Yuyue Wang
Prof. Gang Fang
Bioinformatics Project 1 Report
11.26.2016

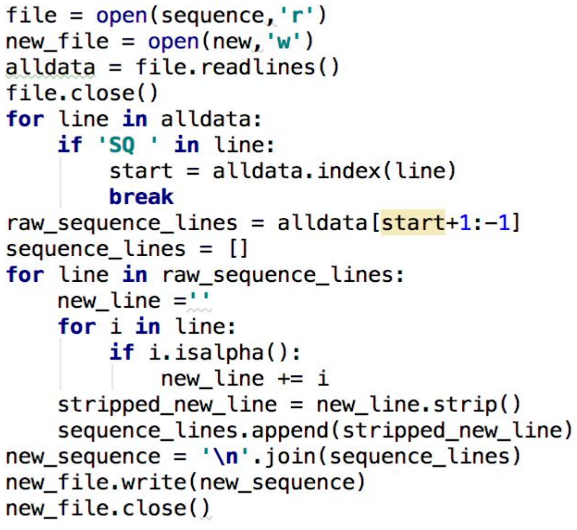
Note: please read the “README.txt” file from the README folder in code_folder to get the info of the code files and other txt files.

Introduction

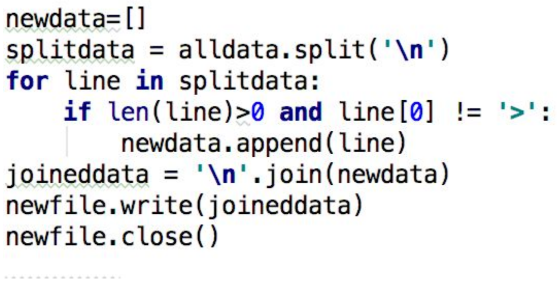
In this project, I would like to get some clues in the reason of some extremophiles’ thermophilic property. In order to do that, I would investigate the dinucleotide frequency and dipeptide frequency for the whole genome of certain extremophile, because it’s a common guess that some proteins or the gene segments of certain extremophile make it special. I would compare these frequencies with the ones of my interested species’ own expected values and the ones of its four neighbor species(controls). The purpose of comparison/ the hypothesis is that there could be some dinucleotide or dipeptide usage bias in certain extremophiles’ genome/peptide sequence that makes it thermophilic.

Materials and Methods

To investigate the thermophilic property, I chose to study Anaerolinea thermophila UNI-1(AT.). It is a species of filamentous thermophilic bacteria, the type and only species of its genus. It is Gram-negative, non-spore-forming(Wikipedia). The optimal growth temperature for AT.’s growth is 55℃ and the optimal PH is 7.0(Yuji Sekiguchi et al., 2003). In this project, I extracted the genome and protein information from EMBL-EBI website(http://www.ebi.ac.uk/genomes/bacteria.html). The python code for extracting pure DNA/Protein sequences and making frequency tables are in the code folder (refer to README.txt for detailed info of the code). Below is some screenshots of the codes: Partial code for extracting pure DNA(left)/Protein(right)sequences(basically the idea is to find symbols of the starting line of sequence and keep whatever after them except some junk symbols at the end of each line):

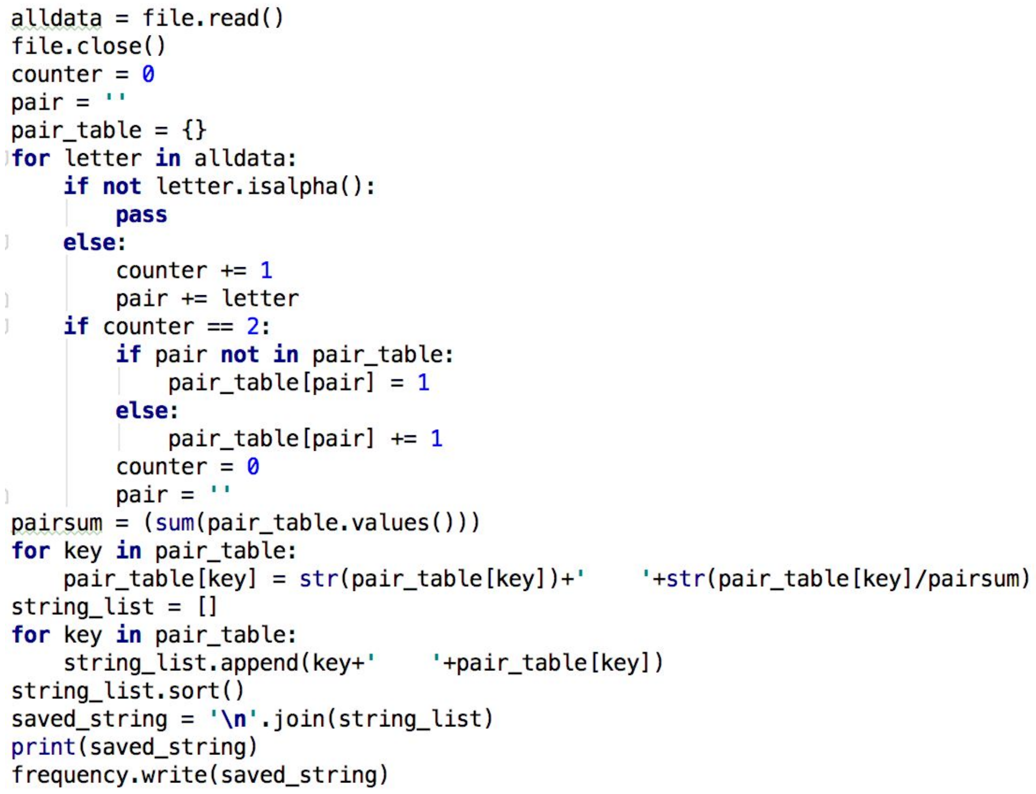


```
file = open(sequence,'r')
new_file = open(new,'w')
alldata = file.readlines()
file.close()
for line in alldata:
    if 'SQ ' in line:
        start = alldata.index(line)
        break
raw_sequence_lines = alldata[start+1:-1]
sequence_lines = []
for line in raw_sequence_lines:
    new_line =''
    for i in line:
        if i.isalpha():
            new_line += i
    stripped_new_line = new_line.strip()
    sequence_lines.append(stripped_new_line)
new_sequence = '\n'.join(sequence_lines)
new_file.write(new_sequence)
new_file.close()
```



```
newdata=[]
splitdata = alldata.split('\n')
for line in splitdata:
    if len(line)>0 and line[0] != '>':
        newdata.append(line)
joineddata = '\n'.join(newdata)
newfile.write(joineddata)
newfile.close()
```

