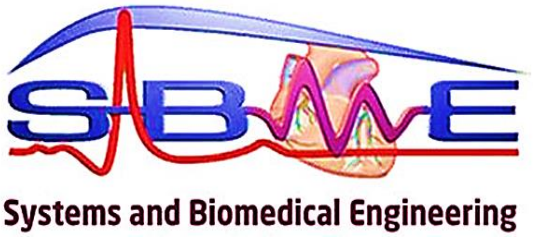


Systems and Biomedical Engineering

Faculty of Engineering

Course Project Report

Submitted to:

Dr. Ibrahim Mohamed Youssef

Submitted by:

Name	Sec	B.N
Aya Abdullah Farag	1	20
Aya Mohamed Abdulrazzaq	1	21
Shorouq Osama Mohamed	1	40
Walaa Salah Abd ellatif	2	44

SBE304_Biostatistics
19 January, 2021

INTRODUCTION

To conduct a study to analyze gene expression (GE) data for the cancer type lung Squamous Cell Carcinoma (LUSC). We need to define some terms regarding the hypothesis test for our project.

•Null hypothesis (H0): A hypothesis associated with a contradiction to the theory that cancer tissues affect expression of the genes.

•Alternative hypothesis (H1): A hypothesis associated with the theory that cancer tissues affect expression of the genes which we would like to prove.

•Region of acceptance: The set of values of the test statistic for which we fail to reject the null hypothesis.

•Region of rejection / Critical region: The set of values of the test statistic for which the null hypothesis is rejected.

•Critical value: The threshold value delimiting the regions of acceptance and rejection for the test statistic.

Significance level of a test (α): The significance level, also denoted as alpha or α, is the probability of rejecting the null hypothesis when it is true. For example, a significance level of 0.05 indicates a 5% risk of concluding that a difference exists when there is no actual difference.

•p-value: The probability, assuming the null hypothesis is true, of observing a result at least as extreme as the test statistic. In case of a composite null hypothesis, the worst case probability.

•Paired t-test: used when the same item or group is tested twice for healthy and cancerous sample from the same subject.

We will be conducting the study based on those measures to determine the significance in differentially expressed genes (DEGs).

Methods:

1) Reading the two txt files for the GE data for this cancer type.

```
import pandas as pd
healthy = pd.read_csv('data/lusc-rsem-fpkm-tcga_paired.txt', sep='\t')
cancer = pd.read_csv('data/lusc-rsem-fpkm-tcga-t_paired.txt', sep='\t')
pd.options.display.max_columns = None
```

2) Filtering the two tables and delete all rows which have 50% of these columns are zeros.

```
#the table of healthy after filtration
h1=healthy[healthy.astype('bool').mean(axis=1)>=0.50]
h1
```

```
#the table of cancer after filteration
c1=cancer[cancer.astype('bool').mean(axis=1)>=0.50]
c1
```

3) Dropping the distinct genes between two tables and getting the filtered tables.

```
#the final table of healthy after filtration and dropping the different genes(h2)
s1 = pd.merge(h1, c1, how='inner', on=['Hugo_Symbol'])
h2=s1.iloc[:, : 52]
h2
```

```
#the final table of cancer after filtration and dropping the different genes(c2)
x=s1[['Hugo_Symbol']]
c2=s1.iloc[:,52 : 103]
c2.insert(0, 'Hugo_Symbol', x)
c2
```

4) Correlation between the normal samples and the diseased samples for each gene:

1- Computing correlation between two samples using pearson correlation:

```
#Computing correlation between two tables using pearson correlation:
from scipy.stats import pearsonr
u_list = []
r_list = []
name=h2['Hugo_Symbol']
i = 0
while i < 17391:
    Gi_h = h2.iloc[i, 2:]
    Gi_c = c2.iloc[i, 2:]
    r, _ = pearsonr(Gi_h, Gi_c)
    u = ['G' + str(i)]
    u_list.append(str(u))
    r_list.append((r))
    i += 1

cc= pd.DataFrame({'Gene_name': u_list,'Hugo_symbol':name,'Entrez_Gene_Id':Id, 'cc': r_list})
cc
```

