Supplementary Material: Robust gene coexpression networks using signed distance correlation

Javier Pardo-Diaz [1,3,*], Lyuba V. Bozhilova [1], Mariano Beguerisse-Díaz [2], Philip S. Poole [3], Charlotte M. Deane [1] and Gesine Reinert [1,†]
[1]Department of Statistics, University of Oxford, Oxford, UK
[2]Mathematical Institute, University of Oxford, Oxford, UK
[3]Department of Plant Sciences, University of Oxford, Oxford, UK
[*]jdiaz@stats.ox.ac.uk
[†]reinert@stats.ox.ac.uk

1 *Rhizobium leguminosarum* expression data

Table S1 provides the details of the gene expression data employed to generate the gene coexpression networks for *R. leguminosarum*. We used 54 different microarrays obtained under 18 conditions (first column). The second column indicates the growth condition of the bacteria; for example, "7 days pea, 1 day PI" stands for inoculation of seven days old pea plant sampled one day post-inoculation. The last column indicates the type of sample, which can be "Free-living" (bacteria grown in liquid culture), "Rhizosphere" (bacteria isolated from the fraction of soil in contact with the plant), or "Bacteroid" (bacteria that have infected the plant and differentiated). This data has been published previously [8, 6, 3] but never analysed jointly.

Microarrays (channels)	**Growth condition**	**Type**
kk21 (Cy3), kk41 (Cy5), kk45 (Cy3)	Succinate NH_4	Free-living
Jay10 (Cy5), Jay11 (Cy5), Jay12 (Cy3)	Glucose Glutamate	Free-living
Jay10 (Cy3), Jay11 (Cy3), Jay12 (Cy5)	Glucose Aspartate	Free-living
kk46 (Cy3), kk48 (Cy3), kk50 (Cy3)	Pyruvate NH_4	Free-living
Vinoy43 (Cy5), Vinoy44 (Cy5), Vinoy45 (Cy3)	Pyruvate NH_4 Hespertin	Free-living
Ade7 (Cy3), Ade9 (Cy3), Ade18 (Cy5)	Pyruvate NH_4 IAA	Free-living
Ade8 (Cy5), Ade11 (Cy3), Ade15 (Cy5)	Pyruvate NH_4 Kinetin	Free-living
kk39 (Cy5), kk43 (Cy5), kk51 (Cy3)	Inositol NH_4	Free-living
Vinoy26 (Cy5), Vinoy31 (Cy5), Vinoy34 (Cy3)	7 days pea, 1 day PI	Rhizosphere
Vinoy27 (Cy5), Vinoy32 (Cy5), Vinoy35 (Cy3)	7 days pea, 3 day PI	Rhizosphere
Vinoy28 (Cy5), Vinoy33 (Cy5), Vinoy36 (Cy3)	7 days pea, 7 day PI	Rhizosphere
Vinoy29 (Cy5), Vinoy37 (Cy5), Vinoy39 (Cy3)	14 days pea, 1 day PI	Rhizosphere
Vinoy30 (Cy5), Vinoy38 (Cy5), Vinoy40 (Cy3)	21 days pea, 1 day PI	Rhizosphere
kk67 (Cy5), kk71 (Cy3), kk72 (Cy5)	1 week pea, 7 days PI	Bacteroid
kk65 (Cy5), kk68 (Cy3), kk69 (Cy3)	1 week pea, 15 days PI	Bacteroid
kk73 (Cy5), kk74 (Cy3), kk75 (Cy5)	1 week pea, 21 days PI	Bacteroid
kk64 (Cy5), Vinoy22 (Cy3), Vinoy24 (Cy3)	1 week pea, 28 days PI	Bacteroid
kk70 (Cy5), kk60 (Cy3), kk61 (Cy5)	Vetch seed, 28 days PI	Bacteroid

Table 1: *Rhizobium leguminosarum* gene expression data description

2 Spearman correlation networks for *R. leguminosarum*

We construct a Spearman correlation matrix R from the pre-processed gene expression matrix M^* by computing the Spearman correlation between every pair of gene expression vectors. We threshold R to construct two networks $NR(d_S)$ and $NR(d_P)$ with edge density d_S and d_P, respectively. Similarly to the networks obtained using Pearson correlation $NP(d_S)$ and $NP(d_P)$, the ones based on Spearman correlation have a larger and less densely connected largest connected component than $NS(d_S)$ and $NS(d_P)$. The summaries of these two networks can be found in Table S2. We evaluate the Spearman networks using STRING as described in the paper. The results obtained are higher than the obtained by the Pearson correlation networks but lower than those using signed distance correlation. Table S3 and Figure S1 show the results of the STRING evaluation for the Spearman networks.

Network	Number of Edges	Number of vertices in LCC	Edge density LCC * 100	Global clustering coeff. LCC
$NR(d_S)$	313,348	6,664	1.41	0.515
$NR(d_P)$	406,977	6,831	1.34	0.515

Table 2: Summaries of Spearman networks for *R. leguminosarum* LCC Denotes largest connected component.

Network	All of STRING information (C)	Only coexpression information ($C^\dagger$)	All information except coexpression ($C^\ddagger$)
$NR(d_S)$	28.52	8.67	25.51
$NR(d_P)$	25.77	7.38	23.15

Table 3: Evaluation of the biological content of the Spearman networks with STRING

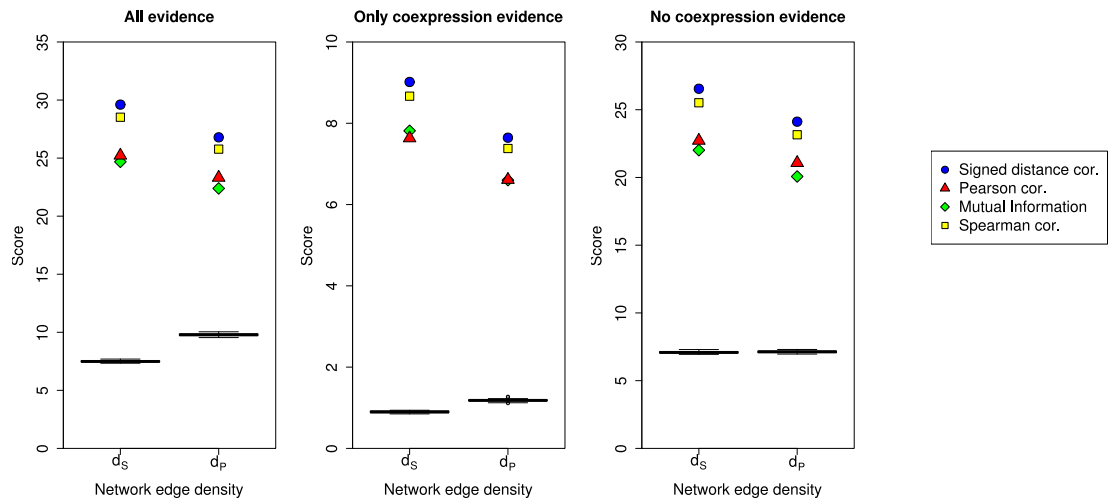

Figure 1: Scores obtained for the *R. leguminosarum* gene coexpression networks using STRING. All panels show the score for the different networks in the y-axis, and the network density on the x-axis. The scores are the result of adding up the confidence scores with all evidence (C), only coexpression evidence ($C^\dagger$) and everything excluding coexpression ($C^\ddagger$) from STRING associated with the edges in the networks, each computed using different of information. The black box plots correspond to the scores obtained by 30 random networks. Blue circles, red triangles and yellow squares represent signed distance correlation, Pearson correlation and Spearman correlation, respectively.

