Introduction to Bio-Informatics 2022

Assignment 5: Gene expression data

Chiara Paglioni (i6249782)

Some toxicology researchers want to know the effect of Cyclosporin A (CSA) on human cells. They take two doses of this compound (20uM and 3uM) and take samples after 12, 24, 48 and 72 hours of exposure. They also take samples from cells exposed to just the solvent which the CSA is normally dissolved into, DMSO (eg the samples labelled DMSO are untreated and the samples labelled CSA are treated). They want to know which genes are affected by the CSA.

- After preprocessing pick at least one dose/timepoint combination and find the differentially expressed genes.

The data for this task is in the file raw_data_labelled.csv the rows are labelled with the affymetrix id of the probe - there will be many probes measuring each gene.

Read data

```
clear;
data = readtable('raw_data_labelled.csv')
```

```
data = 54675×37 table
```

...

	Var1	DMSO_12h_1	CsA_20uM_12h_1	CsA_3uM_12h_1	DMSO_24h_1
1	'1007_s_at'	1.1725e+03	708.5000	795.5000	1051
2	'1053_at'	779	253.5000	459	520
3	'117_at'	170	110.5000	89	150
4	'121_at'	747.5000	317	476	644
5	'1255_g_at'	106	61.5000	54	89
6	'1294_at'	267.5000	149	164	263.5000
7	'1316_at'	280	124	150.5000	230
8	'1320_at'	165	129.5000	102.5000	129
9	'1405_i_at'	105.5000	67	56.5000	95.5000
10	'1431_at'	97	60	74.5000	81
11	'1438_at'	198	138	97.5000	169.5000
12	'1487_at'	474.5000	328.5000	349	493
13	'1494_f_at'	203.5000	120	114.5000	174.5000
14	'1552256_a_at'	2365	1031	1797	2082

	Var1	DMSO_12h_1	CsA_20uM_12h_1	CsA_3uM_12h_1	DMSO_24h_1
15	'1552257_a_at'	870	424	577	660
16	'1552258_at'	219	127	105	149
17	'1552261_at'	144	79	73	144
18	'1552263_at'	292	147	153	298
19	'1552264_a_at'	696	453	357	825
20	'1552266_at'	96	55	55	81
21	'1552269_at'	113	67	74	95
22	'1552271_at'	207	118	123	167
23	'1552272_a_at'	189	111	126	131
24	'1552274_at'	293	221	195	231
25	'1552275_s_at'	206	159	129	191
26	'1552276_a_at'	213	138	120	207
27	'1552277_a_at'	702	815	595	625
28	'1552278_a_at'	215	106	131	160
29	'1552279_a_at'	446	196	249	318
30	'1552280_at'	135	65	67	97
31	'1552281_at'	784	456	665	623
32	'1552283_s_at'	130	79	78	100
33	'1552286_at'	198	104	118	150
34	'1552287_s_at'	324	235	267	280
35	'1552288_at'	111	78	68	87
36	'1552289_a_at'	128	83	80	124
37	'1552291_at'	431	134	314	317
38	'1552293_at'	133	76	96	112
39	'1552295_a_at'	373	177	234	273
40	'1552296_at'	257	123	162	220
41	'1552299_at'	229	96	143	205
42	'1552301_a_at'	215	89	97	141
43	'1552302_at'	270	92	192	296
44	'1552303_a_at'	336	132	225	321
45	'1552304_at'	229	79	148	173
46	'1552306_at'	379	121	225	253
47	'1552307_a_at'	605	151	389	483

