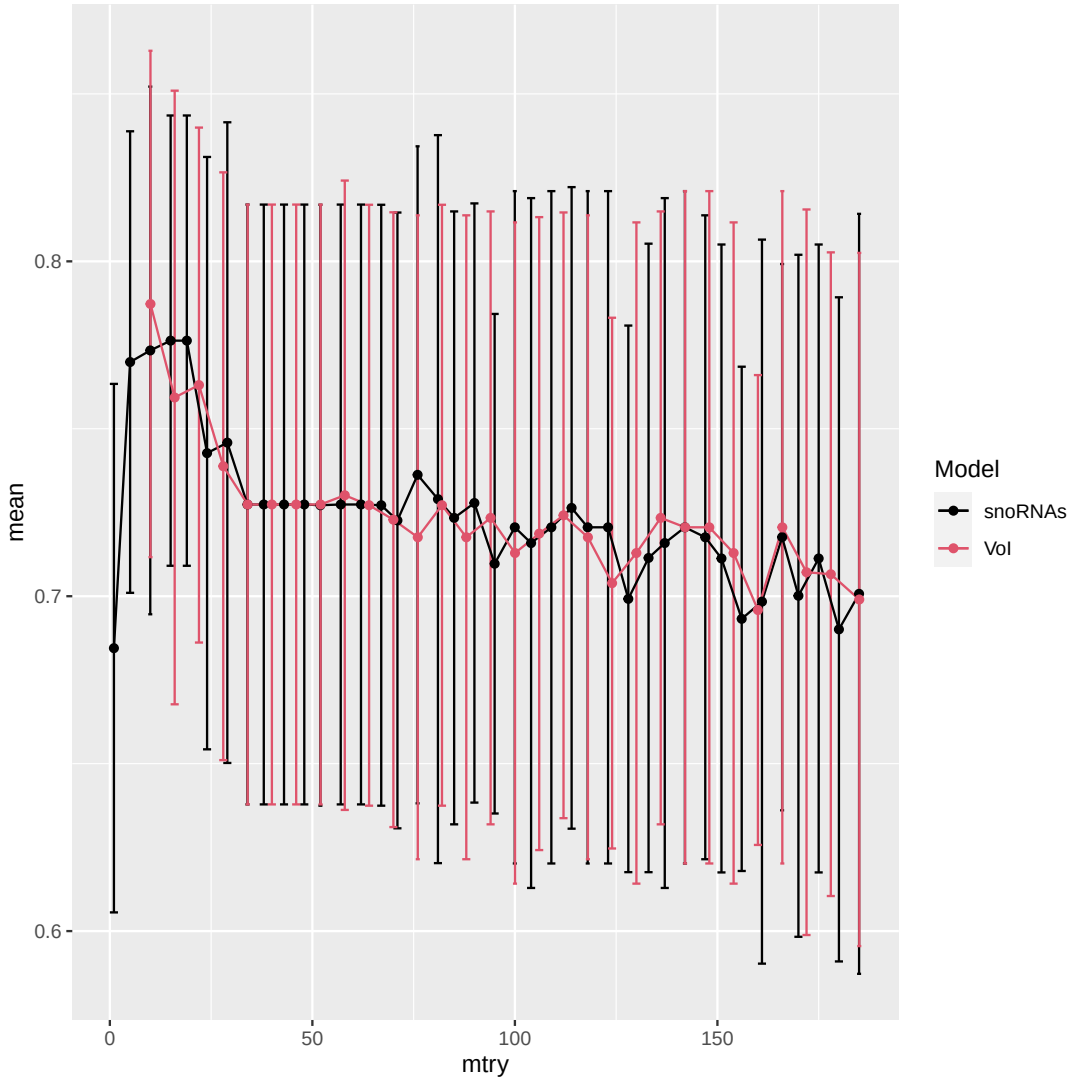

Figure 13. Result of the cross validation where the variation of F1 Measure is in function of mtry

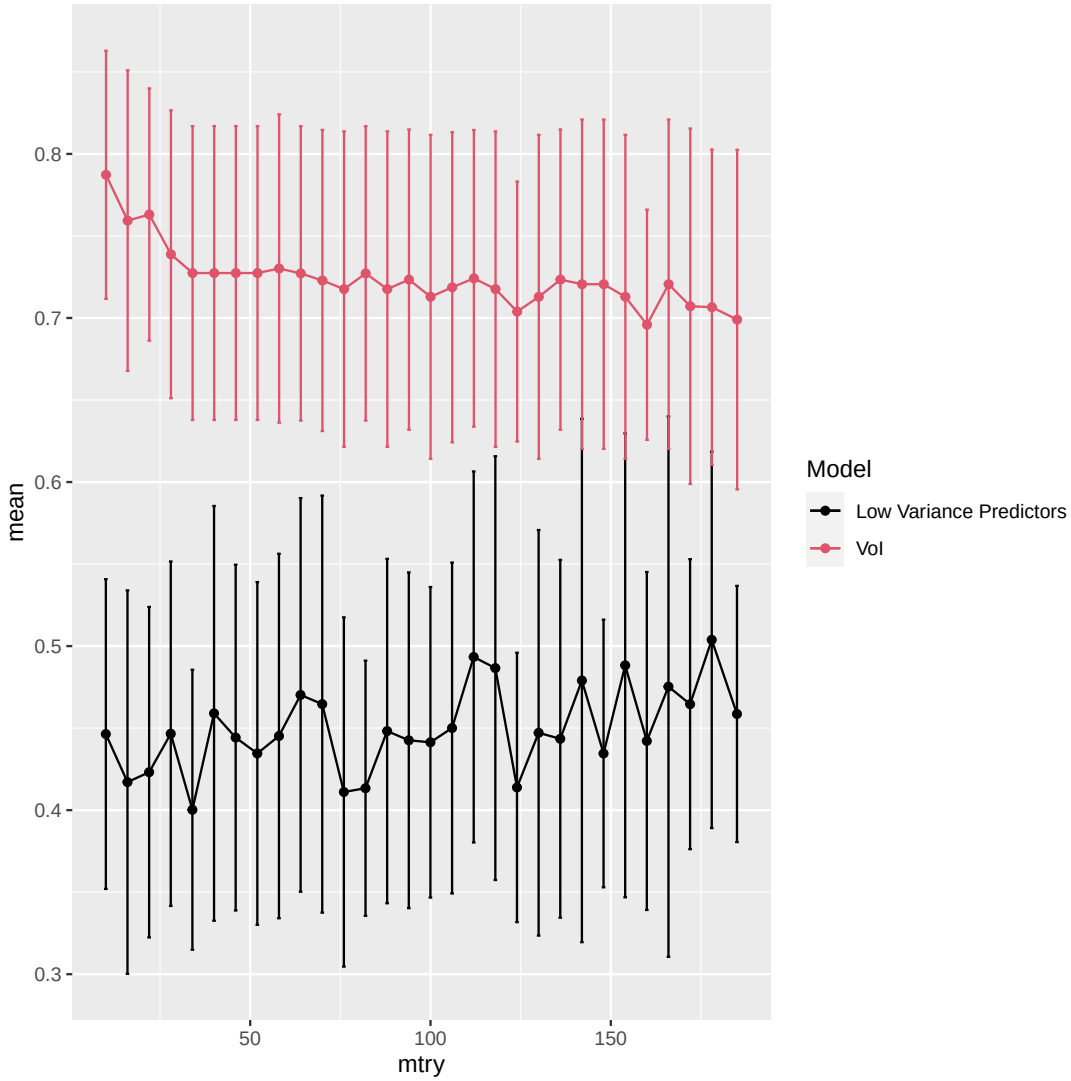

Figure 14. Result of the cross validation where the variation of F1 Measure is in function of mtry

