Digital Normalization Write-up

– **Compare a fragment recruitment plot made with the normalized read set and the one you made with the full read set. Also, describe the difference in read count of your normalized set both in terms of total count and number/percentage aligned to the genome.**

In the FRP plots of non-normalized reads, the reads align to the reference genome at distinct locations. On the other hand, in the FRP plots of normalized reads, the coverage of the reads against the genome is much greater and spread out, but you can still distinguish the positions present in the FRP plots of non-normalized reads.

In order to increase the accuracy of sequence analysis, we need to detect reads that contain errors. In order to detect these errors, we utilized a digital normalization approach that involves 3 steps: normalizing coverage, eliminating low-abundance kmers, and normalizing the coverage again. The fundamental idea is that random sequencing errors result in random sequences which are not part of the actual genomic sequence. Therefore, it is unlikely that these random sequences will be seen frequently. If we detect these rare sequences in a high-coverage sample, we have good reason to assume that it's an error.

I first ran the following prompt to normalize the read coverage. K-mer size was set to 20, abundance cutoff at 10, and minimum hash size at 1e9. Reads with k-mer coverage higher than 20 would be eliminated and eliminate k-mers with frequency of 10 or lower (e.g. random sequences mentioned above).
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 20 -x 1e9 -s fileaf.kh sample-af.fasta

The prompt below filters out the low-abundance k-mers determined from the previous python script.
> python /diag/software/khmer/scripts/filter-abund.py -C 10 fileaf.kh sample-af.fasta.keep

The last prompt normalizes (by median) the coverage once again.
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 5 -x 1e8 -N 4 sample-af.fasta.keep.abundfilt

The differences in read counts are shown below:
#**Reference genome: NC_014804.fna**
> activelayer-frozen
Raw: 37894 (0.31%)
Normalized: 20276 (34%)

> activelayer-day2
Raw: 39547 (0.34%)
Normalized: 17563 (35%)

> activelayer-day7
Raw: 42759 (0.37%)
Normalized: 17563 (35%)

> permafrost-frozen
Raw: 28330 (0.18%)
Normalized: 48667 (15%)

> permafrost-day2
Raw: 35346 (0.22%)
Normalized: 37583 (21%)

> permafrost-day7
Raw: 37549 (0.24%)
Normalized: 41688 (17%)

#Against NC_007948.fna
> activelayer-frozen
Raw: 14918 (0.12%)
Normalized: 10850 (22%)

> activelayer-day2
Raw: 15904 (0.14%)
Normalized: 12355 (21%)

> activelayer-day7
Raw: 17200 (0.15%)
Normalized: 10850 (22%)

> permafrost-frozen
Raw: 16082 (0.1%)
Normalized: 33847 (10%)

> permafrost-day2
Raw: 17468 (0.11%)
Normalized: 25099 (14%)

> permafrost-day7
Raw: 18318 (0.12%)
Normalized: 28649 (12%)

– **A well-written, documented script that reads a read-to-reference alignment result file and produces a fragment recruitment plot.**

Scripts and command line prompts are recorded below. Plots are attached.

Ex:
> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference.fasta -a sample-af.fasta -o output-frhit-af.txt

Due to the amount of metagenomic reads, I set the sequence identity parameter to 50% (-c 50). The -g parameter calculates the global sequence identity. I ran the non-normalized and normalized data through these new parameters.

reference2.fasta	NC_007948.fna	Polaromonas sp. JS666
reference5.fasta	NC_009659.fna	Janthinobacterium sp. Marseille

#**Convert query FASTQ metagenomic file into FASTA**
> perl fastq2fasta.pl core1_activelayer_day2.fastq sample-a2.fasta

```
> perl fastq2fasta.pl core1_activelayer_day7.fastq sample-a7.fasta
> perl fastq2fasta.pl core1_activelayer_frozen.fastq sample-af.fasta
> perl fastq2fasta.pl core1_permafrost_day2.fastq sample-p2.fasta
> perl fastq2fasta.pl core1_permafrost_day7.fastq sample-p7.fasta
> perl fastq2fasta.pl core1_permafrost_frozen.fastq sample-pf.fasta

#usage: fastq2fasta.pl [sequencing reads.fastq] [output.fasta]
#!/usr/bin/perl -w
use strict;
use Bio::SeqIO;
my ($file1,$file2)=@ARGV;
my $seqin = Bio::SeqIO -> new (-format => 'fastq',-file => $file1);
my $seqout = Bio::SeqIO -> new (-format => 'fasta',-file => ">$file2");
while (my $seq_obj = $seqin -> next_seq)
{
  $seqout -> write_seq($seq_obj);
}
```

