DNA Methylation

Kat Kessler, James Vesto

This project was primarily motivated by DNA methylation and the software used to analyze it. When DNA has been methylated and then subjected to a bisulfite conversion, there are single nucleotide variants found in the converted DNA that indicate methylation has taken place. The conserved cytosine or guanine indicate methylated DNA, those that have changed from C → T on the forward strand or G → A on the reverse strand indicate DNA that was not methylated. This becomes slightly more complex as CpG complexes are only occasionally methylated. For this reason we wanted to design a program that would be able to give the percentage of cytosine or guanine that has been methylated (the number of conserved C or G/the total number of C and C → T, or G and G → A), additionally we wanted to report the number of CpG complexes that have been methylated (number of CpG conserved/number of total CpG complexes). This will allow us to report enough information that the user will understand what is happening within the DNA at a molecular level.

Data was obtained from a genome database called Phytozome v12.1 and we attempted to acquire data from sanger sequencing. In terms of the original (reference) strand, we obtained from Phytozome the 500 bp sequence upstream of the UTR of gene AT5G63860.1 on *Arabidopsis Thaliana,* which was the location in the DNA that was amplified and sent to be sequenced in another course. This was directly copied, called the 'original' strand, and was used as a template to compare the experimental sequences to. To ensure that we would be able to read the reverse sequence in the case that the sanger sequence returned the reverse strand, we ensured that the code was able to take the reverse complement of the original strand. The code is able to compare the experimental sequence against the original and reverse template strands and choose the one with a higher LCS.

In terms of the experimental strand, we planned to obtain a sequence from sanger sequencing, however the sequence was mysteriously ~400 bp away from the location our primers were supposed to attach to. To continue with the project, we decided to simulate data using the 500 bp region of DNA previously obtained and convert some of the C → T to simulate partial methylation of the forward strand. This will allow us to run sequences with known methylation states so we can observe whether our code is working properly or not. We also ran some simulated data using the reverse complement of the original sequence to ensure that the alignment code was able to interpret the reverse strand.

A toy sequence was created with 65 possible sites of methylation, 41 of which were methylated, assuming the code runs effectively this should give us ~ 63% methylation overall. The pseudocode includes this sequence. After running the code, the program returned 53.84% methylation, indicating that the alignment removes approximately 6 conserved base pairs otherwise it is able to detect all instances of methylated and unmethylated nucleotides as expected. It also worked with sequences of varying lengths and varying amounts of indels. The results we are reporting are the original sequences, both forward and reverse alignments, their scores and lengths, which one was chosen (forward or reverse), the counts for methylated nucleotides, unmethylated nucleotides, CpG complexes, and methylated CpG complexes. This allows the results to be used to explain what is happening in the DNA at a molecular level by allowing the individual to observe exactly where there were changes to the original sequence and what they were. This can be used to observe the methylation status of any area of DNA as long as there is a control sequence that can be used as the original strand and as long as the sequence isn't longer than 500 bp.

PseudoCode

Main( )

original ←
"GACAATTATAATTCTTCATAAAGTCATGCAATTAATAATAATAATAAAAAAATAATAATTTTCTT
GATTATTGCTCAAAGTGAAAGTCAATCTAAAATTTGTTCGGTTTTTCAATTCTACAATTTTTATTTG
AAAATATATAAATAAAAAAAAATGGTTAATAAAGAATTGCTTAGAGTTTCCTCTTTTTCTAGAACA
AACAAAAATGCTTAGACCTTCTCGAAAGAATAGCACATCACGTTATATCCAAAAGCATATTCTTTT
GTTAGATTAATCGTAGTCATATGGTCATAAGTCATACATAACATGTAACATGTTAGTTCGATCATT
ATTAACTTTCAATTAAGTATTTTCGTAAGACGTAATTTTCAACAATAAACAACAAATCCGCGATAT
ACGGTTTTTGAAAGGGCAGGCAATTGTTTTTTTAAATTATTAATACTATAAAACAATTTTCTTATTG
GCTATAAACTAATATTTAATTCCACGTGGAAGTTA"

experimental ←
"GATAATTATAATTCTTCATAAAGTTATGCAATTAATAATAATAATAATAATAATTTTTTTGATTAA
TTGTTCAAAGTGAAAGTTAATCTAAAATTTGTTCGGTTTTTCAATTTTACAATTTTTATTTGAAAAT
ATATAAAATAAAAAATGGTTAATAAAGAATTGTTTAGAGTTTTTTCTTTTTCTAGAACAAAATGTT
TAGACCTTCTCGAAAGAATAGCACATTACGTTATATTTAAGCATATTCTTTTGTTAGATTAATTGTA
GTCATATGGTTATAAGTCATATATAATATGTAATATGTTAGTTCGATCATTATTAACTTTCAATTAA
GTATTTTTGTAAGATGTAATTTTTAACAATAAACAACAAATCCGCGATATACGGTTTTTGAAAGGG
CAGGCAATT"

Orientation, Original Align, Experimental Align ← Align main(original, experimental)

Methylated count, Possible count, CPG Methylated Count, CPG Possible count ← methyl (Original alignment, Experimental alignment)

Percent score, Percent CPG score ← Percent methyl(Possible Count, Methylated Count, CPG Possible Count, CPG Methylated Count)
return

Align main (original, experimental)

Indel ← -1
Match ← +1
Mismatch ← -1

LeadScore, LeadOriAlign, LeadExpAlign ← Global Align (Original, Experimental, Indel, Match, Mismatch)

LagScore, LagOriAlign, LagExpAlign ← Global Align (Rev comp(Original), Experimental, Indel, Match, Mismatch)

if LagScore > LeadScore
    Orientation ← "Lagging"
    Align Original ← LagOriAlign
    Align ← LagExpAlign

```
    if LagScore < LeadScore
        Orientation ← "Leading"
        Align original ← LeadOriAlign
        Align experimental ← LeadExpAlign

    return Orientation, Original Align, Experimental Align

Methyl(Original, Experimental, Orientation)
    Possible Count ← 0
    Methylated Count ← 0
    CPG Possible Count ← 0
    CPG Methylated Count ← 0

    if Orientation= 'lead'
        nucleotide ← 'C'
        CPG← 'G'
    if Orientation = 'lag'
        nucleotide ← 'G'
        CPG ← 'C'

    for i  ← 0 to |original|
        if original[i] = nucleotide
            Possible Count ← Possible Count + 1
            if experimental[i] = nucleotide
                Methylated Count ← Methylated Count + 1
            if i>0 and original[i-1] = CPG
                CPG Possible Count ← CPG Possible Count+1
                if experimental[i]= nucleotide
                    CPG Methylated Count ← CPG Methylated Count + 1
            if i < |original|-1 and original[i+1] = CPG
                CPG Possible Count ← CPG Possible Count +1
                if experimental[i]= nucleotide
                    CPG Methylated Count ← CPG Methylated Count + 1

    return Methylated count, Possible count, CPG Methylated Count, CPG Possible Count

Percent methyl (Possible Count, Methylated Count,  CPG Possible Count, CPG Methylated Count)
        Percent score ← 0
        Percent CPG score ← 0
        if count>0
            Percentscore  ← Score/count
        if cpgcount>0
            Percent CPG score ←  Score/cpgcount
        return Percent score, Percent CPG score
```

Functions used from previous assignments:

Global Align (sequence 1, sequence 2, indel, match, mismatch) Homework 6.2
Rev comp (sequence) Lab 3.3
Initialize Table (nrows,ncols) Lab 8