3 Gene coexpression network analysis for the study of Yeast RNA-Seq data

We use our signed correlation pipeline to generate a gene coexpression network from a dataset obtained using RNA-Seq of yeast (*Saccharomyces cerevisiae*) expressing pathways designed to increase ATP or GTP consumption. We obtain all the raw-counts for experiment E-MTAB-5174 in Expression Atlas [5] and remove the genes with zero expression variance. The final dataset which we feed into our pipeline includes the expression of 6,930 genes across 209 samples.

Following the pipeline described in the methodology section, we pre-process the data and obtain the distance correlation matrix D, the Pearson correlation matrix P, the signed correlation matrix S, and the matrix $|P|$ of absolute values of the Pearson correlation. Table S4 and Figure S2 show the summaries and distribution of these matrices.

Correlation matrix	Min	1st q	Median	3rd q	Max	Mean
D	0.00	0.26	0.40	0.57	1.00	0.42
$\|P\|$	0.00	0.13	0.30	0.51	1.00	0.34
P	-0.96	-0.29	0.00	0.31	1.00	0.01
S	-0.97	-0.39	-0.05	0.40	1.00	0.01

Table 4: Summaries for the correlation matrices of the Yeast dataset

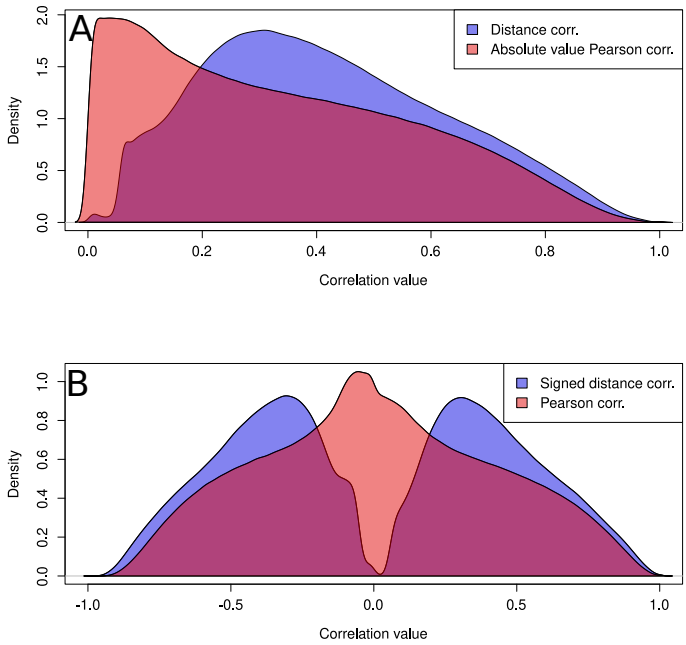

Figure 2: Density plots of the distribution of the values of the correlation matrices from the Yeast dataset. Panel A shows the distribution of values in the distance correlation matrix (blue) and of the absolute value of the values in the Pearson correlation matrix (red). Panel B shows the distribution of values in the signed distance correlation matrix (blue) and of the values in the Pearson correlation matrix (red).

We estimate the optimal threshold values θ^*, $\theta^\star$, and $\theta^\diamond$ to construct the unweighted gene coexpression networks $A_S(\theta^*)$, $A_P(\theta^\star)$, and and $A_I(\theta^\diamond)$ using COGENT [2]. We follow our pipeline to calculate the self-consistency of the networks. We adjust the similarity score by subtracting the network density. Fig S3 shows the variation of the score function for the correlation matrices across different edge densities. For the three matrices, there is a edge density value for which the score function reaches its maximum. This value is $d_S = 0.0131$ for the signed distance correlation (score of 0.746, which is achieved for $\theta^*= 0.84$), $d_P = 0.0146$ for the Pearson correlation (score of 0.733, which is achieved for $\theta^\star = 0.81$), and $d_I = 0.024$ for Mutual Information (score of 0.381, which is achieved for $\theta^\diamond = 1.04$. The results show that the use of signed distance correlation offers more stable networks. As in the study of *R. leguminosarum*, the scores obtained using signed mutual information are lower than those retrieved using the correlation methods. However, the difference between the Pearson correlation and signed distance correlation methods is lower than the observed in the study of *R. leguminosarum*. We analyse the networks $NS(d_S)$ (edge density d_S), $NS(d_P)$ (edge density d_P), and $NS(d_I)$ (edge density d_I) retrieved from S, the networks $NP(d_S)$ (edge density d_S), $NP(d_P)$ (edge density d_P) and $NP(d_I)$ (edge density d_I) retrieved from P, the networks $NI(d_S)$ (edge density d_S),$NI(d_P)$ (edge density d_P), and $NI(d_I)$ (edge density d_I) retrieved from I, and the networks $NR(d_S)$ (edge density d_S), $NR(d_P)$ (edge density d_P), and $NR(d_I)$ (edge density d_I) obtained using Spearman correlation. The summaries of the 12 networks are detailed in Table S5.