	Var1	DMSO_12h_1	CsA_20uM_12h_1	CsA_3uM_12h_1	DMSO_24h_1
48	'1552309_a_at'	179	79	143	115
49	'1552310_at'	435	158	359	366
50	'1552311_a_at'	305	198	186	294
51	'1552312_a_at'	377	255	237	389
52	'1552314_a_at'	191	133	129	178
53	'1552315_at'	131	86	76	105
54	'1552316_a_at'	95	53	60	73
55	'1552318_at'	106	58	71	95
56	'1552319_a_at'	177	145	145	160
57	'1552320_a_at'	113	67	65	129
58	'1552321_a_at'	104	68	57	85
59	'1552322_at'	87	52	55	71
60	'1552323_s_at'	124	68	89	106
61	'1552325_at'	87	52	47	61
62	'1552326_a_at'	101	65	65	93
63	'1552327_at'	87	48	55	71
64	'1552329_at'	763	518	459	926
65	'1552330_at'	476	298	347	361
66	'1552332_at'	324	165	176	257
67	'1552334_at'	178	97	130	142
68	'1552335_at'	239	134	136	278
69	'1552337_s_at'	113	71	67	89
70	'1552338_at'	128	87	87	99
71	'1552340_at'	136	97	82	103
72	'1552343_s_at'	183	99	104	150
73	'1552344_s_at'	728	319	418	625
74	'1552347_at'	285	118	178	261
75	'1552348_at'	166	102	98	135
76	'1552349_a_at'	166	82	80	119
77	'1552354_at'	183	106	92	121
78	'1552355_s_at'	191	124	103	172
79	'1552359_at'	83	49	49	64
80	'1552360_a_at'	216	120	147	187

	Var1	DMSO_12h_1	CsA_20uM_12h_1	CsA_3uM_12h_1	DMSO_24h_1
81	'1552362_a_at'	902	211	629	1643
82	'1552364_s_at'	554	379	275	542
83	'1552365_at'	73	49	46	70
84	'1552367_a_at'	89	62	54	87
85	'1552368_at'	88	59	53	67
86	'1552370_at'	183	75	114	126
87	'1552372_at'	71	49	47	65
88	'1552373_s_at'	81	54	52	65
89	'1552375_at'	127	95	75	88
90	'1552377_s_at'	324	214	197	254
91	'1552378_s_at'	362	178	209	304
92	'1552379_at'	79	50	49	70
93	'1552381_at'	130	113	109	122
94	'1552383_at'	245	113	139	206
95	'1552384_a_at'	189	115	93	165
96	'1552386_at'	82	54	55	64
97	'1552388_at'	232	118	134	199
98	'1552389_at'	110	59	61	74
99	'1552390_a_at'	98	56	63	76
100	'1552391_at'	179	114	85	139