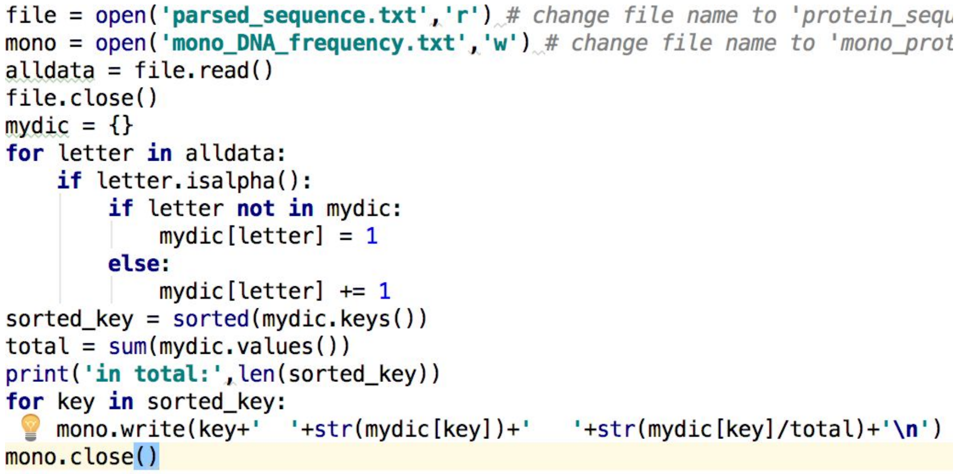
Partial code for make dinucleotide/dipeptide frequency tables(this piece of code deals with single species). The general idea is to read through the whole sequence in pairs and make a python dictionary(map) where the key is the dinucleotide/dipeptide name and the value is the number of occurrence and the frequency of it. The following code works for all sequences(DNA/protein/control species/interested species):



```
alldata = file.read()
file.close()
counter = 0
pair = ''
pair_table = {}
for letter in alldata:
    if not letter.isalpha():
        pass
    else:
        counter += 1
        pair += letter
    if counter == 2:
        if pair not in pair_table:
            pair_table[pair] = 1
        else:
            pair_table[pair] += 1
        counter = 0
        pair = ''
pairsum = (sum(pair_table.values()))
for key in pair_table:
    pair_table[key] = str(pair_table[key])+'    '+str(pair_table[key]/pairsum)
string_list = []
for key in pair_table:
    string_list.append(key+'    '+pair_table[key])
string_list.sort()
saved_string = '\n'.join(string_list)
print(saved_string)
frequency.write(saved_string)
```

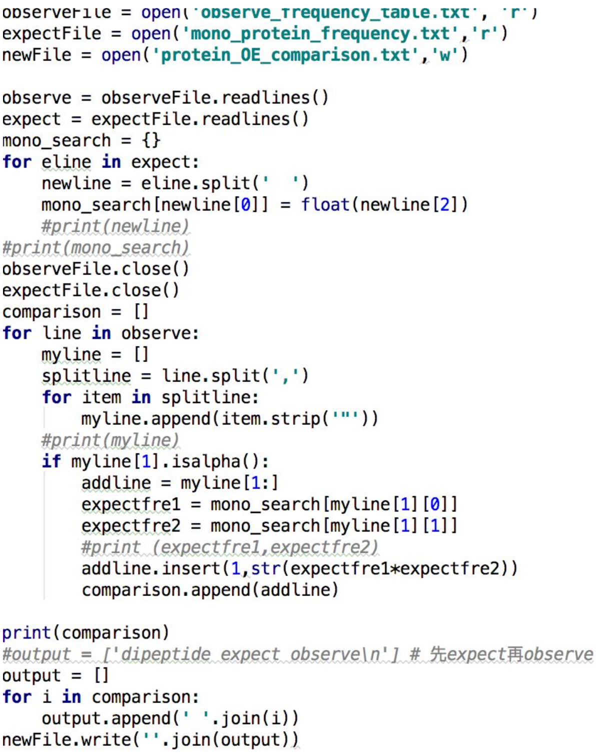
To get the theoretical numbers of occurrence and frequencies of the dinucleotide/dipeptide, I used python code to them separately.