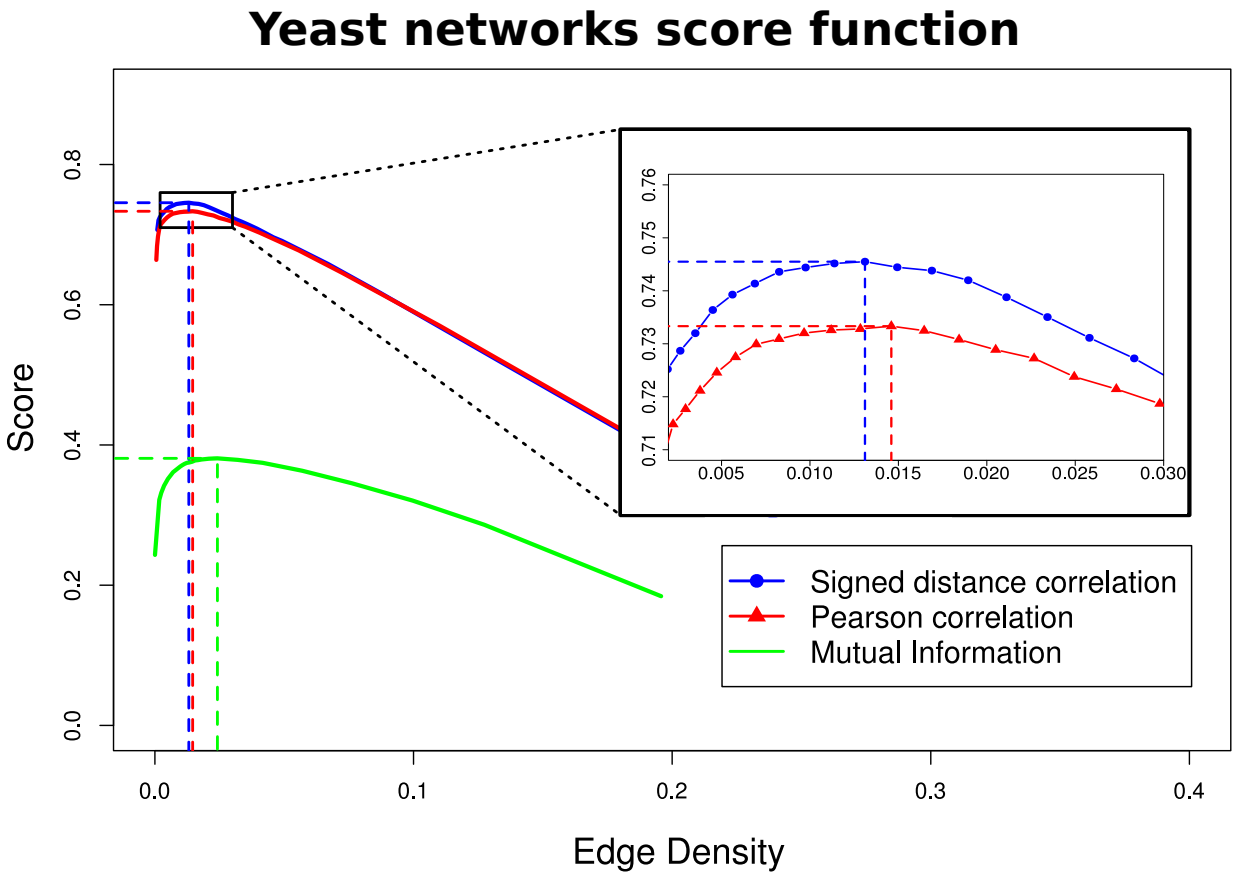

Figure 3: Score function values for different edge densities using the Yeast dataset. The blue line with circles shows the scores obtained using signed distance correlation and the red line with triangles those obtained using Pearson correlation. The green line those obtained using mutual information. The dotted lines indicate the position of the highest score point for each line. This value is 0.746 for the signed distance correlation network (giving edge density 0.0131 which is achieved with $\theta^* = 0.84$), 0.733 for the Pearson correlation network (giving edge density 0.0146 which is achieved with $\theta^\star = 0.81$) and 0.381 for the mutual information network (giving edge density 0.024 which is achieved with $\theta^\diamond = 1.04$.

We evaluate all the networks using STRING [9]. We use the three different sets of confidence scores:

Network	Number of Edges	Number of vertices in LCC	Edge density LCC * 100	Global clustering coeff. LCC
$NS(d_S)$	314,771	3,987	3.91	0.651
$NS(d_P)$	350,549	4,154	4.02	0.607
$NS(d_I)$	580,360	4,954	4.73	0.715
$NP(d_S)$	314,771	4,497	3.07	0.431
$NP(d_P)$	350,549	4,615	3.25	0.516
$NP(d_I)$	580,360	5,334	4.08	0.644
$NI(d_S)$	314,771	4,001	3.93	0.411
$NI(d_P)$	350,549	4,085	4.20	0.46
$NI(d_I)$	580,360	4,424	5.93	0.44
$NR(d_S)$	314,771	5,060	2.39	0.578
$NR(d_P)$	350,549	5,117	2.44	0.578
$NR(d_I)$	580,361	5,432	3.92	0.594

Table 5: Summaries of yeast networks. LCC denotes largest connected component.

obtained using all the evidence in STRING (C), obtained using only coexpression information ($C^{\dagger}$), and obtained using all the evidence except the coexpression information ($C^{\ddagger}$). Table S6 and Figure S4 present the results. For all the studied cases, the results obtained using the signed distance correlation networks are the highest. Unlike with the *R. leguminosarum* dataset, the Pearson correlation networks perform better than the Spearman correlation networks and the mutual information networks. In fact, the results for the signed distance correlation and Pearson correlation are almost identical (as in the case of the self-consistency study). These results reinforces our hypothesis that a high similarity in the self-consistency of the networks may imply a similarity in the amount of biological information that the networks are able to capture.

We generate 90 random networks of which 30 have edge density d_S, 30 have edge density d_P and other 30 have edge density d_I. We evaluate these networks with the STRING information and find that the all the networks in Table S5 outperform all of them. The highest difference to random is obtained for signed distance correlation networks when using only coexpression information ($C^{\dagger}$) to evaluate the networks; for $NS(d_S)$ the score is 17.69 times higher than the mean score obtained by random networks with matching densities.

Network	All of STRING information (C)	Only coexpression information ($C^{\dagger}$)	All information except coexpression ($C^{\ddagger}$)
$NS(d_S)$	**117.36**	**85.03**	**67.14**
$NP(d_S)$	116.31	84.85	66.3
$NI(d_S)$	68.31	45.70	41.67
$NR(d_S)$	74.84	48.11	50.48
RE d_S	14.02±0.14	4.81±0.08	11.89±0.12
$NS(d_P)$	**111.29**	**79.69**	**64.59**
$NP(d_P)$	110.14	79.37	63.64
$NI(d_P)$	66.36	44.15	40.72
$NR(d_P)$	72.45	46.05	49.19
RE d_P	14.02±0.14	4.79±0.07	11.91±0.13
$NS(d_I)$	**86.65**	**58.62**	**53.34**
$NP(d_I)$	85.56	57.94	52.60
$NI(d_I)$	58.30	37.37	36.93
$NR(d_I)$	62.03	37.67	43.10
RE d_I	13.98±0.08	4.78±0.04	11.87±0.08

Table 6: Evaluation of the biological content of the networks with STRING. RE indicates the expected (mean) result based on random networks with the indicated edge density and its standard deviation.