⋮

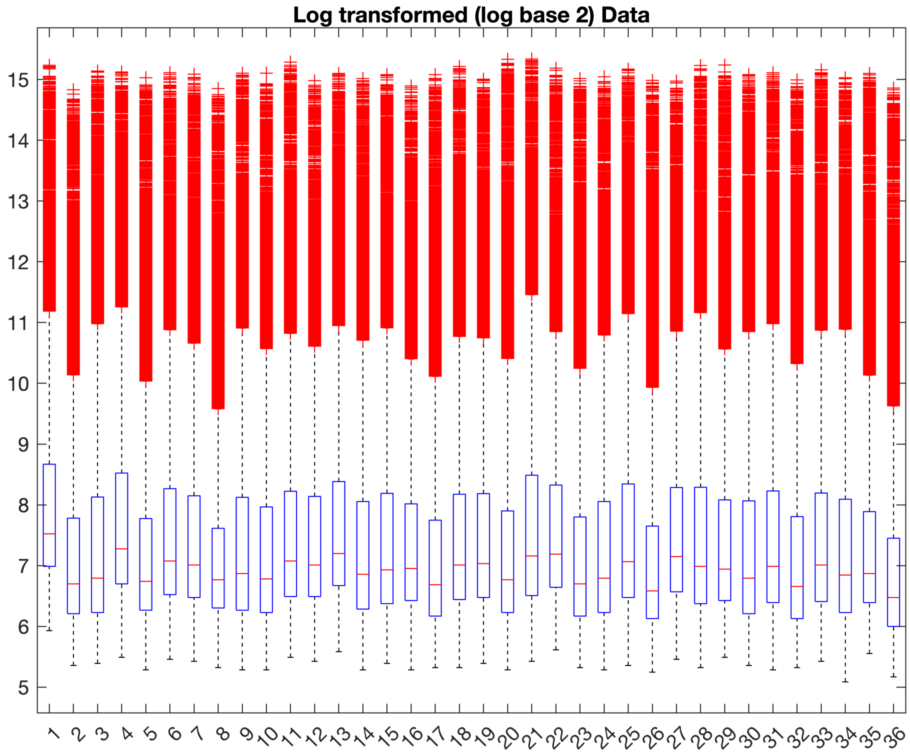
Plot data and Preprocessing

First of all, the given values are log transformed. To do so, the log base 2 of each value of the previous table is computed and saved into a new array: log_data.

```
% Log transform (log base 2) the data
log_data = table2array(data(1:54675, 2:37));
for i = 1 : length(log_data(:,1))
    for j = 1 : length(log_data(1,:))
        log_data(i, j) = log2(log_data(i, j));
    end
end
log_data
```

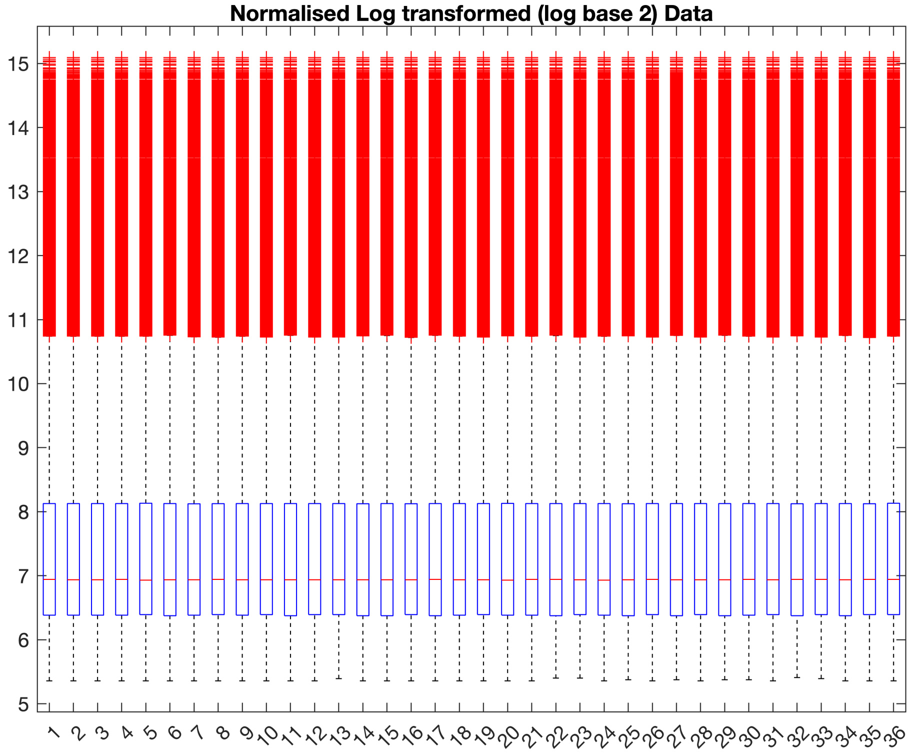
```
log_data = 54675×36
   10.1954    9.4686    9.6357   10.0375    9.9174    9.7013    9.2503    9.7288 ···
    9.6055    7.9858    8.8424    9.0224    8.0417    8.6202    7.6900    7.5196
    7.4094    6.7879    6.4757    7.2288    7.6795    6.8704    6.8765    7.6724
    9.5459    8.3083    8.8948    9.3309    8.3061    9.0444    8.9099    8.1749
    6.7279    5.9425    5.7549    6.4757    6.0980    6.1599    6.0553    6.0980
    8.0634    7.2192    7.3576    8.0417    7.0661    7.6830    7.3083    7.0875
    8.1293    6.9542    7.2336    7.8455    7.2432    7.5118    7.7142    7.1497
    7.3663    7.0168    6.6795    7.0112    7.5850    6.7879    6.7814    7.4594
    6.7211    6.0661    5.8202    6.5774    6.2479    6.4757    6.2479    6.1799
    6.5999    5.9069    6.2192    6.3399    6.0980    6.1599    6.1898    6.1599
      ⋮
```

```
% Boxplot of log transformed data before normalisation
figure;
boxplot(log_data)
title('Log transformed (log base 2) Data')
```