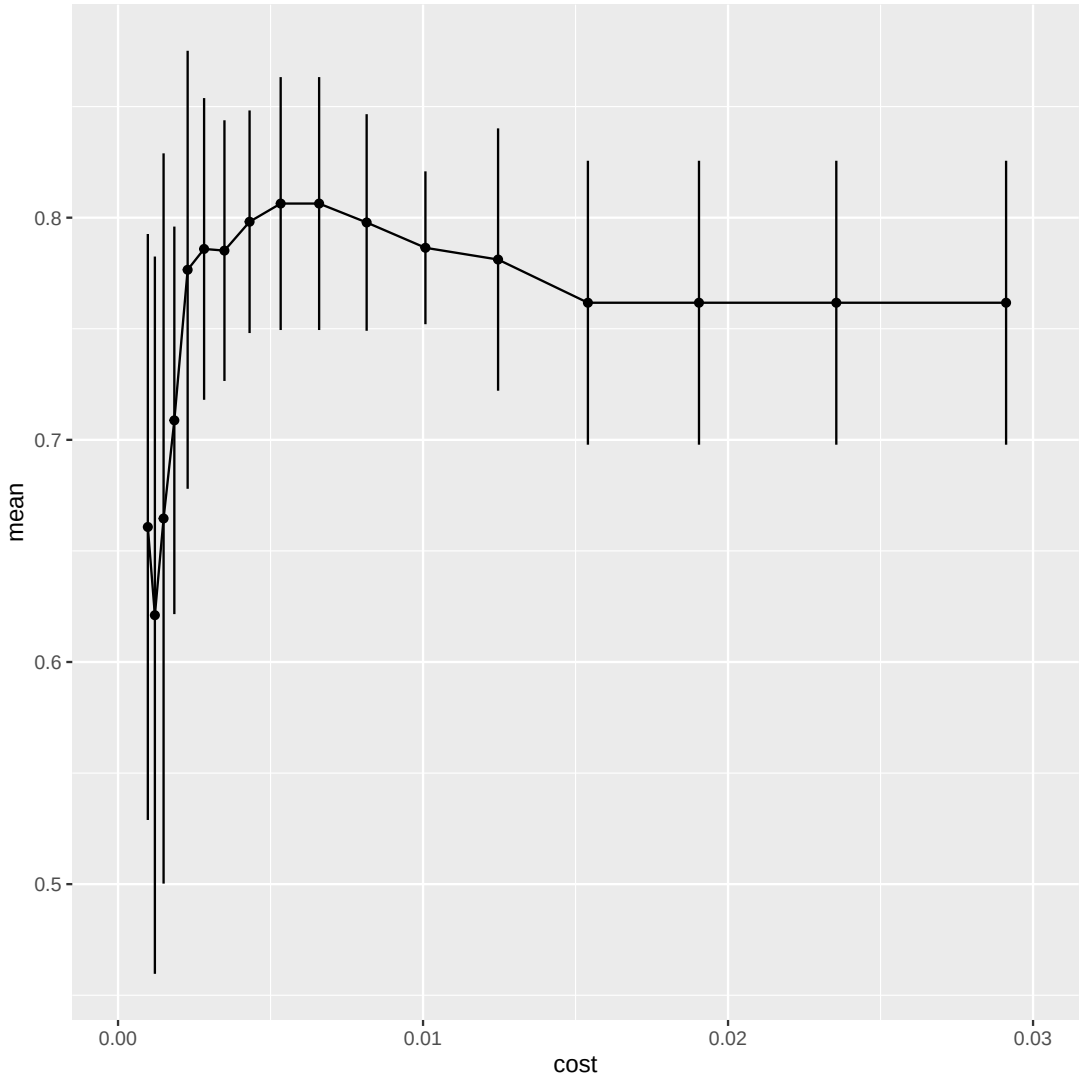

Figure 15. Result of the cross validation where the variation of F1 Measure is in function of the cost