We conclude that the Yeast gene coexpression networks obtained using signed distance correlation are more stable and recover more biological information than those based on Pearson correlation, mutual information, or Spearman correlation.

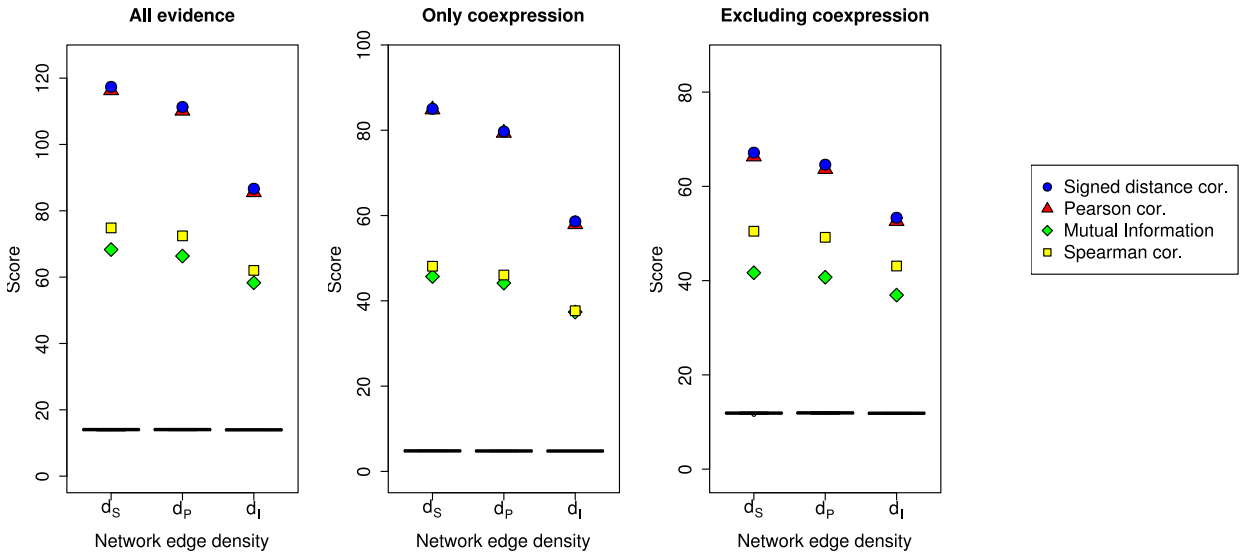

Figure 4: Scores obtained for the yeast gene coexpression networks using STRING. All panels show the score for the different networks in the y-axis, and the network density on the x-axis. The scores are the result of adding up the confidence scores from STRING associated with the edges in the networks. Each plot corresponds to a different set of values: using all evidence C, only coexpression evidence $C^{\dagger}$ and excluding coexpression evidence $C^{\ddagger}$. The black box plots correspond to the scores obtained by 30 random networks. Blue circles, red triangles, green diamonds, and yellow squares represent signed distance correlation, Pearson correlation, mutual information, and Spearman correlation, respectively.

4 Gene coexpression network analysis for the study of human liver single-cell RNA-Seq data

We use our signed correlation pipeline to generate a gene coexpression network from a dataset obtained using single-cell RNA-Seq of human liver cells [4]. Most of the methods to study single-cell RNA-Seq data aim to cluster cells according to their levels of gene expression [11, 10]. However, in this section we aim to group genes based on their expression across different samples (or cells, in this context) by constructing a gene coexpression network. Both approaches are complementary and we hope to combine them in the future. The original dataset measures the expression of 15,353 genes in 1,622 cells.

This dataset is considerably different from the previous two, both in the data itself – the measurements correspond to different cells instead of to different samples – and the organism – while both *R. leguminosarum* and *S. cerevisiae* are unicellular organisms, humans are not. Hence, a considerable proportion of genes are not expressed in the studied cells (for example, specific genes in neurons will not be expressed in cells from liver). For this reason, we employ a different pre-processing strategy in this case. In the first place, we quantile-normalise the data [1] to make the measurements in the different cells comparable. Afterwards, as in [7], we identify the "non-changing genes". These genes are those for which the difference between its highest and lowest expression value ("expression difference") is lower than the median of all the expression differences calculated for each gene, and in addition for which the mean expression signal between samples is lower than the median of all the expression signals calculated for each gene. After removing the "non-changing genes" we obtain an already quantile-normalised dataset with information for 8,585 genes. We do not apply more preprocessing steps to this dataset (we do not quantile-normalise again the data and we do not set the lowest expressed genes from each sample to the lowest expression value).

We calculate the distance correlation matrix (D), the Pearson correlation matrix (P), the signed correlation matrix (S), and the matrix $|P|$ of absolute values of the Pearson correlation. Table S7 and Figure S5 show the summaries and distribution of these matrices.

Correlation matrix	Min	1st q	Median	3rd q	Max	Mean
D	0.00	0.03	0.04	0.05	0.88	0.04
$\|P\|$	0.00	0.01	0.02	0.03	0.91	0.02
P	-0.63	-0.02	0.00	0.02	0.91	0.00
S	-0.71	-0.03	-0.02	0.04	0.88	0.00

Table 7: Summaries for the correlation matrices of the single-cell dataset from human liver

We estimate the optimal threshold values θ^*, $\theta^\star$, and $\theta^\diamond$ to construct the unweighted gene coexpression networks $A_S(\theta^*)$, $A_P(\theta^\star)$, and $A_I(\theta^\diamond)$ using COGENT [2]. We follow a similar pipeline as the one employed in the case of *R. leguminosarum* to calculate and adjust the self-consistency of the networks. The only difference is that as the input expression matrix is already quantile-normalised and low expressed genes have already been filtered out, in each of the 25 iterations the pre-processing steps are omitted. Figure S6 shows the variation of the score function for the correlation matrices across different edge densities. For the three tested methods, there is a edge density value for which the score function reaches its maximum. This value is $d_S = 0.00009$ for signed distance correlation (score of 0.896, which is achieved for $\theta^* = 0.44$), $d_P = 0.000089$ for Pearson correlation (score of 0.781, which is achieved for $\theta^\star = 0.43$), and $d_I = 0.00017$ for mutual information (score of 0.63, which is achieved for $\theta^\star = 0.63$. The results show that for all tested edge densities, the use of signed distance correlation offers more stable networks. We analyse the networks $NS(d_S)$ (edge density d_S), $NS(d_P)$ (edge density d_P), and $NS(d_I)$ (edge density d_I) retrieved from S, the networks $NP(d_S)$ (edge density d_S), $NP(d_P)$ (edge density d_P) and $NP(d_I)$ (edge density d_I) retrieved from P, the networks $NI(d_S)$ (edge density d_S),$NI(d_P)$ (edge density d_P), and $NI(d_I)$ (edge density d_I) retrieved from I, and the networks $NR(d_S)$ (edge density d_S), $NR(d_P)$ (edge density d_P), and $NR(d_I)$ (edge density d_I) obtained using Spearman correlation. The summaries of all the networks are detailed in Table S8. We note that the largest connected components of the networks contain a lower proportion of