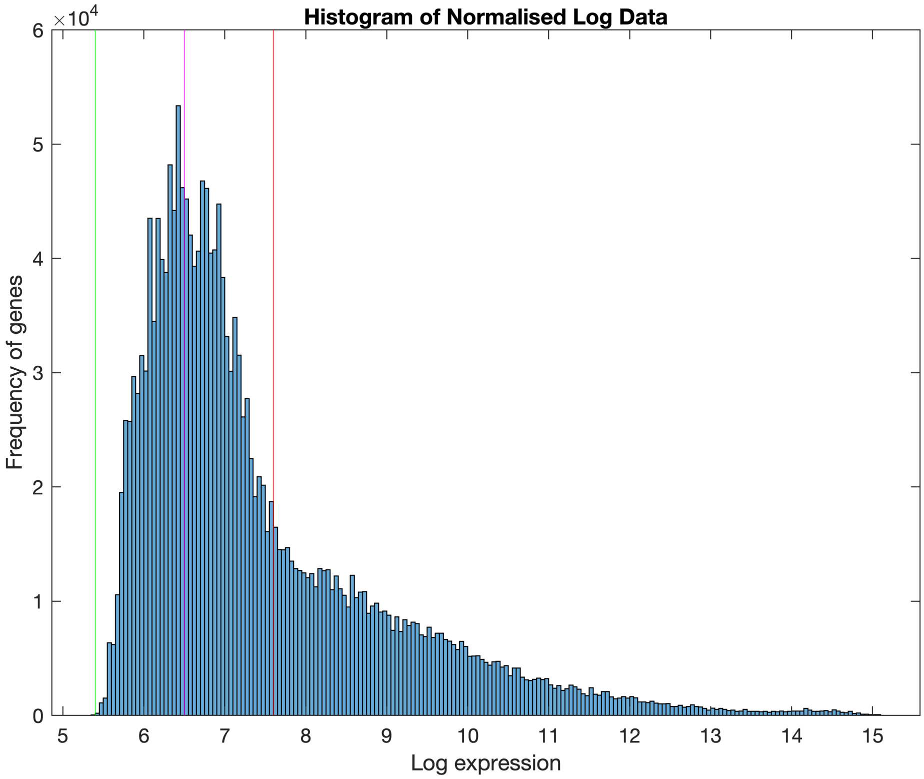


```
% Data is normalised through quantile normalisation
% The median of the ranked values is taken instead of the mean
normalised_logData = quantilenorm(log_data, 'Median', true);

% Boxplot of log transformed data after normalisation
figure;
boxplot(normalised_logData)
title('Normalised Log transformed (log base 2) Data')
```



```
% The cutuff is estimated by observing the histogram
cutOff = 7.6;
% Histogram of normalised data
figure;
histogram(normalised_logData)
xline(5.4, 'Color', 'Green')
xline(6.5, 'Color', 'Magenta')
xline(cutOff, 'Color', 'Red')
title('Histogram of Normalised Log Data')
xlabel('Log expression')
ylabel('Frequency of genes')
```



The histgram shows the data after being log transformed and normalised. Two different distributions appear in this plot:

- Gaussian distribution: represents the unexpressed genes, giving a background noise signal.
- Lognormal distribution: represents the expressed genes giving real signal.