Firstly, to calculate the number of occurrence(as shown in the partial code below):

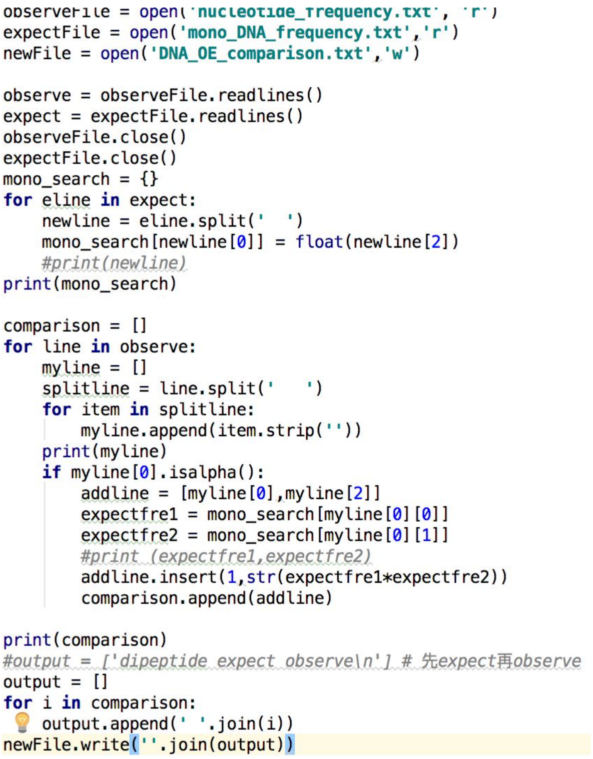


```
file = open('parsed_sequence.txt','r') # change file name to 'protein_sequ
mono = open('mono_DNA_frequency.txt','w') # change file name to 'mono_prot
alldata = file.read()
file.close()
mydic = {}
for letter in alldata:
    if letter.isalpha():
        if letter not in mydic:
            mydic[letter] = 1
        else:
            mydic[letter] += 1
sorted_key = sorted(mydic.keys())
total = sum(mydic.values())
print('in total:',len(sorted_key))
for key in sorted_key:
    mono.write(key+'   '+str(mydic[key])+'   '+str(mydic[key]/total)+'\n')
mono.close()
```

Then, Taking the table of observed dipeptide frequency(got from some other code, can get in the code folder) and the table of single nucleotide/peptide occurrence(done in the above partial code), another piece of code calculates the expected frequency of each dinucleotide/dipeptide and stores both observed and expected frequency in an output table(as shown in the partial code below, left is for dipeptide, right is for dinucleotide):

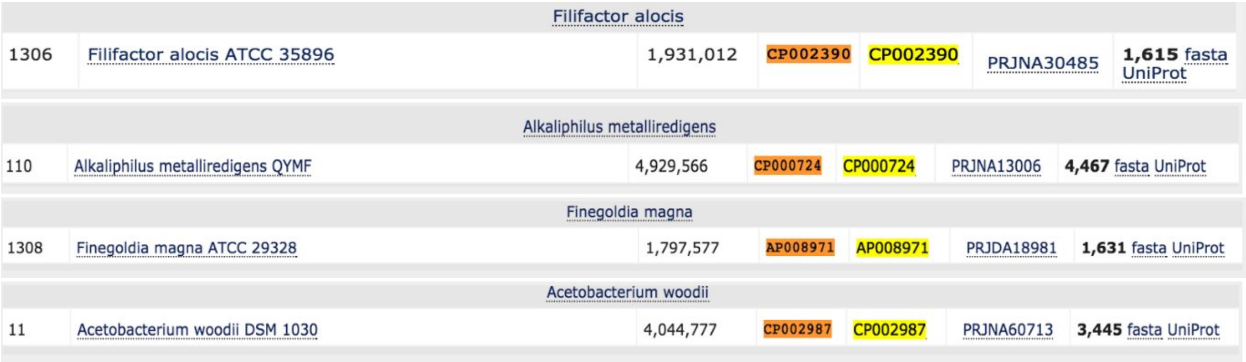


```
observeFile = open('observe_frequency_table.txt', 'r')
expectFile = open('mono_protein_frequency.txt','r')
newFile = open('protein_OE_comparison.txt','w')

observe = observeFile.readlines()
expect = expectFile.readlines()
mono_search = {}
for eline in expect:
    newline = eline.split('  ')
    mono_search[newline[0]] = float(newline[2])
    #print(newline)
#print(mono_search)
observeFile.close()
expectFile.close()
comparison = []
for line in observe:
    myline = []
    splitline = line.split(',')
    for item in splitline:
        myline.append(item.strip('"'))
    #print(myline)
    if myline[1].isalpha():
        addline = myline[1:]
        expectfre1 = mono_search[myline[1][0]]
        expectfre2 = mono_search[myline[1][1]]
        #print (expectfre1,expectfre2)
        addline.insert(1,str(expectfre1*expectfre2))
        comparison.append(addline)

print(comparison)
#output = ['dipeptide expect observe\n'] # 先expect再observe
output = []
for i in comparison:
    output.append(' '.join(i))
newFile.write(''.join(output))
```



```
observeFile = open('nucleotide_frequency.txt', 'r')
expectFile = open('mono_DNA_frequency.txt','r')
newFile = open('DNA_OE_comparison.txt','w')

observe = observeFile.readlines()
expect = expectFile.readlines()
observeFile.close()
expectFile.close()
mono_search = {}
for eline in expect:
    newline = eline.split('  ')
    mono_search[newline[0]] = float(newline[2])
    #print(newline)
print(mono_search)

comparison = []
for line in observe:
    myline = []
    splitline = line.split('    ')
    for item in splitline:
        myline.append(item.strip(''))
    print(myline)
    if myline[0].isalpha():
        addline = [myline[0],myline[2]]
        expectfre1 = mono_search[myline[0][0]]
        expectfre2 = mono_search[myline[0][1]]
        #print (expectfre1,expectfre2)
        addline.insert(1,str(expectfre1*expectfre2))
        comparison.append(addline)

print(comparison)
#output = ['dipeptide expect observe\n'] # 先expect再observe
output = []
for i in comparison:
    output.append(' '.join(i))
newFile.write(''.join(output))
```

The four control species are shown in the table below:



Filifactor alocis

1306	Filifactor alocis ATCC 35896	1,931,012	CP002390	CP002390	PRJNA30485	**1,615** fasta UniProt

Alkaliphilus metalliredigens

110	Alkaliphilus metalliredigens QYMF	4,929,566	CP000724	CP000724	PRJNA13006	**4,467** fasta UniProt

Finegoldia magna

1308	Finegoldia magna ATCC 29328	1,797,577	AP008971	AP008971	PRJDA18981	**1,631** fasta UniProt

Acetobacterium woodii

11	Acetobacterium woodii DSM 1030	4,044,777	CP002987	CP002987	PRJNA60713	**3,445** fasta UniProt

These four genomes are picked because they locate close to my AT.(my interested thermophilic species) in the genetic tree but without the thermophilic property.
After plotting(using R) the dipeptide/dinucleotide frequencies of control species VS. experimental species, and expected values VS. observed values, I will set some appropriate high/low thresholds, which will be symmetric(high threshold * low threshold ~= 1). Then we can get some outliers from the plots that are beyond the threshold, which are the dipeptide/dinucleotide in AT. that are much more/less than in expected models or non-thermophilic neighbor species.
To statistically test whether those outliers have a significant difference with whatever they are compared with, proportional test will be done to each outlier data(by using prop.test in R)

Results

Note that the four control species are abbreviated to species 1/2/3/4 in the plots :

species1-

Filifactor alocis

1306	Filifactor alocis ATCC 35896	1,931,012	CP002390	CP002390	PRJNA30485	**1,615** fasta UniProt

species2-

Acetobacterium woodii

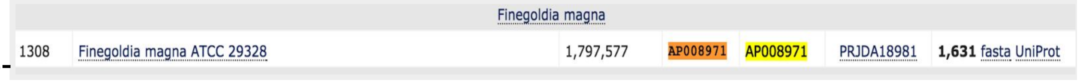
11	Acetobacterium woodii DSM 1030	4,044,777	CP002987	CP002987	PRJNA60713	**3,445** fasta UniProt

species3-

Alkaliphilus metalliredigens

110	Alkaliphilus metalliredigens QYMF	4,929,566	CP000724	CP000724	PRJNA13006	**4,467** fasta UniProt

species4-



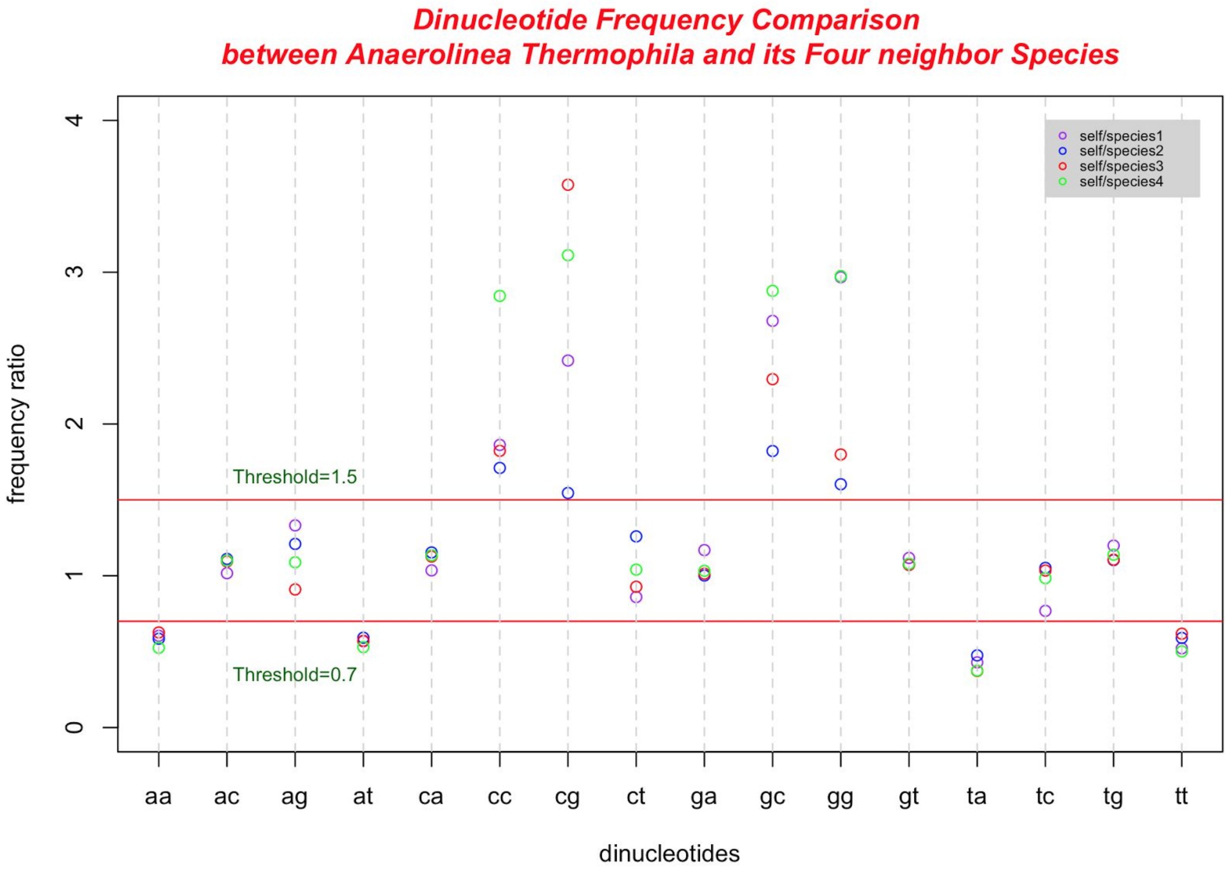
Finegoldia magna

1308	Finegoldia magna ATCC 29328	1,797,577	AP008971	AP008971	PRJDA18981	**1,631** fasta UniProt

Four plots are drawn by using R to compare the frequency of dinucleotide and dipeptide of observed AT. with expected AT. and other four control species. As stated before, high/low thresholds are properly chosen to find out the outliers. The points beyonds red threshold lines(shown in the following plots) corresponds to the dipeptide/dinucleotide in AT. that are much more/less than in expected models or non-thermophilic neighbor species.