2- Ranking genes based on their correlation coefficient (CC):

```
#Rank genes based on their correlation coefficient (CC)
cc['cc_Rank'] = cc['cc'].rank(ascending = 0)
cc = cc.set_index('cc_Rank')
cc = cc.sort_index()
cc
```

3- Reporting the highest positive CC and the lowest negative CC:

```
# The gene which has highest positive CC
cc_max = cc.iloc[0, :]
cc_max
```

```
# The gene which has  lowest negative CC
cc_min = cc.iloc[17390, :]
cc_min
```

4- Plotting the expression levels of the above two genes:

```
#Plotting the expression levels of the above two genes:
import matplotlib.pyplot as plt
# Picking high positive from healthy set
Gp_h = h2.iloc[10863, 2:]

# Picking high positive from cancer set
Gp_c = c2.iloc[10863, 2:]

# Picking low negative from healthy set
Gn_h = h2.iloc[13015, 2:]

# Picking low negative from cancer set
Gn_c = c2.iloc[13015, 2:]

plt.figure(1)
plt.scatter(Gp_h,Gp_c, color='blue')
plt.title('high positive correlation')
plt.xlabel("healthy")
plt.ylabel('cancer')
plt.show()

plt.figure(2)
plt.scatter(Gn_h,Gn_c, color='red')
plt.title(' low negative correlation')
plt.xlabel("healthy")
plt.ylabel('cancer')
plt.show()
```

5) Doing **hypothesis testing:**

1-getting pvalues for all genes for paired and independent samples:

```
#the appropriate test statistic:(pvalues for paired and independent)

from scipy.stats import ttest_rel
from scipy.stats import ttest_ind

name=h2['Hugo_Symbol']
Id=h2['Entrez_Gene_Id_x']
n_list=[]
p_valpaired_list=[]
p_valindep_list=[]
i = 0
while i < 17391:
    Gi_h = h2.iloc[i, 2:]
    Gi_c = c2.iloc[i, 2:]
    p_valpaired = ttest_rel(Gi_h,Gi_c).pvalue
    p_valindep = ttest_ind(Gi_h,Gi_c).pvalue
    n=['G' + str(i)]
    n_list.append(n)
    p_valpaired_list.append(p_valpaired)
    p_valindep_list.append((p_valindep))
    i += 1
```

```
# 1.Samples are paired:
Datapaired = pd.DataFrame({'Gene_name': n_list ,'Hugo_symbol':name,'Entrez_Gene_Id':Id ,'pvalue_paired': p_valpaired_lis
Datapaired
```

```
# 2.Samples are independent:
Dataindependent = pd.DataFrame({'Gene_name': n_list ,'Hugo_symbol':name,'Entrez_Gene_Id':Id , 'pvalue-independent': p_va
Dataindependent
```

2- The FDR multiple tests correction method for paired samples:

```
#The FDR multiple tests correction method for paired samples:
from statsmodels.stats.multitest import multipletests
import numpy as np
corrected_p_values1 = multipletests(p_valpaired_list, alpha=0.05, method='fdr_bh')[1]
significance_genes1 =pd.DataFrame({'Gene_name': n_list ,'Hugo_symbol':name,'Entrez_Gene_Id':Id ,'pvalue_paired': p_valpa
significance_genes1
```

```
significance_genes1['significance:p_value'] = significance_genes1['pvalue_paired'].apply(lambda x: x < 0.05)
significance_genes1['significance:p_value_fdr'] = significance_genes1['pval_paired_fdr'].apply(lambda x: x < 0.05)
significance_genes1
```

Comparing pvalues with Significance level of a test (α)

```
# significant genes after fdr correction for paired samples:
diffrentially_genes1 = significance_genes1[significance_genes1['significance:p_value_fdr']== True]
diffrentially_genes1
```