A. Network analysis

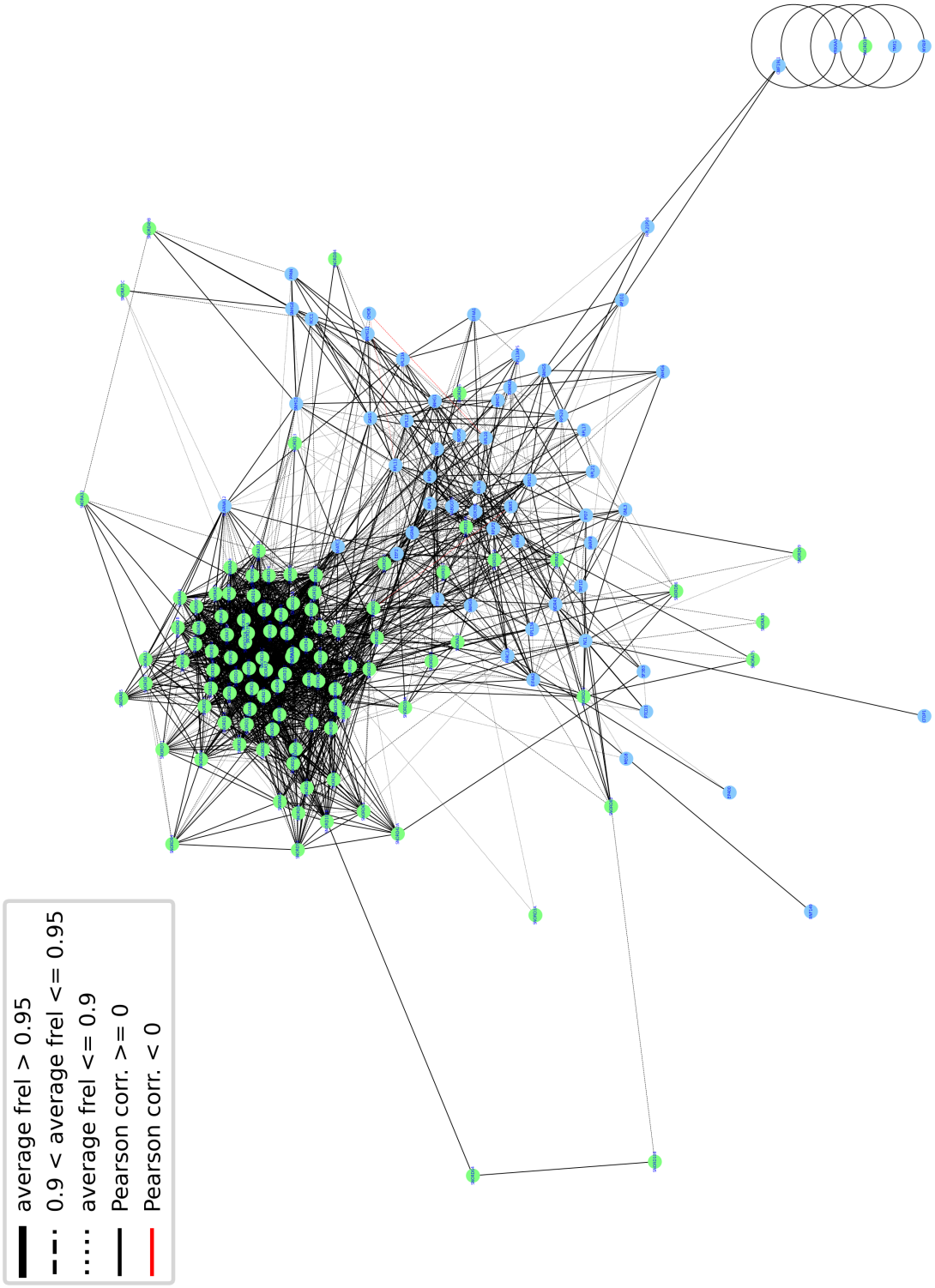

Figure 16. Complete network graph with ignored edges between isoforms of the same gene - 0.85 threshold

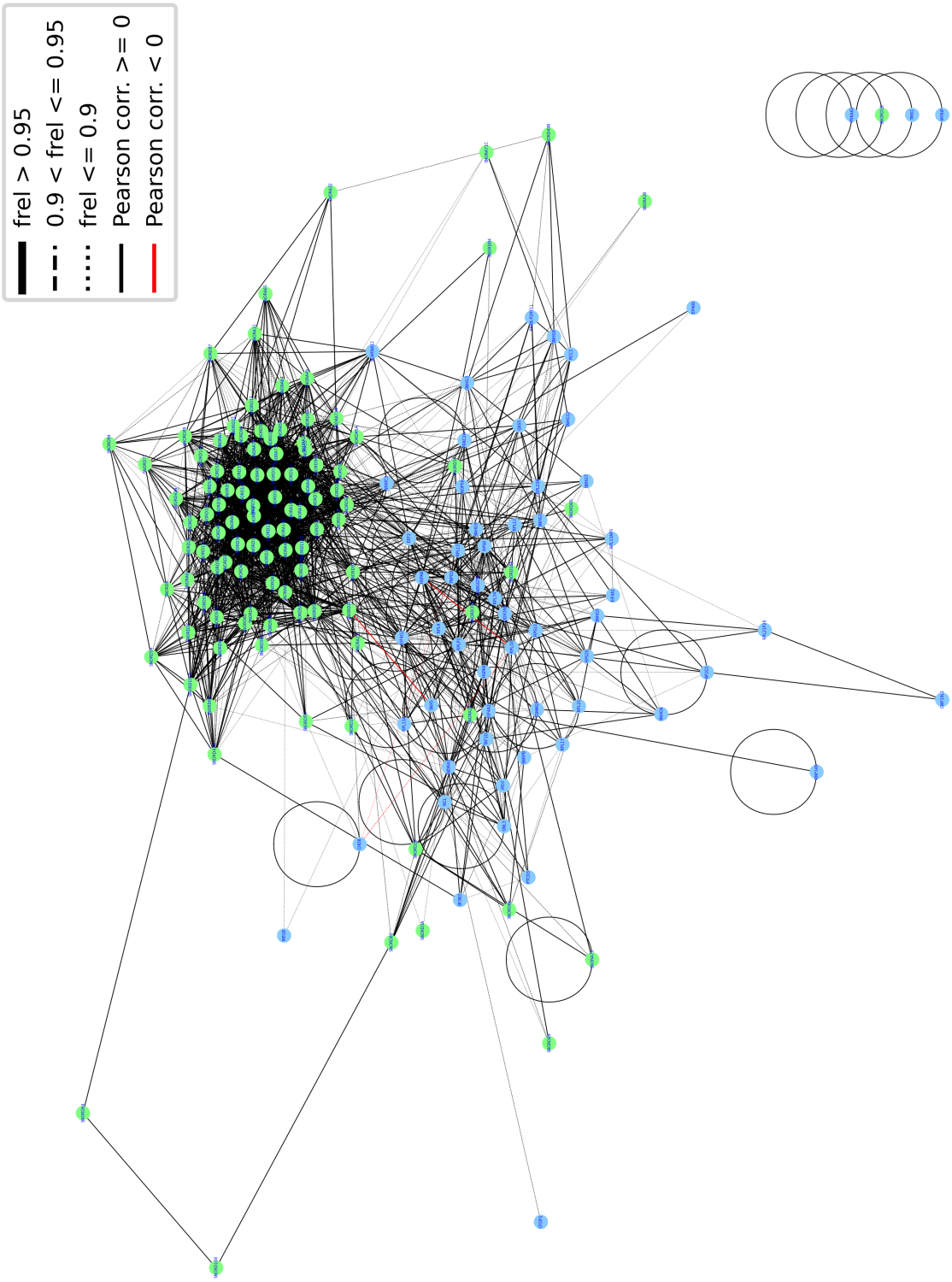

Figure 17. Complete network graph with also edges between isoforms of the same gene - 0.85 threshold