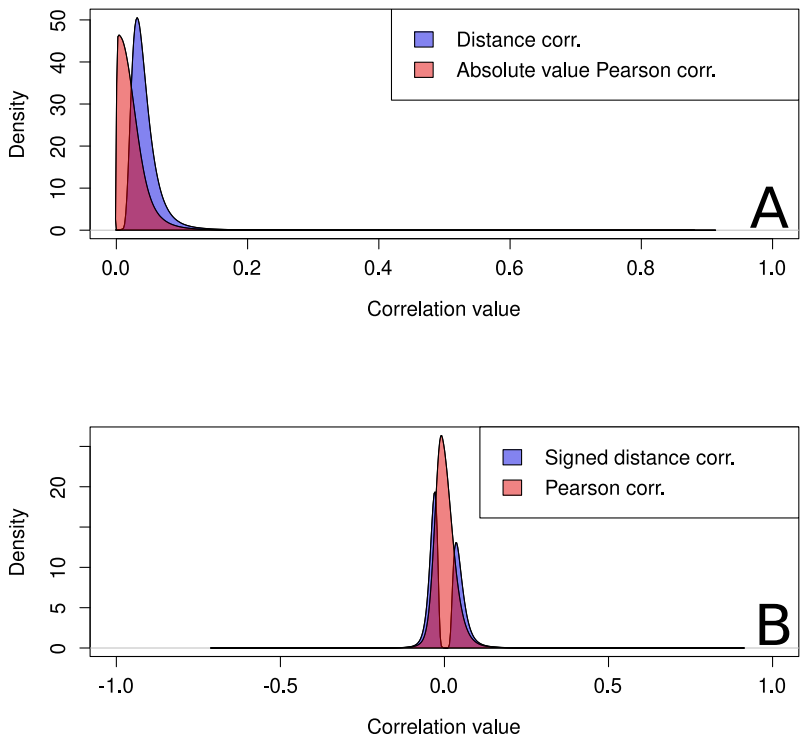

Figure 5: Density plots of the distribution of the values of the correlation matrices from the Yeast dataset. Panel A shows the distribution of values in the distance correlation matrix (blue) and of the absolute value of the values in the Pearson correlation matrix (red). Panel B shows the distribution of values in the signed distance correlation matrix (blue) and of the values in the Pearson correlation matrix (red).

the total vertices than in previous datasets. The networks obtained using signed distance correlation have a smaller and denser largest connected component than those obtained using Pearson correlation. In contrast to the two previous datasets, the largest connected components of the networks obtained using Spearman correlation and mutual information are smaller and denser than those from signed distance correlation and Pearson correlation networks; however, the signed distance correlation networks still have the highest global clustering coefficient. We observe that when increasing the edge density from d_S to d_I, the size of the largest connected component in the signed distance correlation networks increases considerably. This change is more subtle for the mutual information network. This finding may be connected with the fact that the optimal edge density for mutual information is higher than for signed distance correlation and Pearson correlation.

Similarly as for the two previous datasets, we evaluate the constructed networks using three sets of confidence values obtained from STRING. We also evaluate 60 random networks with edge densities d_S (30 networks), d_P (30 networks), and d_I (30 networks). Table S9 and Figure S7 show the results. Overall, the networks obtained using signed distance correlation $NS(d_S)$ and $NS(d_P)$ retrieve a higher score than their competitors based on either Pearson or Spearman correlation or mutual information. This result suggest that among the assessed methods, signed distance correlation captures the broadest range of biological information, which makes it a tool of choice to generating gene coexpression networks from single-cell RNA-Seq data. All the analysed networks obtain a higher score than the random ones. The highest difference to random is obtained for signed distance correlation networks when using only coexpression information ($C^{\dagger}$) to evaluate the networks; for $NS(d_P)$ the score is 92.92 times higher than the mean score obtained by random networks with matching densities; for $NS(d_S)$ it is even 94.28 than the corresponding mean score for the random networks.

We conclude that the liver gene coexpression networks obtained using signed distance correlation are more stable and show recover more biological information than those based on Pearson correlation and that our signed distance correlation method is suitable for studying single cell gene expression

Network	Number of Edges	Number of vertices in LCC	Edge density LCC * 100	Global clustering coeff. LCC
$NS(d_S)$	3320	212	14.61	0.87
$NS(d_P)$	3279	210	14.71	0.87
$NS(d_I)$	6150	398	7.69	0.83
$NP(d_S)$	3320	219	12.42	0.83
$NP(d_P)$	3279	216	12.62	0.83
$NP(d_I)$	6150	455	5.13	0.81
$NI(d_S)$	3320	204	16.00	0.80
$NI(d_P)$	3279	198	16.77	0.81
$NI(d_I)$	6150	279	15.84	0.77
$NR(d_S)$	3320	161	25.54	0.81
$NR(d_P)$	3279	160	25.54	0.81
$NR(d_I)$	6150	273	16.45	0.78

Table 8: Summaries of single cell human liver networks.
LCC denotes largest connected component.

Network	All of STRING information (C)	Only coexpression information ($C^{\dagger}$)	All information except coexpression ($C^{\ddagger}$)
$NS(d_S)$	**212.02**	**185.35**	**203.55**
$NP(d_S)$	196.30	169.84	188.12
$NI(d_S)$	189.05	172.79	184.31
$NR(d_S)$	183.07	173.02	179.42
RE d_S	6.75 ± 0.73	1.97 ± 0.3	5.90 ± 0.73
$NS(d_P)$	**212.35**	**185.80**	**204.00**
$NP(d_P)$	196.93	170.86	188.85
$NI(d_P)$	187.68	171.82	182.96
$NR(d_P)$	184.18	174.13	180.61
RE d_P	6.82 ± 0.82	2.00 ± 0.32	5.98 ± 0.78
$NS(d_I)$	**176.21**	**144.00**	**162.72**
$NP(d_I)$	162.60	132.43	152.03
$NI(d_I)$	140.49	121.55	131.09
$NR(d_I)$	146.91	131.03	136.13
RE d_I	7.03 ± 0.91	2.06 ± 0.25	6.17 ± 0.86

Table 9: Evaluation of the biological content of the networks with STRING. RE indicates the expected (mean) result based on random networks with
the indicated edge density and its standard deviation.