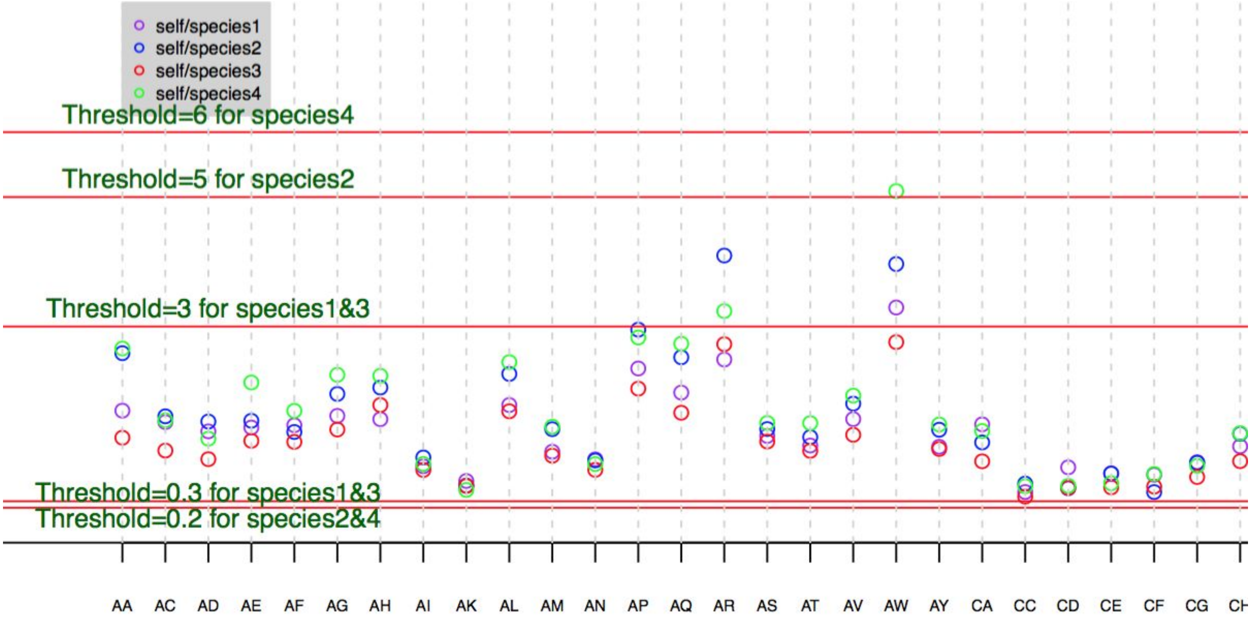
- 1). plot of the observed dinucleotide versus dinucleotide of control species

(x-axis is the name of the dinucleotide, y-axis is the ratio of AT. dinucleotide frequency over frequency of control species, points of different color refer to the comparison between AT. and different control species)



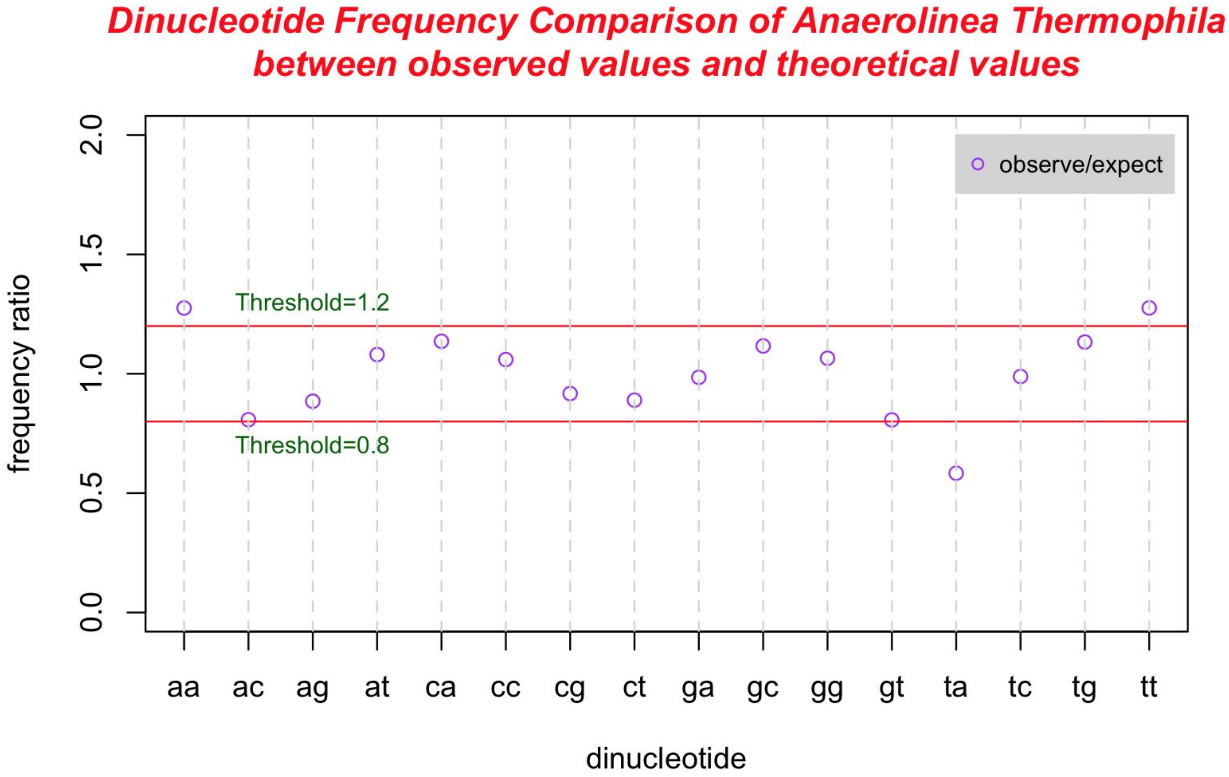
- 2). plot of the observed dipeptide versus dipeptide of control species