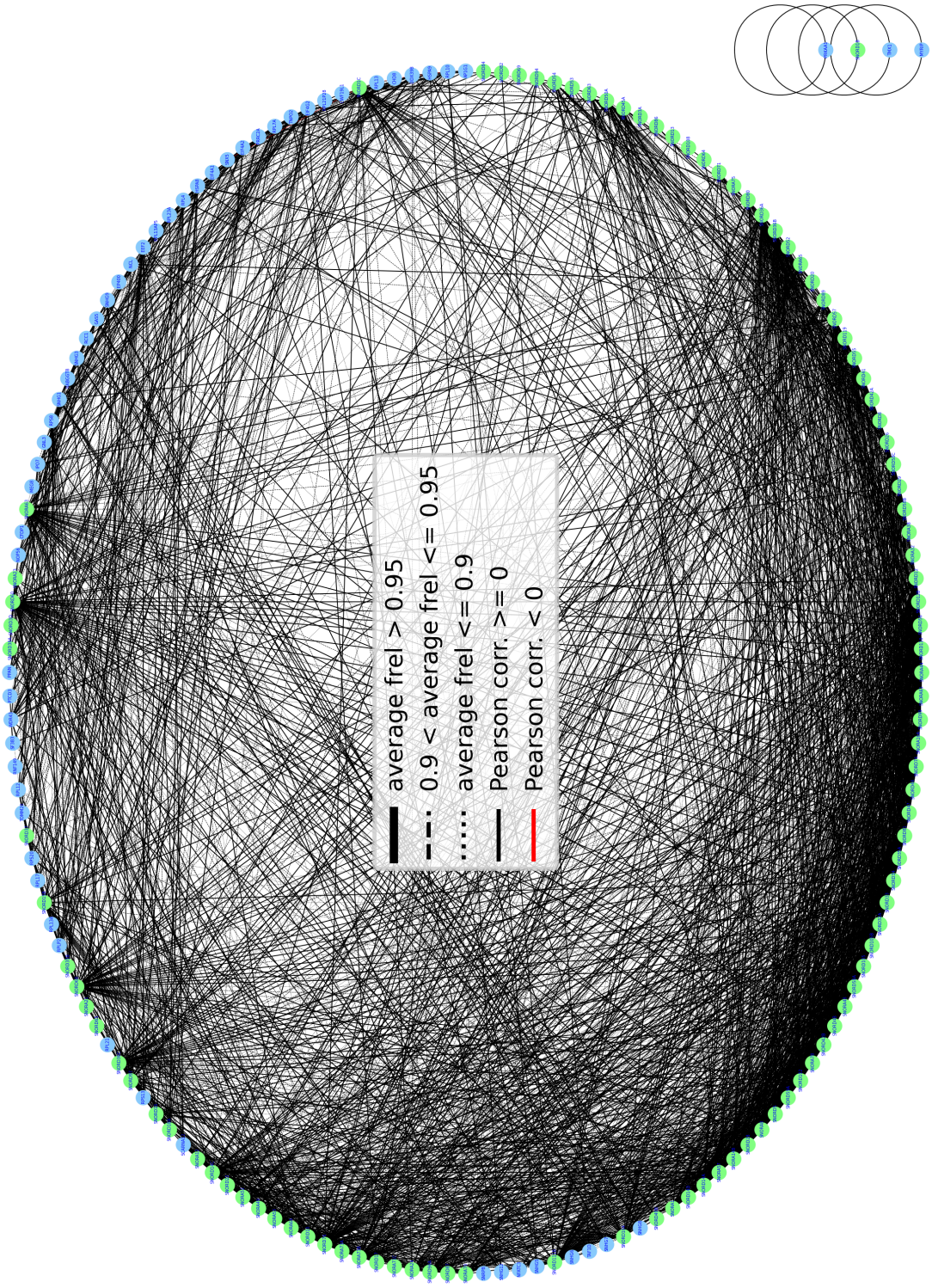

Figure 18. Circular variant of the complete network graph with ignored edges between isoforms of the same gene - 0.85 threshold

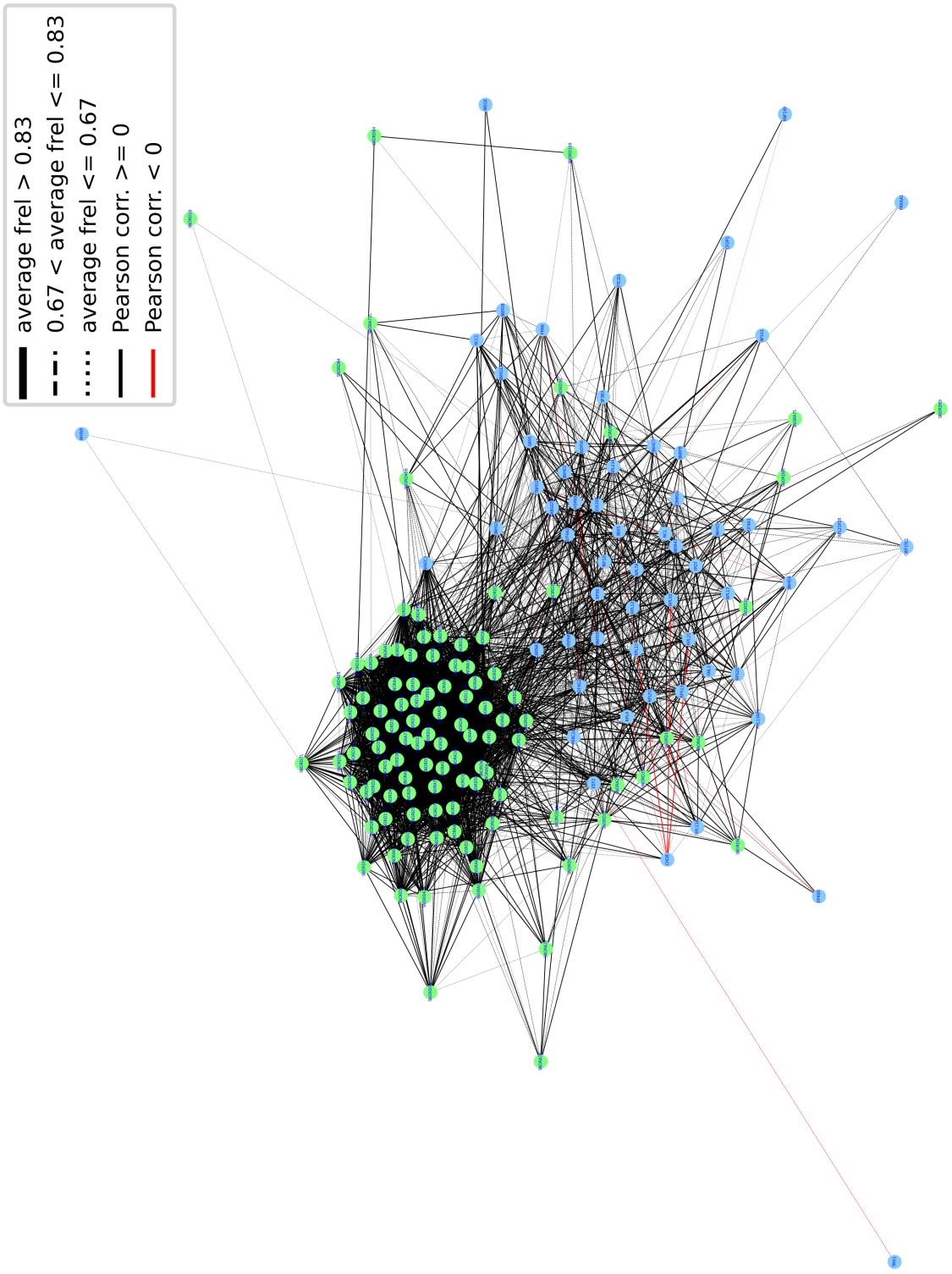

Figure 19. Core network graph with ignored edges between isoforms of the same gene - 0.5 threshold