The data that belongs to the Gaussian distribution is not relevant thus, a cutoff is selected to filter unexpressed genes from the dataset. In this case, the cutoff is estimated as follows:

- Start of Gaussian distribution (Green Line): 5.4
- Middle of Gaussian distribution (Pink Line): 6.5
- End of Gaussian distribution (Red Line): 7.6 = 6.5 + 1.1

Now, after preprocessing the data two different dose/timepoint combinations are studied. First, the differentially expressed genes are found. Then, the results of the two combinations are compared. In order for the results to be relevant, the two combinations need to be synchronized. The following are the chosen ones:

- 1st combination: DMSO at 12h and CSA (20uM) at 12h
- 2nd combination: DMSO at 72h and CSA (20uM) at 72h

1st combination: DMSO at 12h and CSA (20uM) at 12h

First, the corresponding columns are retrieved from the previously normalised log data.

```
dmso_12 = [normalised_logData(:, 2), normalised_logData(:, 14), normalised_logData(:, 
csa_12 = [normalised_logData(:, 3), normalised_logData(:, 15), normalised_logData(:, 2
```

Next, the expressed genes are counted and retrieved. A gene is expressed whenever it is expressed both in the dmso sample and the csa sample. Moreover, to be expressed, a gene needs to be in the Lognormal distribution, i.e. greater than the cufoff.

After checking this conditions, 175997 genes turns out to be expressed.

```
% Count and retrieve the expressed genes
expressed_count = 0;
length_data = length(log_data(:,1));
expressed_genes_index = zeros(1, length_data);

for i = 1 : length_data

    expressed_dmso = false;
    expressed_csa = false;

    if (dmso_12(i, 1) >= cutOff && dmso_12(i, 2) >= cutOff && dmso_12(i, 3) >= cutOff)
       expressed_dmso = true;
    end

    if (csa_12(i, 1) >= cutOff && csa_12(i, 2) >= cutOff && csa_12(i, 3) >= cutOff)
        expressed_csa = true;
    end

    if (expressed_dmso || expressed_csa)
        expressed_count = expressed_count + 1;
        expressed_genes_index(expressed_count) = i;
    end
end
expressed_count
```

```
expressed_count = 17597
```

```
% Indeces of the expressed genes
expressed_index = nonzeros(expressed_genes_index)'
```

```
expressed_index = 1×17597
     1     2     4    12    14    15    19    24    27    29    31    34    37···
```

```
% Merged data of the two samples in a signle matrix
index_samples = [2, 14, 26, 3, 15, 27];
expressed = normalised_logData(expressed_index, index_samples)
```

```
expressed = 17597×6
    9.7830    9.8353    9.4051    9.5314    9.6830    9.5878
    8.3332    8.2969    8.3663    8.7748    8.8734    9.0112
    8.6456    8.6339    8.6635    8.8265    8.9915    8.8750
    8.6917    8.8321    8.8408    8.4051    8.3663    8.4553
   10.3077   10.4146   10.0130   10.6402   10.5134   10.2825
    9.0375    8.9144    9.1098    9.0815    9.1006    9.0620
    9.1344    8.8517    8.1997    8.4367    8.4828    8.6402
    8.1319    8.0980    8.0795    7.6330    7.7780    7.3083
    9.9585   10.1189    9.5372    9.1267    9.0000    9.0580
    7.9366    7.6900    7.8494    7.9658    8.1163    8.2761
    ⋮
```

Now, to find which of the expressed genes are differentially expressed the two samples are compared to find the fold change and ttest p-value.