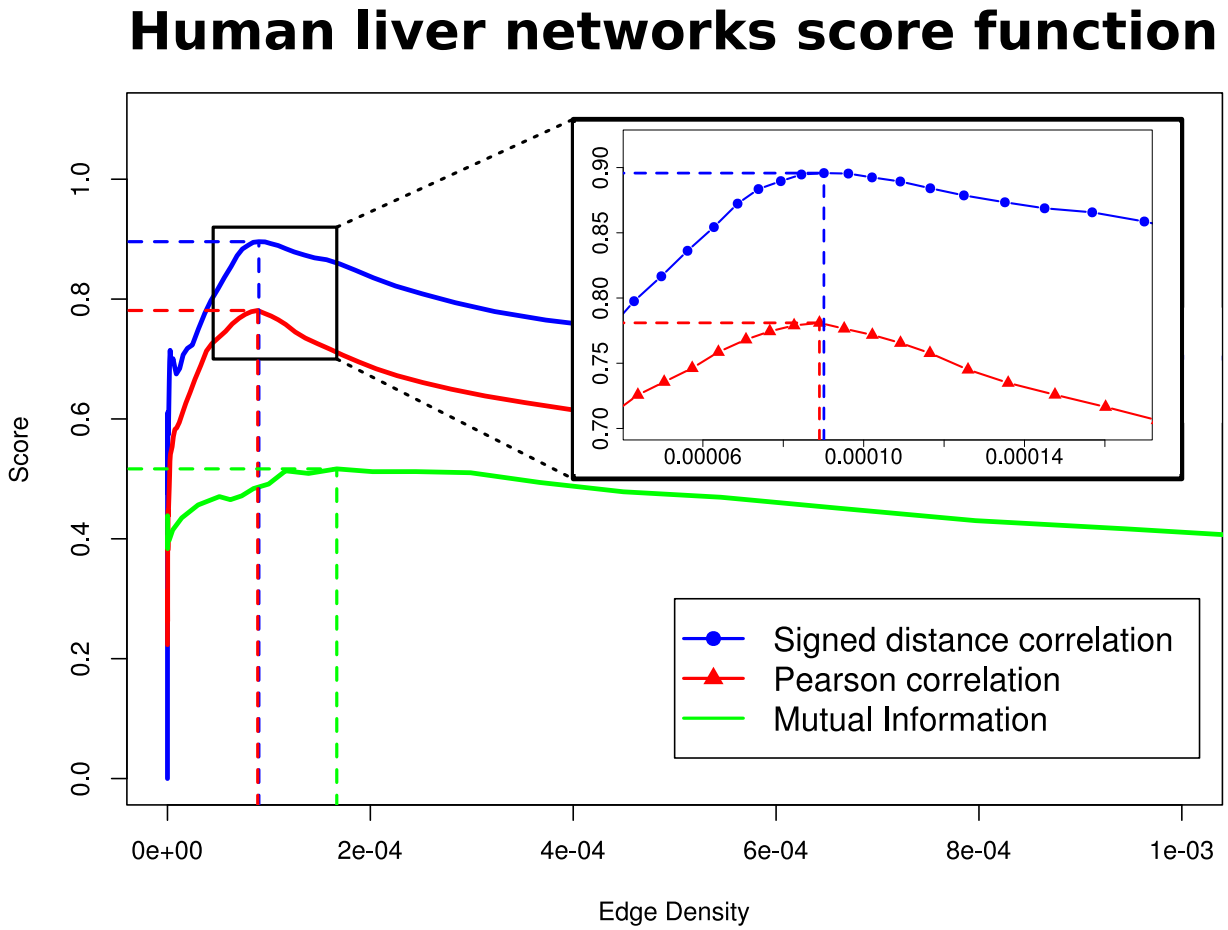

Figure 6: Plot of the score function values for different edge densities using the human liver single-cell RNA-Seq dataset. The blue line shows the scores obtained using signed distance correlation, the red line those obtained using Pearson correlation, and the green line those obtained using mutual information. The dotted lines indicate the position of the highest score point for each line. This value is 0.896 for the signed distance correlation network (giving edge density $9.01e^{-05}$ which is achieved with $\theta^* = 0.44$), 0.781 for the Pearson correlation network (giving edge density $8.90e^{-05}$ which is achieved with $\theta^* = 0.43$) and 0.517 for the mutual information network (giving edge density $1.67e^{-0.4}$.

data.