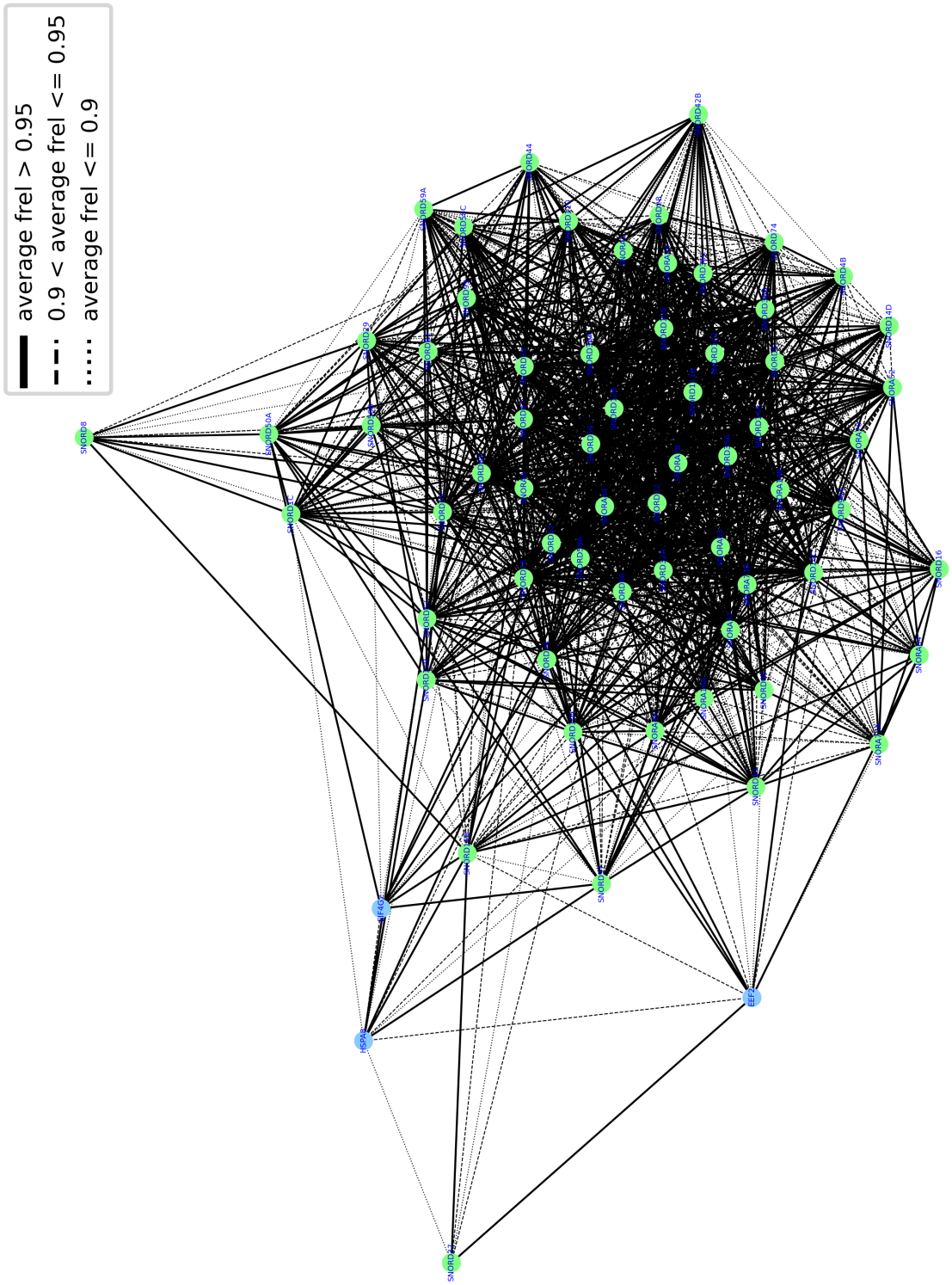

Figure 20. Core network graph with ignored edges between isoforms of the same gene - 0.85 threshold

B. Differential gene expression analysis

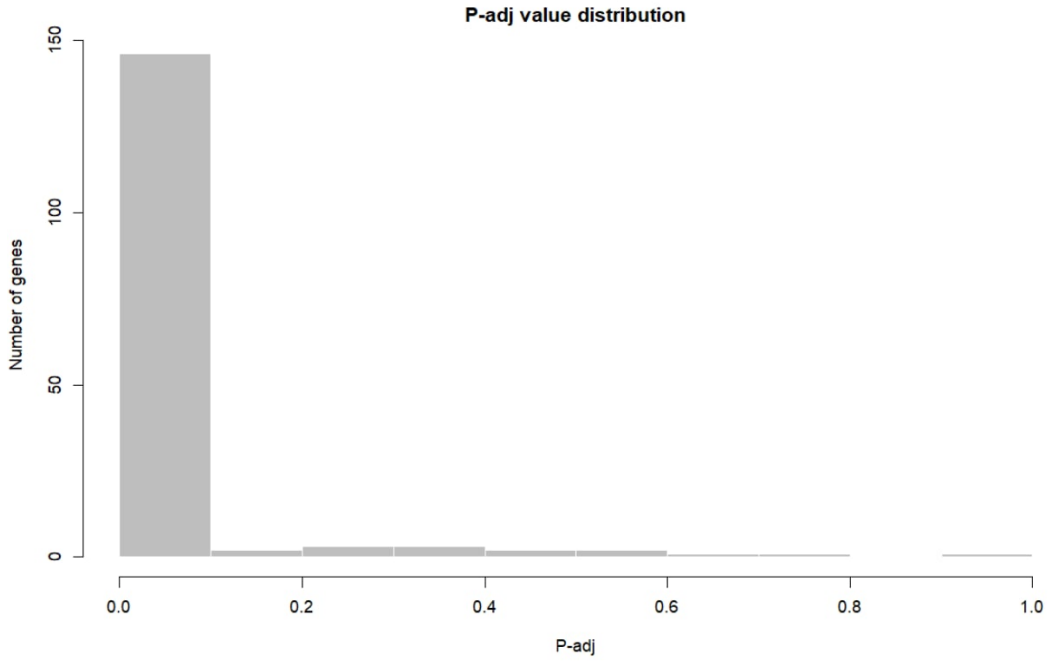

Figure 21. Distribution of adjusted p-value.