(check dipeptide_comparison.pdf in "plot_result" folder for full plot) Below is a screenshot of the plot(x-axis is the name of the dipeptide, y-axis is the ratio of AT. dipeptide frequency over frequency of control species, points of different color refer to the comparison between AT. and different control species):



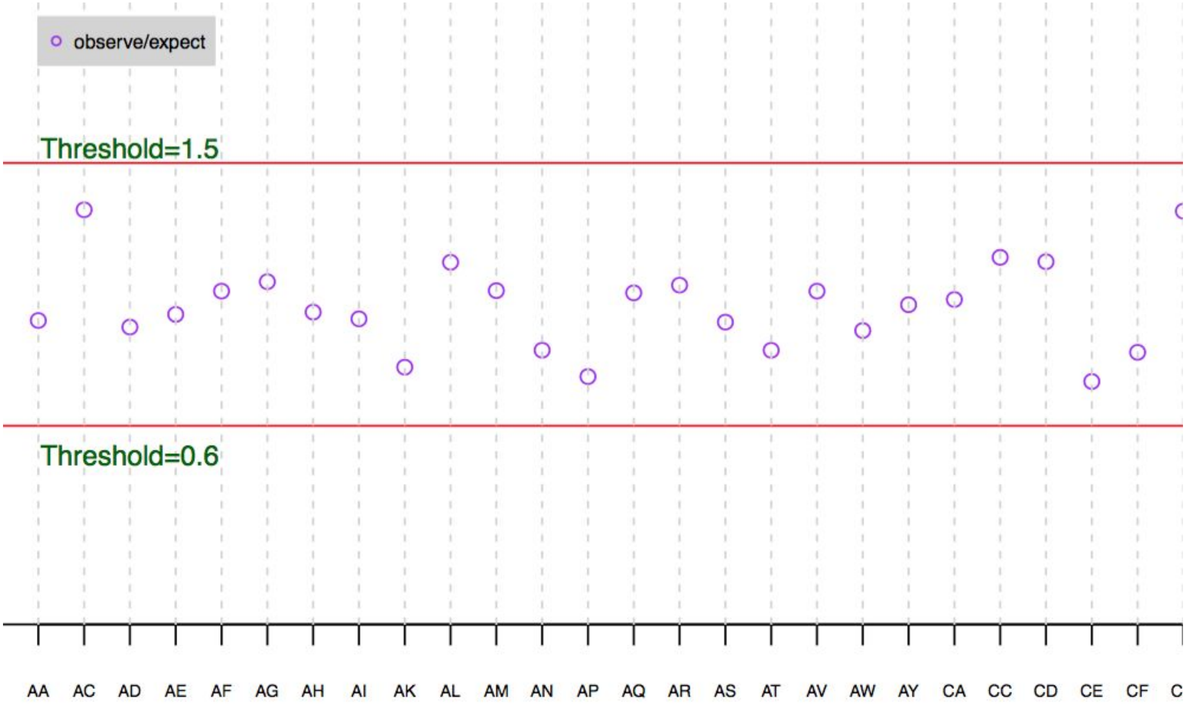
- 3). plot of the comparison between observed versus theoretical dinucleotide

(x-axis is the name of the dinucleotide, y-axis is the ratio of observed frequency over expected frequency)



- 4). plot of the comparison between observed versus theoretical dipeptide.

(check dipeptide_expect_comparison.pdf in "plot_result" folder for full plot) Below is a screenshot of the plot(x-axis is the name of the dipeptide, y-axis is the ratio of observed frequency over expected frequency):