3- Getting significant genes for paired after fdr correction:

4- The FDR multiple tests correction method for independent samples:

```
#The FDR multiple tests correction method for independent samples:
corrected_p_values2 = multipletests(p_valindep_list, alpha=0.05, method='fdr_bh')[1]
significance_genes2 =pd.DataFrame({'Gene_name': n_list ,'Hugo_symbol':name,'Entrez_Gene_Id':Id ,'pvalue_independent': p_
significance_genes2
```

Comparing pvalues with Significance level of a test (α)

```
significance_genes2['significance:p_value'] = significance_genes2['pvalue_independent'].apply(lambda x: x < 0.05)
significance_genes2['significance:p_value_fdr'] = significance_genes2['pval_independent_fdr'].apply(lambda x: x < 0.05)
significance_genes2
```

5- Getting significant genes for independent after fdr correction:

```
# significant genes after fdr correction for independent samples:
diffrentially_genes2 = significance_genes2[significance_genes2['significance:p_value_fdr']== True]
diffrentially_genes2
```

6- Getting the common genes between the two DEGs sets (paired and independent) after the FDR correction:

```
#the common genes between the two DEGs sets (paired and independent) after the FDR correction:
s = pd.merge(diffrentially_genes1, diffrentially_genes2, how='inner', on=['Hugo_symbol'])
common=s[['Gene_name_x','Hugo_symbol','Entrez_Gene_Id_x']]
common
```

7- Getting the distinct between the two tables after the FDR correction:

```
#the distinct genes in the paired table and not in the independent table after the FDR correction:
d1 = diffrentially_genes1[~(diffrentially_genes1['Hugo_symbol'].isin(diffrentially_genes2['Hugo_symbol']))].reset_index
d1
```

```
#the distinct genes in the independent table and not in the paired table after the FDR correction:
d2 = diffrentially_genes2[~(diffrentially_genes2['Hugo_symbol'].isin(diffrentially_genes1['Hugo_symbol']))].reset_index
d2
```

Software packages which we use:

•NumPy and pandas

• scipy.stats.pearsonr

• scipy.stats.ttest_ind

• scipy.stats.ttest_rel

•statsmodels.stats.multitest.multipletests

RESULTS AND DISCUSSION

1) filtration tables:

-Before filtering tables, we had two tables that each had 19648 rows x 52 columns.

-But now after filtering tables, the number of rows decreased and we have the healthy table with 17726 rows x 52 columns and the cancer table with 17825 rows x 52 columns.

-Then, we selected the common rows between two tables and became 17391 rows x 52 columns.

-Now, our tables are h2 and c2.

2) Correlation:

- We computed the correlation between the normal samples and the diseased samples for each gene and ranked genes based on their correlation coefficient (CC).

cc	Entrez_Gene_Id	Hugo_symbol	Gene_name	cc_Rank
0.969044	374	AREGB	['G10863']	**1.0**
0.930574	162998	OR7D2	['G5395']	**2.0**
0.878029	2978	GUCA1A	['G13459']	**3.0**
0.847577	100462981	MTRNR2L2	['G6606']	**4.0**
0.826948	0	NUTM2E	['G17041']	**5.0**
...	...	...	...	**...**
0.402969-	6277	S100A6	['G3816']	**17387.0**
0.416206-	9730	VPRBP	['G12407']	**17388.0**
0.418618-	64145	ZFYVE20	['G11429']	**17389.0**
0.424345-	5795	PTPRJ	['G13610']	**17390.0**
0.452807-	55731	FAM222B	['G13015']	**17391.0**

rows × 4 columns 17391

-From this table it is clear that the gene which has highest positive CC is (AREGB) and the gene which has lowest negative CC is (FAM222B).