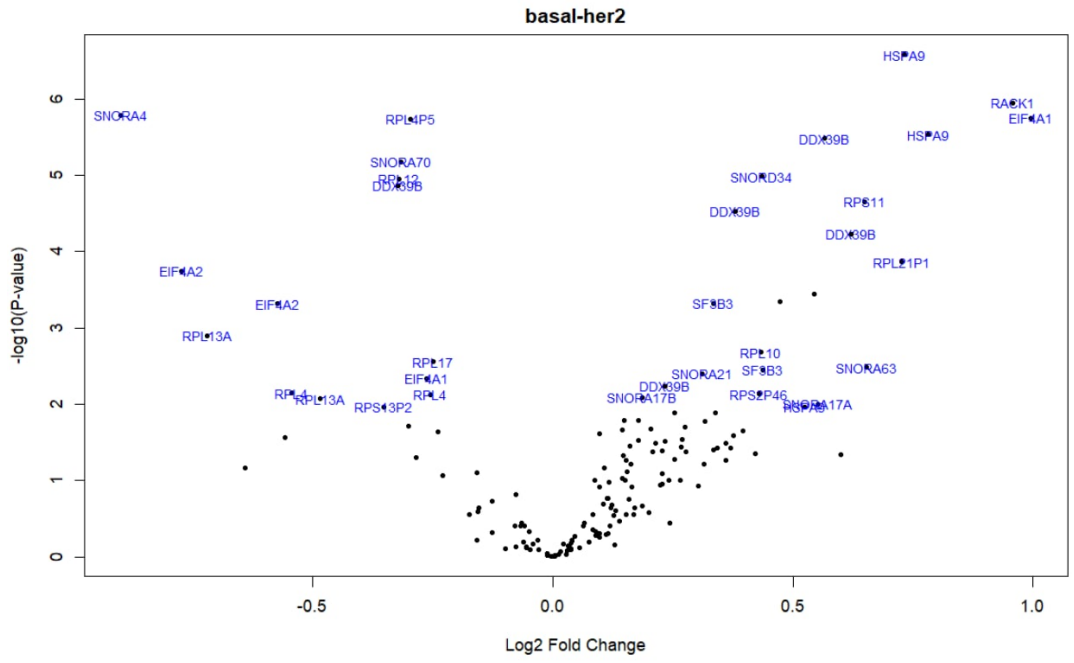

Figure 22. volcano plot Basal VS Her2

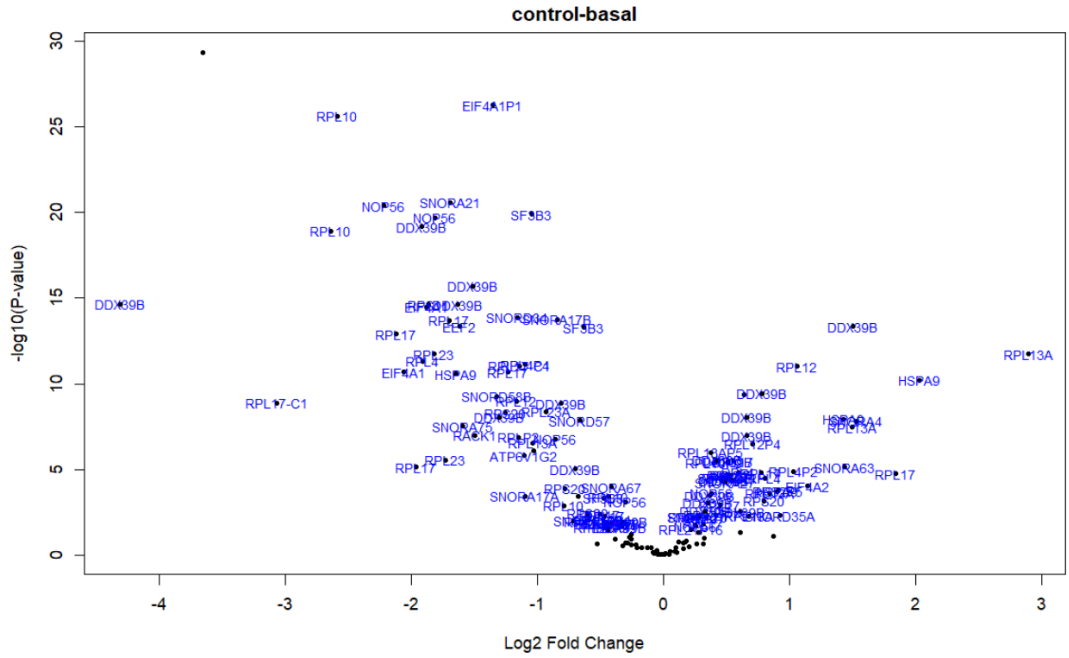

Figure 23. volcano plot Control VS Basal

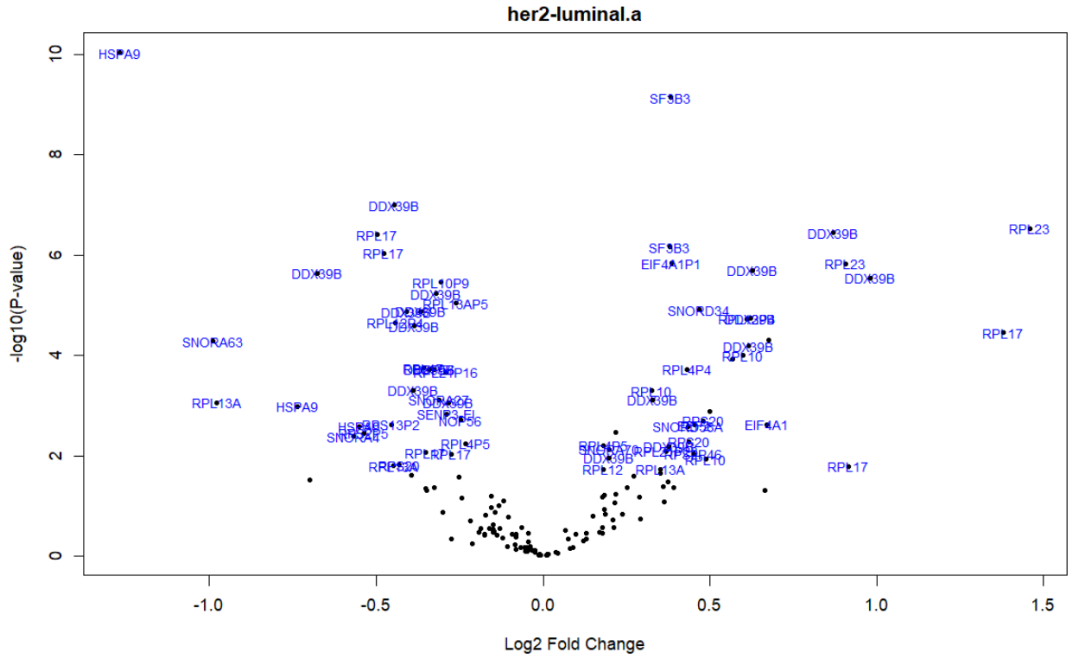

Figure 24. volcano plot HER2 VS Luminal A

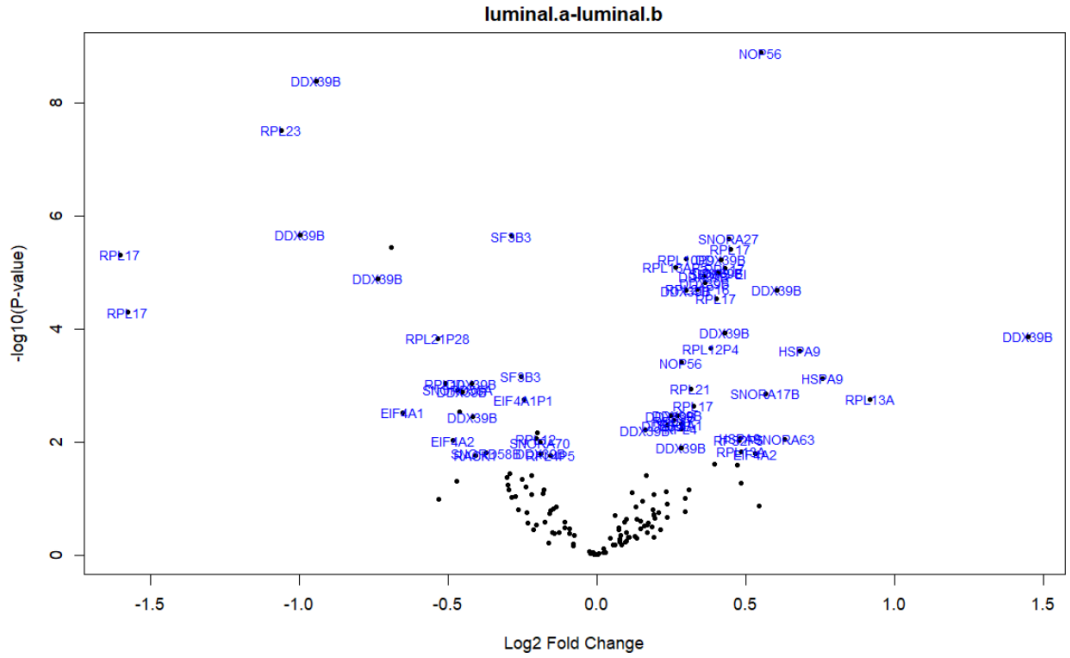