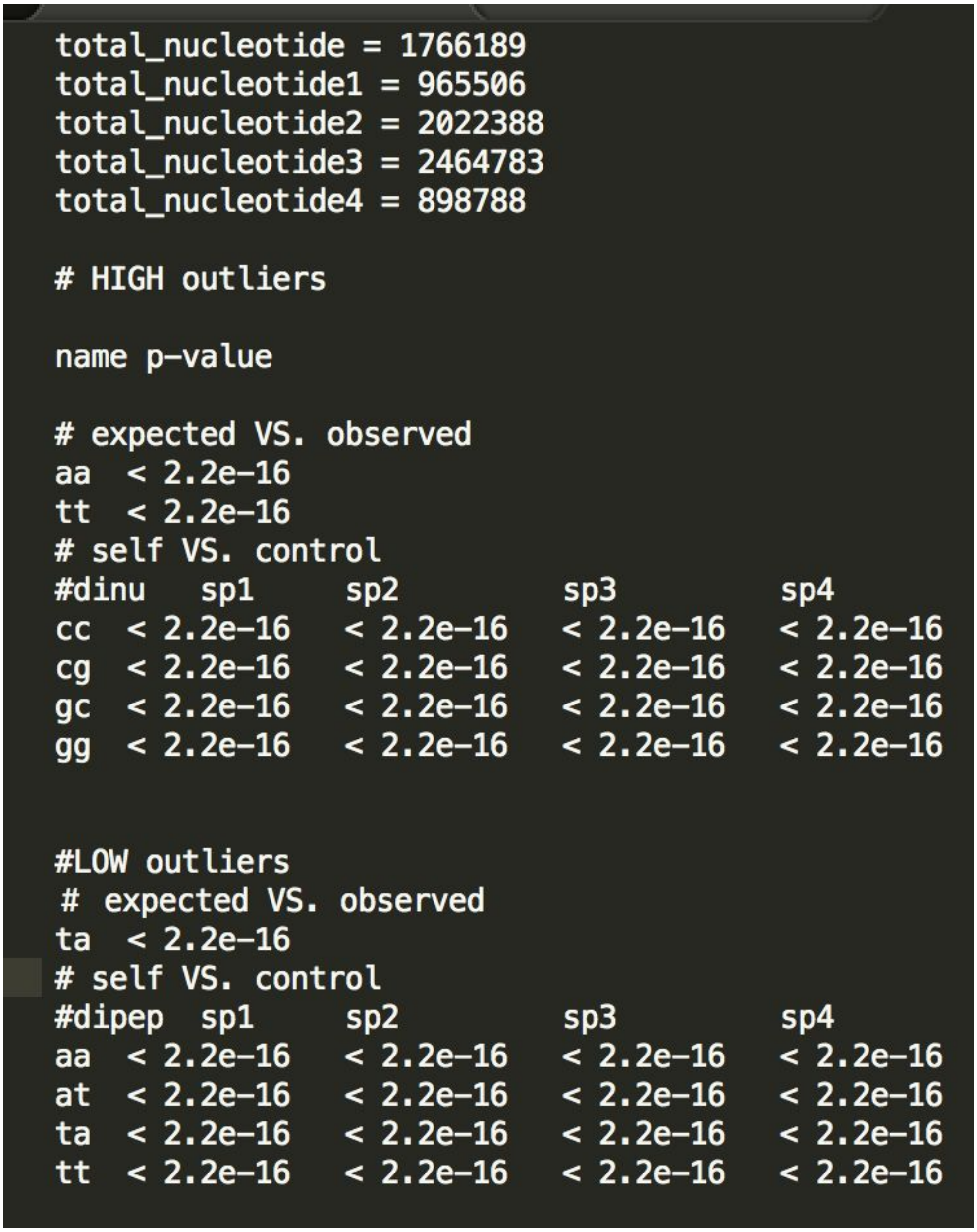


The significance of the differences for the outliers will be discussed in the Discussion section.

Discussions

- The outliers of dinucleotide comparisons are listed in the following table:

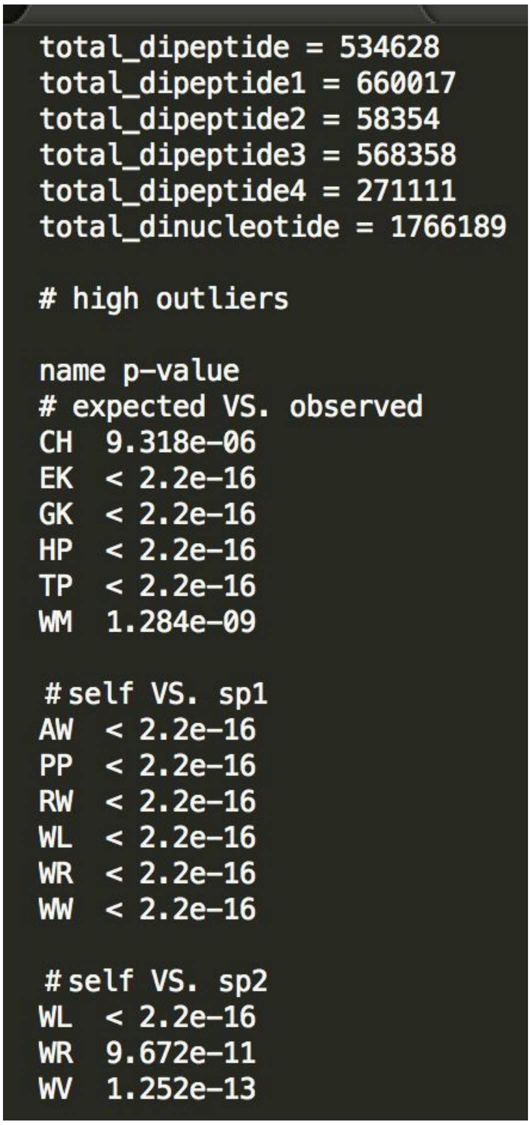
Note: high outliers mean the outliers that are above the high threshold, vice versa for low outliers. The numbers are P-value calculated by using “pro.test” command in R when passing in the numbers of occurrence of the compared frequency and the numbers of total dinucleotides.