First, the log (base 2) fold change for each expressed gene is calculated. The values are log base 2, so the fold change will be calculated with subtraction. Moreover, the weakest compound (DMSO) is subtracted from the stronger one (CSA 20uM).

```
dmso_exp = expressed(:, 1:3);
csa_exp = expressed(:, 4:6);
foldchanges = zeros(1, expressed_count);
for i = 1 : expressed_count
    foldchanges(1, i) = mean(csa_exp(i, :)) - mean(dmso_exp(i, :));
end
foldchanges
```

```
foldchanges = 1×17597
   -0.0738    0.5543    0.2500   -0.3793    0.2336    0.0608   -0.2087   -0.5300 ···
```

Next, a ttest is performed to compare the two samples and calculate the p-values. Moreover, a muliple testing correction is applied to the previous p-values. The is achieved through the **mafdr** function with the BHFDR flag to perform the Benjamini-Hochberg procedure.

```
[h, p] = ttest2(dmso_exp', csa_exp')
```

```
h = 1×17597
     0     1     1     1     0     0     0     1     1     0     0     0     1 ···
p = 1×17597
    0.6322    0.0015    0.0073    0.0023    0.2165    0.3546    0.5025    0.0192 ···
```

```
adjusted_p = mafdr(p, 'BHFDR', 'true')
```

```
adjusted_p = 1×17597
    0.7162    0.0158    0.0334    0.0185    0.3196    0.4639    0.6041    0.0587 ···
```

Finally, to asnwer the research question, the upregulated and downregulated genes are counted. These are the genes of human cells that have been affected by Cyclosporin A (CSA). The following constratins apply:

- Upregulated genes: log base 2 fold change >1 if upregulated
- Downregulated genes: log base 2 fold change < -1 if downregulated

Moreover, to increase the precision of the results, only the genes with an adjusted p-value lower than 0.05 are counted.

In the end, there are:

- Upregulated genes: 462
- Downregulated genes: 376
- Total human genes affected by CSA: 838

```
count_upregulated = 0;
```

```
count_downregulated = 0;
for i = 1 : length(foldchanges)
    if adjusted_p(i) < 0.05
        if foldchanges(i) > 1
            count_upregulated = count_upregulated + 1;

        end

        if foldchanges(i) < -1
            count_downregulated = count_downregulated + 1;
        end
    end
end
count_upregulated
```

```
count_upregulated = 462
```

```
count_downregulated
```

```
count_downregulated = 376
```

2nd combination: DMSO at 72h and CSA (20uM) at 72h

Repeat expriment with 2nd combination to draw a conclusion and compare it with the previous one.

First, the corresponding columns are retrieved from the previously normalised log data.

```
dmso_72 = [normalised_logData(:, 11), normalised_logData(:, 23), normalised_logData(:,
csa20_72 = [normalised_logData(:, 12), normalised_logData(:, 24), normalised_logData(:
```