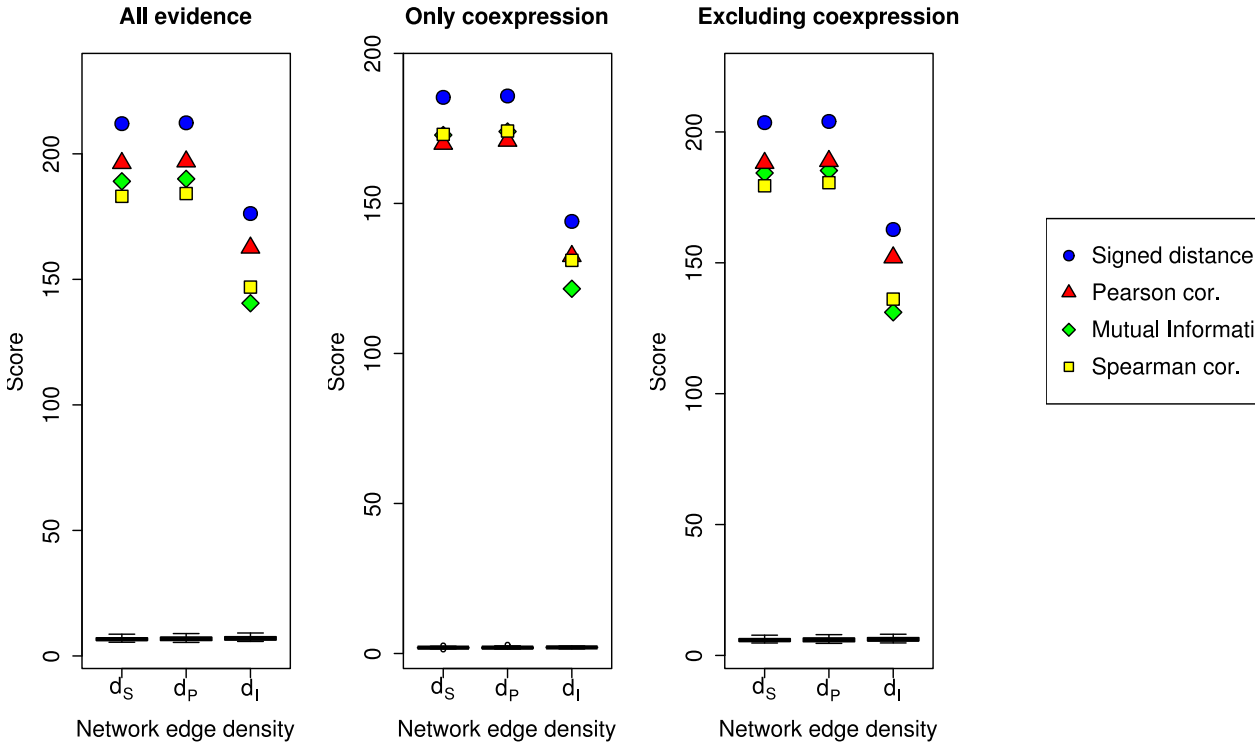

Figure 7: Scores obtained for the single-cell human liver gene coexpression networks using STRING. All panels show the score for the different networks in the y-axis, and the network density on the x-axis. The scores are the result of adding up the confidence scores from STRING associated with the edges in the networks. Each plot corresponds to a different set of values: using all evidence C, only coexpression evidence $C^{\dagger}$ and excluding coexpression evidence $C^{\ddagger}$. The black box plots correspond to the scores obtained by 30 random networks. Blue circles, red triangles and yellow squares represent signed distance correlation, Pearson correlation and Spearman correlation, respectively.

5 Network evaluation using information from STRING

We use the interactions between proteins reported in STRING to evaluate the amount of biological information that the constructed networks capture. The association evidence in STRING is categorized into independent channels, weighted, and integrated, resulting in a confidence score for all recorded protein interactions. For each organism, we use three different sets of interactions and confidence scores: the overall confidence score C provided by STRING, the $C^{\dagger}$ obtained attending only to coexpression information, and the $C^{\ddagger}$ obtained excluding all coexpression information.

The recomputing of the each of the scores was done using the python script located on the STRING webpage (Figure 8), which was accessed on the 20$^{\text{th}}$ of April 2020 using the URL `https://string-db.org/cgi/help.pl?&subpage=faq%23how-are-the-scores-computed`. The script can be found at the end of this section. We commented out lines 96–98, 100–102 in the script to compute the $C^{\dagger}$ values; and the line 99 in the script to obtain the $C^{\ddagger}$ values.

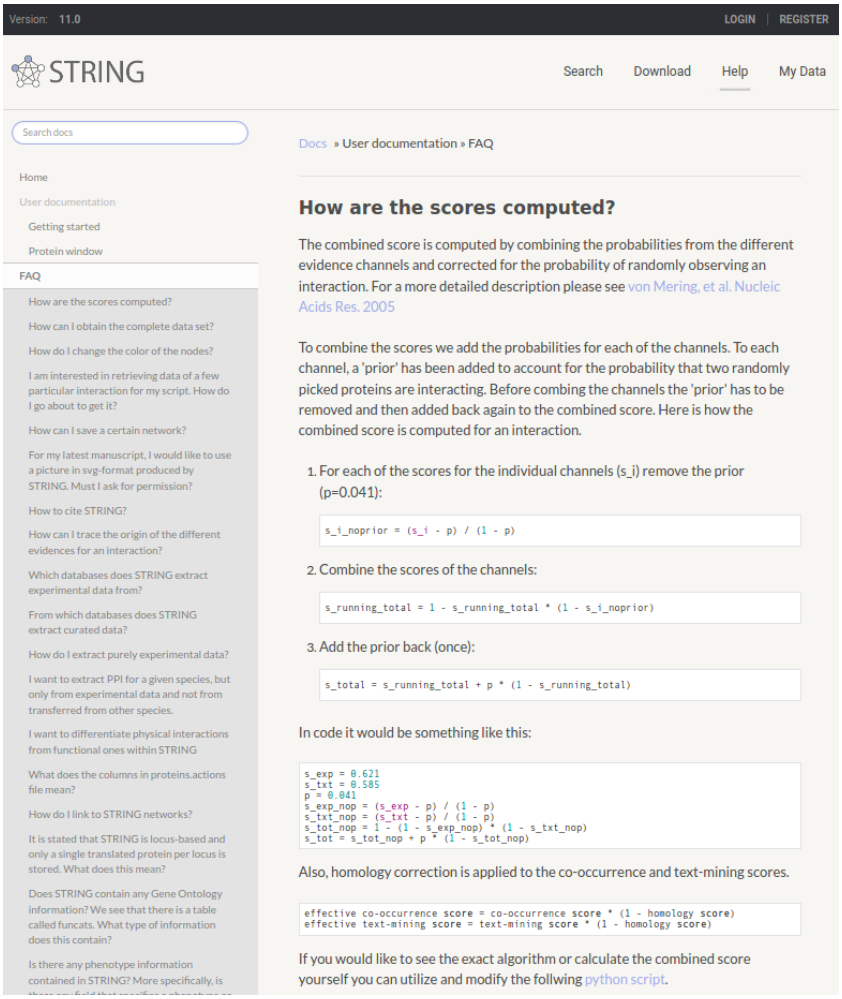

Figure 8: STRING FAQ webpage. Accessed on the 20$^{\text{th}}$ of April 2020. The content describes how the STRING combines the scores of the different channels. At the end of the section there is the link to the script which was modified and used in this work.

```
from __future__ import print_function
import os
import sys

############################################################
## This script combines all the STRING's channels subscores
## into the final combined STRING score.
## It uses unpacked protein.links.full.xx.txt.gz as input
## which can be downloaded from the download subpage:
##      https://string-db.org/cgi/download.pl
############################################################

input_file = "9606.protein.links.full.v10.5.txt"

if not os.path.exists(input_file):
    sys.exit("Can't locate input file %s" % input_file)

prior = 0.041

def compute_prior_away(score, prior):

    if score < prior: score = prior
    score_no_prior = (score - prior) / (1 - prior)

    return score_no_prior

header = True
for line in open(input_file):

    if header:
        header = False
        continue

    l = line.split()

    ## load the line

    (protein1, protein2,
     neighborhood, neighborhood_transferred,
     fusion, cooccurrence,
     homology,
     coexpression, coexpression_transferred,
     experiments, experiments_transferred,
     database, database_transferred,
     textmining, textmining_transferred,
     initial_combined) = l

    ## divide by 1000

    neighborhood = float(neighborhood) / 1000
    neighborhood_transferred = float(neighborhood_transferred) / 1000
    fusion = float(fusion) / 1000
    cooccurrence =  float(cooccurrence) / 1000
    homology = float(homology) / 1000
    coexpression = float(coexpression) / 1000
    coexpression_transferred = float(coexpression_transferred) / 1000
    experiments = float(experiments) / 1000
    experiments_transferred = float(experiments_transferred) / 1000
    database = float(database) / 1000
    database_transferred = float(database_transferred) / 1000
    textmining = float(textmining) / 1000
    textmining_transferred = float(textmining_transferred) / 1000
    initial_combined = int(initial_combined)

    ## compute prior away