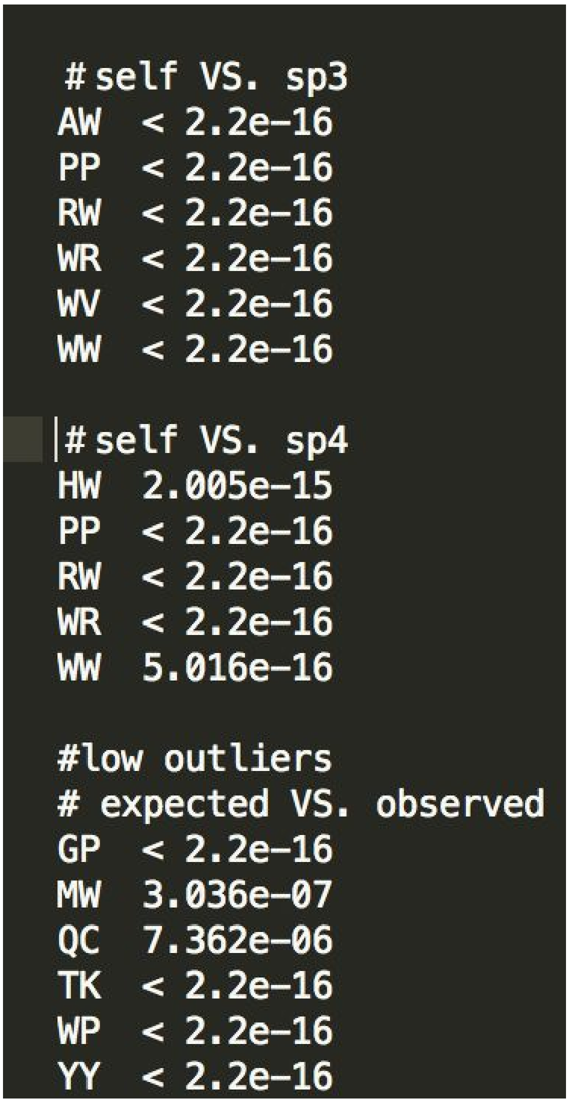
```
total_nucleotide = 1766189
total_nucleotide1 = 965506
total_nucleotide2 = 2022388
total_nucleotide3 = 2464783
total_nucleotide4 = 898788

# HIGH outliers

name p-value

# expected VS. observed
aa  < 2.2e-16
tt  < 2.2e-16
# self VS. control
#dinu   sp1      sp2          sp3          sp4
cc  < 2.2e-16   < 2.2e-16   < 2.2e-16   < 2.2e-16
cg  < 2.2e-16   < 2.2e-16   < 2.2e-16   < 2.2e-16
gc  < 2.2e-16   < 2.2e-16   < 2.2e-16   < 2.2e-16
gg  < 2.2e-16   < 2.2e-16   < 2.2e-16   < 2.2e-16


#LOW outliers
 #  expected VS. observed
ta  < 2.2e-16
# self VS. control
#dipep  sp1      sp2          sp3          sp4
aa  < 2.2e-16   < 2.2e-16   < 2.2e-16   < 2.2e-16
at  < 2.2e-16   < 2.2e-16   < 2.2e-16   < 2.2e-16
ta  < 2.2e-16   < 2.2e-16   < 2.2e-16   < 2.2e-16
tt  < 2.2e-16   < 2.2e-16   < 2.2e-16   < 2.2e-16
```

- The outliers of dipeptide comparisons are listed in the following tables:

Note: high outliers mean the outliers that are above the high threshold, vice versa for low outliers. The numbers are P-value calculated by using “pro.test” command in R when passing in the numbers of occurrence of the compared frequency and the numbers of total dipeptides.



```
total_dipeptide = 534628
total_dipeptide1 = 660017
total_dipeptide2 = 58354
total_dipeptide3 = 568358
total_dipeptide4 = 271111
total_dinucleotide = 1766189

# high outliers

name p-value
# expected VS. observed
CH  9.318e-06
EK  < 2.2e-16
GK  < 2.2e-16
HP  < 2.2e-16
TP  < 2.2e-16
WM  1.284e-09

 #self VS. sp1
AW  < 2.2e-16
PP  < 2.2e-16
RW  < 2.2e-16
WL  < 2.2e-16
WR  < 2.2e-16
WW  < 2.2e-16

 #self VS. sp2
WL  < 2.2e-16
WR  9.672e-11
WV  1.252e-13
```



```
 #self VS. sp3
AW  < 2.2e-16
PP  < 2.2e-16
RW  < 2.2e-16
WR  < 2.2e-16
WV  < 2.2e-16
WW  < 2.2e-16

 #self VS. sp4
HW  2.005e-15
PP  < 2.2e-16
RW  < 2.2e-16
WR  < 2.2e-16
WW  5.016e-16

#low outliers
# expected VS. observed
GP  < 2.2e-16
MW  3.036e-07
QC  7.362e-06
TK  < 2.2e-16
WP  < 2.2e-16
YY  < 2.2e-16
```

Since all the p-values of all the outlier dinucleotide/dipeptide are clearly less than 0.05, the differences are all significant.

If we take a look back at the plots, it's clearly shown there are significantly more [cc, cg, gc, gg] and less [aa, at, ta, tt] pairs in AT. than in ALL the control species. That demonstrates the fact that in AT., c and g/a and t are more likely to appear together.Plus, dipeptide pair WW appears in the outlier three times when comparing to control species. These results have verified our hypothesis that there are some dinucleotide or dipeptide usage bias in certain extremophiles(in this case, AT.'s)' genome/peptide sequence that makes it thermophilic.

We may make a reasonable guess that higher likelihood in having C(cytosine) and G(guanine) in a roll and less likelihood in having A(adenine) and T(thymine) in a roll in the genome may lead to thermophilic property. Furthermore, the results also suggest that the gene that codes for certain proteins that makes AT. (or even all extremophiles) survive in high temperature may be rich in CG pairs. And the gene that codes for proteins that can't survive in high temperature may be rich in AT pairs. Also, the dipeptide comparison may indicate that W tends to cluster together in thermophilic bacteria.

The results of this project is quite significant because there are clusters in the outliers in both dinucleotide and dipeptide, which guides us to some specific interesting pairs. Further study may be conducted on the expressed proteins related to those specified outliers.

In addition, the control genomes are picked up from phylogenetic tree which are close to my interested species but without the thermophilic property. The results got in this way may be “noisy” because there may be many other variables related the thermophilic property of the studied bacteria.

References

Yuji Sekiguchi, Takeshi Yamada, Satoshi Hanada, Akiyoshi Ohashi, Hideki Harada and Yoichi Kamagata(2003), Anaerolinea thermophila gen. nov., sp. nov. and Caldilinea aerophila gen. nov., sp. nov., novel filamentous thermophiles that represent a previously uncultured lineage of the domain Bacteria at the subphylum level, *International Journal of Systematic and Evolutionary Microbiology* 53, 1843–1851