Next, the expressed genes are counted and retrieved. 18292 genes turns out to be expressed.

```
% Count and retrieve the expressed genes
expressed_count_2 = 0;
expressed_genes_index_2 = zeros(1, length_data);

for i = 1 : length_data

    expressed_dmso_2 = false;
    expressed_csa_2 = false;

    if (dmso_72(i, 1) >= cutOff && dmso_72(i, 2) >= cutOff && dmso_72(i, 3) >= cutOff)
       expressed_dmso_2 = true;
    end

    if (csa_12(i, 1) >= cutOff && csa_12(i, 2) >= cutOff && csa_12(i, 3) >= cutOff)
        expressed_csa_2 = true;
    end

    if (expressed_dmso_2 || expressed_csa_2)
        expressed_count_2 = expressed_count_2 + 1;
        expressed_genes_index_2(expressed_count_2) = i;
    end
```

```
end
expressed_count_2
```

```
expressed_count_2 = 18292
```

```
% Indeces of the expressed genes
expressed_index_2 = nonzeros(expressed_genes_index_2)'
```

```
expressed_index_2 = 1×18292
     1     2     3     4     8    12    14    15    19    24    27    29    31⋯
```

```
% Merged data of the two samples in a signle matrix
% dmso: 11  23  35
% csa: 12  24  36
index_samples_2 = [11, 23, 35, 12, 24, 36];
expressed_2 = normalised_logData(expressed_index_2, index_samples_2)
```

```
expressed_2 = 18292×6
    9.9968   10.0317   10.0553    9.5142    9.5660    9.8082
    8.0485    8.1849    7.9129    7.7381    7.8734    8.2668
    7.8579    8.0688    8.1649    6.7813    7.0334    7.0715
    8.7347    8.6036    8.6564    8.7863    8.8455    9.1472
    7.6257    7.6830    7.7780    6.6220    6.8392    6.4429
    9.5362    9.3783    9.7960    9.2679    9.1137    9.3151
    9.4168    9.1824    9.1861   10.5274   10.5673   10.8868
    8.6184    8.6402    8.9957    9.2958    9.3526    9.3055
    9.1586    9.2538    8.4918    8.4998    8.5868    8.0498
    8.1344    8.2216    8.2408    8.2784    7.9129    7.9189
      ⋮
```

Now, to find which of the expressed genes are differentially expressed the two samples are compared to find the fold change and ttest p-value. The weakest compound (DMSO) is subtracted from the stronger one (CSA 20uM).

```
dmso_exp2 = expressed_2(:, 1:3);
csa_exp2 = expressed_2(:, 4:6);
% For the subtraction of the mean do: strongest - weakest
foldchanges2 = zeros(1, expressed_count_2);
for i = 1 : expressed_count
    foldchanges2(1, i) = mean(csa_exp2(i, :)) - mean(dmso_exp2(i, :));
end
foldchanges2
```

```
foldchanges2 = 1×18292
   -0.3985   -0.0893   -1.0685    0.2614   -1.0608   -0.3379    1.3987    0.5665⋯
```

Next, a ttest is performed to compare the two samples and calculate the p-values. Moreover, a muliple testing correction is applied to the previous p-values.

```
[h2, p2] = ttest2(dmso_exp2', csa_exp2')
```

```
h2 = 1×18292
     1     0     1     0     1     0     1     1     0     0     1     1     1⋯
p2 = 1×18292
    0.0124    0.6402    0.0011    0.0912    0.0010    0.0680    0.0005    0.0101⋯
```

```
adjusted_p2 = mafdr(p2, 'BHFDR', 'true')
```

```
adjusted_p2 = 1×18292
    0.0431    0.7316    0.0113    0.1723    0.0105    0.1392    0.0079    0.0379 ···
```

Finally, to asnwer the research question, the upregulated and downregulated genes are counted. Moreover, to increase the precision of the results, only the genes with an adjusted p-value lower than 0.05 are considered.

In the end, there are:

- Upregulated genes: 1117
- Downregulated genes: 927
- Total human genes affected by CSA: 2044

```
% upregulated and downregulated are the genes that answer the research
% question = which genes are affected by the CSA
count_upregulated_2 = 0;
count_downregulated_2 = 0;
for i = 1 : length(foldchanges2)
    if adjusted_p2(i) < 0.05
        if foldchanges2(i) > 1
            count_upregulated_2 = count_upregulated_2 + 1;

        end

        if foldchanges2(i) < -1
            count_downregulated_2 = count_downregulated_2 + 1;
        end
    end
end
count_upregulated_2
```

```
count_upregulated_2 = 1117
```

```
count_downregulated_2
```

```
count_downregulated_2 = 927
```

Conclusion:

After analysing the two combinations, it is possible to notice an increase in the expressed genes and, more importantly, in the number of upregulated and downregulated genes. This information is relevant since it means that the number of human genes affected by CSA increase with time. In particular, it goes from 838 to 2044 in the span of 60 hours (from 12 to 72 hours).