```

```
    neighborhood_prior_corrected             = compute_prior_away(neighborhood, prior)
    neighborhood_transferred_prior_corrected = compute_prior_away(neighborhood_transferred, prior)
    fusion_prior_corrected                   = compute_prior_away(fusion, prior)
    cooccurrence_prior_corrected             = compute_prior_away(cooccurrence, prior)
    coexpression_prior_corrected             = compute_prior_away(coexpression, prior)
    coexpression_transferred_prior_corrected = compute_prior_away(coexpression_transferred, prior)
    experiments_prior_corrected              = compute_prior_away(experiments, prior)
    experiments_transferred_prior_corrected  = compute_prior_away(experiments_transferred, prior)
    database_prior_corrected                 = compute_prior_away(database, prior)
    database_transferred_prior_corrected     = compute_prior_away(database_transferred, prior)
    textmining_prior_corrected               = compute_prior_away(textmining, prior)
    textmining_transferred_prior_corrected   = compute_prior_away(textmining_transferred, prior)

    ## then, combine the direct and transferred scores for each category:

    neighborhood_both_prior_corrected = 1.0 - (1.0 - neighborhood_prior_corrected) * (1.0 - neighborhood_transferred_prior_corrected)
    coexpression_both_prior_corrected = 1.0 - (1.0 - coexpression_prior_corrected) * (1.0 - coexpression_transferred_prior_corrected)
    experiments_both_prior_corrected  = 1.0 - (1.0 - experiments_prior_corrected) * (1.0 - experiments_transferred_prior_corrected)
    database_both_prior_corrected     = 1.0 - (1.0 - database_prior_corrected) * (1.0 - database_transferred_prior_corrected)
    textmining_both_prior_corrected   = 1.0 - (1.0 - textmining_prior_corrected) * (1.0 - textmining_transferred_prior_corrected)

    ## now, do the homology correction on cooccurrence and textmining:

    cooccurrence_prior_homology_corrected = cooccurrence_prior_corrected * (1.0 - homology)
    textmining_both_prior_homology_corrected = textmining_both_prior_corrected * (1.0 - homology)

    ## next, do the 1 - multiplication:

    combined_score_one_minus = (
        (1.0 - neighborhood_both_prior_corrected) *
        (1.0 - fusion_prior_corrected) *
        (1.0 - cooccurrence_prior_homology_corrected) *
        (1.0 - coexpression_both_prior_corrected) *
        (1.0 - experiments_both_prior_corrected) *
        (1.0 - database_both_prior_corrected) *
        (1.0 - textmining_both_prior_homology_corrected) *
        1)

    ## and lastly, do the 1 - conversion again, and put back the prior *exactly once*

    combined_score = (1.0 - combined_score_one_minus)            ## 1- conversion
    combined_score *= (1.0 - prior)                              ## scale down
    combined_score += prior                                      ## and add prior.

    ## round

    combined_score = int(combined_score * 1000)
    print(protein1, protein2, combined_score)
```

Listing 1: Combine subscores script

References

[1] Benjamin M Bolstad, Rafael A Irizarry, Magnus Åstrand, and Terence P. Speed. A comparison of normalization methods for high density oligonucleotide array data based on variance and bias. *Bioinformatics*, 19(2):185–193, 2003.

[2] Lyuba V Bozhilova, Javier Pardo-Diaz, Gesine Reinert, and Charlotte M Deane. COGENT: evaluating the consistency of gene co-expression networks. *bioRxiv*, 2020.

[3] R Karunakaran, V K Ramachandran, J C Seaman, A K East, B Mouhsine, T H Mauchline, J Prell, A Skeffington, and P S Poole. Transcriptomic analysis of Rhizobium leguminosarum biovar viciae in symbiosis with host plants Pisum sativum and Vicia cracca. *Journal of Bacteriology*, 191(12):4002–4014, 2009.

[4] Florian Klimm, Enrique M Toledo, Thomas Monfeuga, Fang Zhang, Charlotte M Deane, and Gesine Reinert. Functional module detection through integration of single-cell rna sequencing data with protein–protein interaction networks. *bioRxiv*, page 698647, 2019.

[5] Robert Petryszak, Maria Keays, Y Amy Tang, Nuno A Fonseca, Elisabet Barrera, Tony Burdett, Anja Füllgrabe, Alfonso Munoz-Pomer Fuentes, Simon Jupp, Satu Koskinen, et al. Expression atlas update—an integrated database of gene and protein expression in humans, animals and plants. *Nucleic Acids Research*, 44(D1):D746–D752, 2016.

[6] Francesco Pini, Alison K East, Corinne Appia-Ayme, Jakub Tomek, Ramakrishnan Karunakaran, Marcela Mendoza-Suarez, Anne Edwards, Jason Terpolilli, Joshua Rowoth, J Allan Downie, et al. Lux bacterial biosensors for in vivo spatiotemporal mapping of root secretion. *Plant Physiology*, pages pp–01302, 2017.

[7] Carlos Prieto, Alberto Risueño, Celia Fontanillo, and Javier De Las Rivas. Human gene coexpression landscape: confident network derived from tissue transcriptomic profiles. *PloS One*, 3(12):e3911, 2008.

[8] Vinoy K Ramachandran, Alison K East, Ramakrishnan Karunakaran, J Allan Downie, and Philip S Poole. Adaptation of Rhizobium leguminosarum to pea, alfalfa and sugar beet rhizospheres investigated by comparative transcriptomics. *Genome Biology*, 12(10):R106, 2011.

[9] Damian Szklarczyk, Annika L Gable, David Lyon, Alexander Junge, Stefan Wyder, Jaime Huerta-Cepas, Milan Simonovic, Nadezhda T Doncheva, John H Morris, Peer Bork, et al. STRING v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic Acids Research*, 47(D1):D607–D613, 2019.

[10] Daniel E Wagner, Caleb Weinreb, Zach M Collins, James A Briggs, Sean G Megason, and Allon M Klein. Single-cell mapping of gene expression landscapes and lineage in the zebrafish embryo. *Science*, 360(6392):981–987, 2018.

[11] Caleb Weinreb, Samuel Wolock, and Allon M Klein. Spring: a kinetic interface for visualizing high dimensional single-cell expression data. *Bioinformatics*, 34(7):1246–1248, 2018.