Figure 25. volcano plot Luminal A VS Luminal B

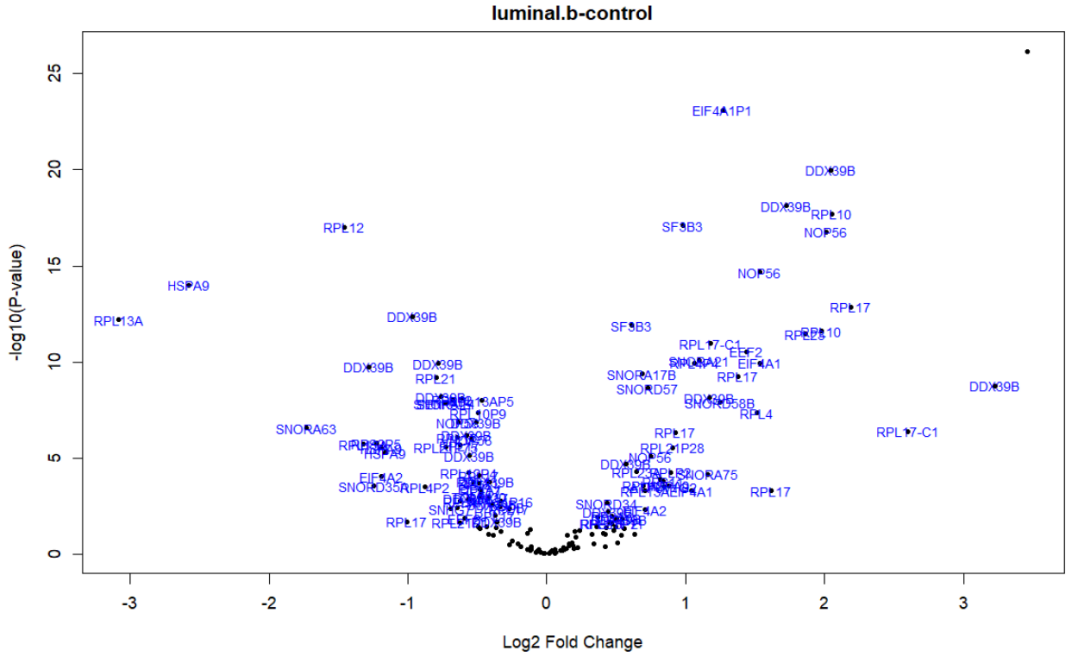

Figure 26. volcano plot Luminal B VS Control

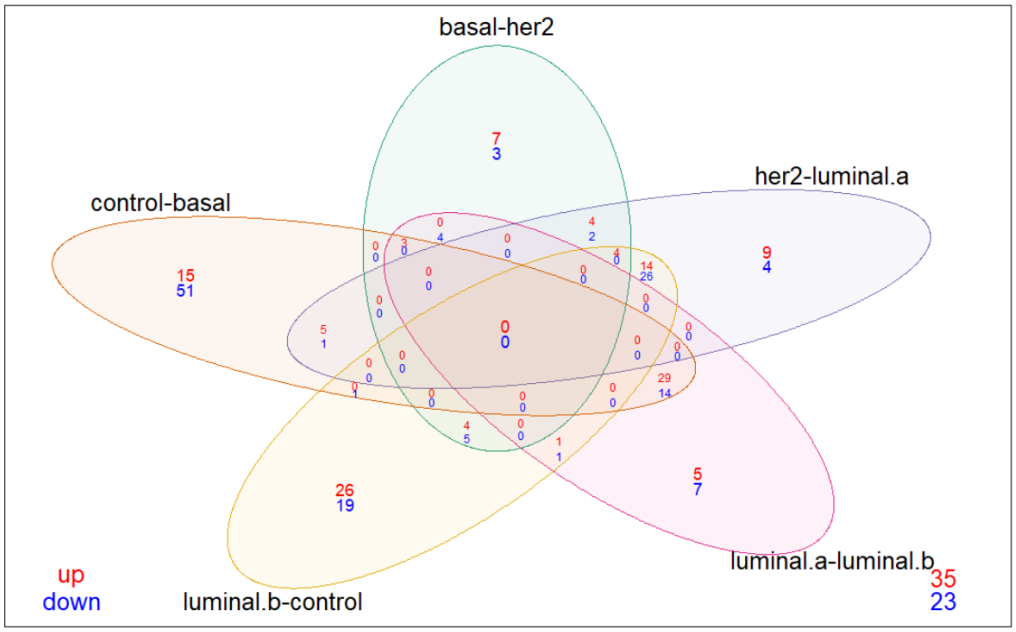

Figure 27. Venn diagram showing both upregulated and downregulated genes.