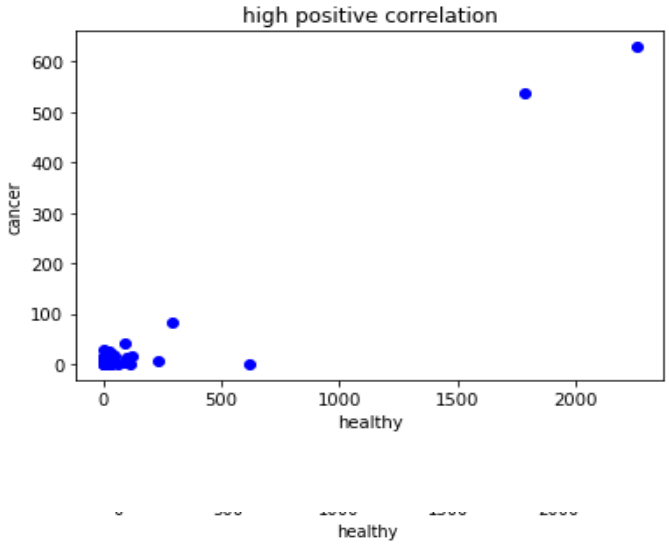
```
Gene_name          ['G10863']
Hugo_symbol             AREGB
Entrez_Gene_Id            374
cc                   0.969044
Name: 1.0, dtype: object
```

```
Gene_name          ['G13015']
Hugo_symbol            FAM222B
Entrez_Gene_Id           55731
cc                   -0.452807
Name: 17391.0, dtype: object
```

-And we Ploted the expression levels of these two genes:

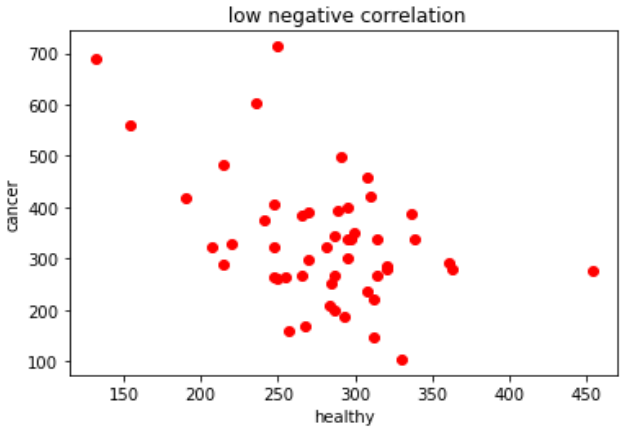
1- The gene of the highest positive CC:

That ratio indicates the amount of variation between the same gene in the two samples: the healthy and the cancer one. This gene has the highest positive correlation coefficient which is equal 0.9690441442970705. This gene is AREGB .



2- The gene of the lowest negative CC:

That ratio indicates the amount of variation between the same gene in the two samples: the healthy and the cancer one. This gene has the lowest negative correlation coefficient which is equal -0.45280727852470826. This gene is FAM222B.



3) Hypothesis Testing:

After getting the p-values for each paired and independent test which reject the null hypothesis i.e. alpha < 0.05 and applying the FDR correction, we can clearly see the number of rows for the paired test decreased from 17391to **12435** rows and the number of rows for the independent test decreased from 17391 to **12346** rows .

In [12]:
```
significance_genes1['significance:p_value'] = significance_genes1['pvalue_paired'].apply(lambda x: x < 0.05)
significance_genes1['significance:p_value_fdr'] = significance_genes1['pval_paired_fdr'].apply(lambda x: x < 0.05)

diffrentially_genes1 = significance_genes1[significance_genes1['significance:p_value_fdr']== True]
diffrentially_genes1
```

Out[12]:

	Gene_name	Hugo_symbol	Entrez_Gene_Id	pvalue_paired	pval_paired_fdr	significance:p_value	significance:p_value_fdr
0	[G0]	HIST3H2A	92815	4.043607e-08	1.457760e-07	True	True
2	[G2]	LXN	56925	2.322367e-04	4.599509e-04	True	True
3	[G3]	CNKSR2	22866	3.420577e-12	2.461202e-11	True	True
6	[G6]	GSDMD	79792	3.041721e-06	8.159582e-06	True	True
7	[G7]	AKR1C1	1645	1.938575e-05	4.568878e-05	True	True
...	...	...	...	...	...	...	...
17386	[G17386]	ZNF521	25925	4.142164e-06	1.087506e-05	True	True
17387	[G17387]	SPINT2	10653	2.452619e-07	7.801993e-07	True	True
17388	[G17388]	HAVCR2	84868	2.435125e-13	2.173987e-12	True	True
17389	[G17389]	CTD-2116N17.1	0	4.129496e-11	2.433618e-10	True	True
17390	[G17390]	FUT2	2524	1.166719e-06	3.343838e-06	True	True

12435 rows × 7 columns

In [40]:
```
significance_genes2['significance:p_value'] = significance_genes2['pvalue_independent'].apply(lambda x: x < 0.05)
significance_genes2['significance:p_value_fdr'] = significance_genes2['pval_independent_fdr'].apply(lambda x: x < 0.05)


diffrentially_genes2 = significance_genes2[significance_genes2['significance:p_value_fdr']== True]
diffrentially_genes2
```

Out[40]:

	Gene_name	Hugo_symbol	Entrez_Gene_Id	pvalue_independent	pval_independent_fdr	significance:p_value	significance:p_value_fdr
0	[G0]	HIST3H2A	92815	3.607140e-09	1.382062e-08	True	True
2	[G2]	LXN	56925	8.164044e-05	1.725372e-04	True	True
3	[G3]	CNKSR2	22866	6.374652e-15	5.050641e-14	True	True
6	[G6]	GSDMD	79792	5.344289e-06	1.340582e-05	True	True
7	[G7]	AKR1C1	1645	7.857877e-06	1.926637e-05	True	True
...	...	...	...	...	...	...	...
17386	[G17386]	ZNF521	25925	2.273493e-06	6.007037e-06	True	True
17387	[G17387]	SPINT2	10653	5.250215e-08	1.725042e-07	True	True
17388	[G17388]	HAVCR2	84868	1.228186e-14	9.368150e-14	True	True
17389	[G17389]	CTD-2116N17.1	0	1.068283e-12	6.317075e-12	True	True
17390	[G17390]	FUT2	2524	2.133666e-07	6.463436e-07	True	True

12346 rows × 7 columns

Also, getting the common and distinct genes as the common number was the less between the two which is 12267 and the distinct number of rows were 168 and 79

In [170]:
```
w=d2[['Gene_name_x']]
w
```

Out[170]:

	Gene_name_x
28	[G40]
256	[G353]
280	[G386]
399	[G548]
403	[G552]
...	...
12168	[G17032]
12220	[G17099]
12227	[G17111]
12231	[G17115]
12304	[G17211]

168 rows × 1 columns

In [169]:
```
z=d1[['Gene_name_y']]
z
```

Out[169]:

	Gene_name_y
12435	[G34]
12436	[G147]
12437	[G533]
12438	[G1104]
12439	[G2009]
...	...
12509	[G15743]
12510	[G16010]
12511	[G16418]
12512	[G16536]
12513	[G17171]

79 rows × 1 columns

I. **C**ONCLUSION

II. Calculating the number of DEGs after the FDR multiple test correction method in the paired and independent t-test we can see the number of genes which reject the proposed null hypothesis i.e. the genes expression level does not differ from one condition (healthy) to another (diseased).

III. **C**ONTRIBUTIONS

- Aya Abdallah: reading and importing the .txt file, finding distinct and common genes.
- Aya Abdulrazzaq: independent and paired t-test and FDR correction.
- Shorouq Osama: correlation coefficients ranking and plotting.
- Walaa Salah: Filtering, preparing the healthy and cancer rows, paired t-test and FDR correction.