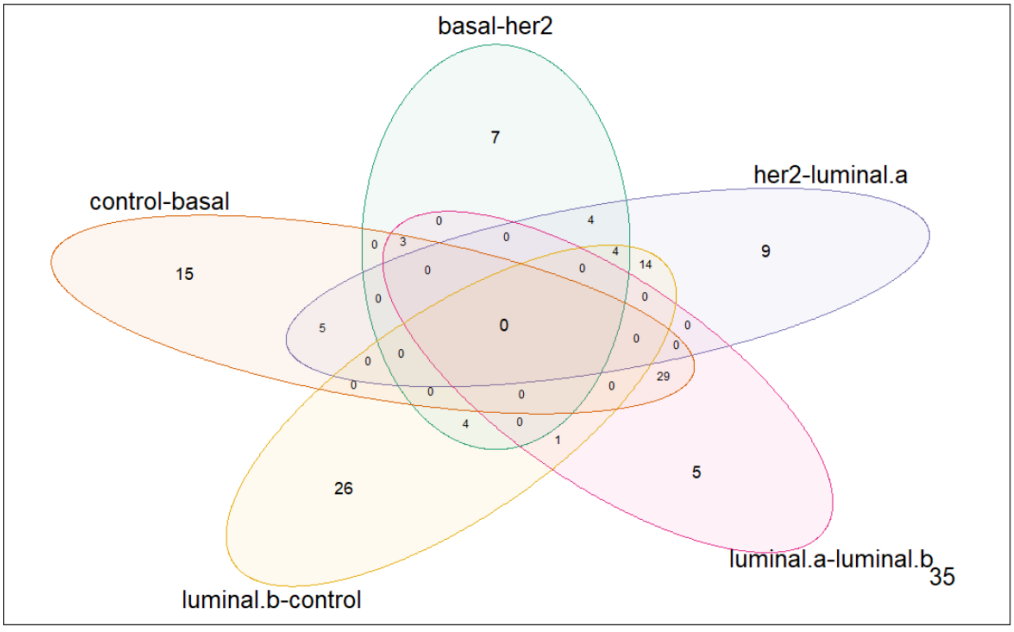

Figure 28. Venn diagram showing the upregulated genes.

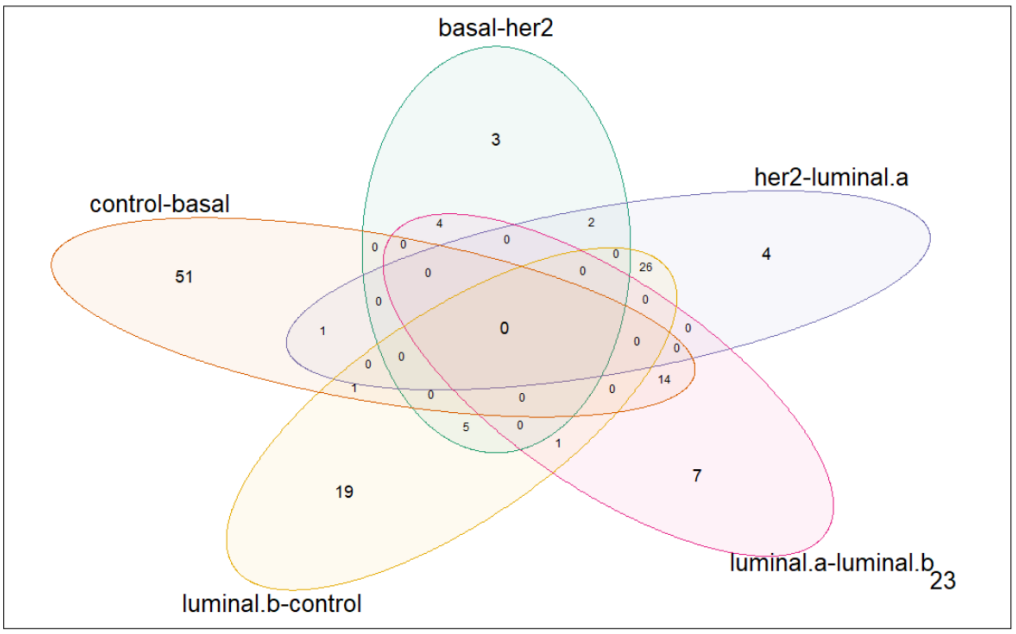

Figure 29. Venn diagram showing the downregulated genes.

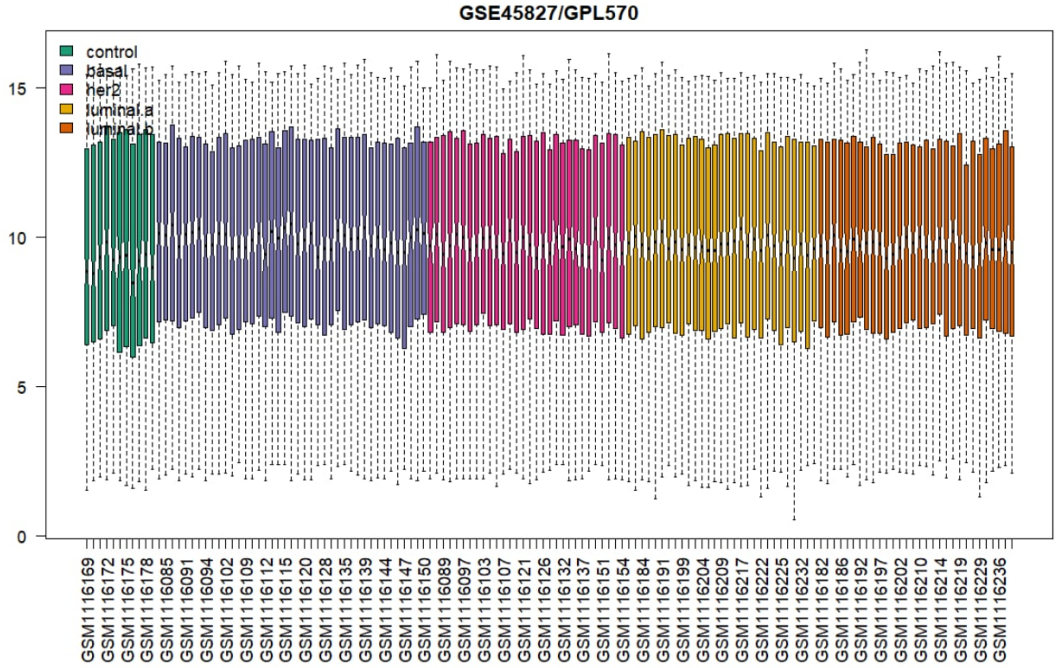

Figure 30. Boxplot with the intensity distributions of the individual arrays.