#**Digital normalization of metagenomic reads**

```
#sample-af.fasta.keep.abundfilt.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 20 -x 1e9 -s
fileaf.kh sample-af.fasta
> python /diag/software/khmer/scripts/filter-abund.py -C 10 fileaf.kh sample-af.fasta.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 5 -x 1e8 -N 4
sample-af.fasta.keep.abundfilt

#sample-a2.fasta.keep.abundfilt.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 20 -x 1e9 -s file.kh
sample-a2.fasta
> python /diag/software/khmer/scripts/filter-abund.py -C 10 file.kh sample-a2.fasta.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 5 -x 1e8 -N 4
sample-a2.fasta.keep.abundfilt

#sample-a7.fasta.keep.abundfilt.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 20 -x 1e9 -s
filea7.kh sample-a7.fasta
> python /diag/software/khmer/scripts/filter-abund.py -C 10 filea7.kh sample-a7.fasta.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 5 -x 1e8 -N 4
sample-a7.fasta.keep.abundfilt

#sample-pf.fasta.keep.abundfilt.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 20 -x 4e9 -s
filepf.kh sample-pf.fasta
> python /diag/software/khmer/scripts/filter-abund.py -C 10 filepf.kh sample-pf.fasta.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 5 -x 1e8 -N 4
sample-pf.fasta.keep.abundfilt

#sample-p2.fasta.keep.abundfilt.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 20 -x 4e9 -s
filep2.kh sample-p2.fasta
> python /diag/software/khmer/scripts/filter-abund.py -C 10 filep2.kh sample-p2.fasta.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 5 -x 1e8 -N 4
```

sample-p2.fasta.keep.abundfilt

#sample-p7.fasta.keep.abundfilt.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 20 -x 2e9 -N 4 -s filep7.kh sample-p7.fasta
> python /diag/software/khmer/scripts/filter-abund.py -C 10 filep7.kh sample-p7.fasta.keep
> python /diag/software/khmer/scripts/normalize-by-median.py -k 20 -C 5 -x 1e8 -N 4 sample-p7.fasta.keep.abundfilt

#**Convert normalized reads file extension to .fasta for FR-HIT compatibility**
> mv sample-af.fasta.keep.abundfilt.keep sample-af-norm.fasta
> mv sample-a2.fasta.keep.abundfilt.keep sample-a2-norm.fasta
> mv sample-a7.fasta.keep.abundfilt.keep sample-a7-norm.fasta
> mv sample-pf.fasta.keep.abundfilt.keep sample-pf-norm.fasta
> mv sample-p2.fasta.keep.abundfilt.keep sample-p2-norm.fasta
> mv sample-p7.fasta.keep.abundfilt.keep sample-p7-norm.fasta

#**FR-HIT**
#**Align query metagenomic file to reference genome**
#**Usage: frhit –g 1-a [sequencing read.fasta] -d [reference.fasta] -o [alignment output]**
#**-g 1 option outputs global %identity; default = 0 for local %id**
#**Against NC_014804.fna**
> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference2.fasta -a sample-af-norm.fasta -o output2-frhit-af-norm.txt
Total number of reads recruited: 20276 (34%)

> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference2.fasta -a sample-a2-norm.fasta -o output2-frhit-a2-norm.txt
Total number of reads recruited: 17563 (35%)

> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference2.fasta -a sample-a7-norm.fasta -o output2-frhit-a7-norm.txt
Total number of reads recruited: 17563 (35%)

> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference2.fasta -a sample-pf-norm.fasta -o output2-frhit-pf-norm.txt
Total number of reads recruited: 48667 (15%)

> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference2.fasta -a sample-p2-norm.fasta -o output2-frhit-p2-norm.txt
Total number of reads recruited: 37583 (21%)

> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference2.fasta -a sample-p7-norm.fasta -o output2-frhit-p7-norm.txt
Total number of reads recruited: 41688 (17%)

#Against NC_007948.fna
> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference5.fasta -a sample-af-norm.fasta -o output5-frhit-af-norm.txt
Total number of reads recruited: 10850 (22%)

> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference5.fasta -a sample-a2-norm.fasta -o output5-frhit-a2-norm.txt
Total number of reads recruited: 12355 (21%)

> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference5.fasta -a sample-a7-norm.fasta -o output5-frhit-a7-norm.txt
Total number of reads recruited: 10850 (22%)

> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference5.fasta -a sample-pf-norm.fasta -o output5-frhit-pf-norm.txt
Total number of reads recruited: 33847 (10%)

> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference5.fasta -a sample-p2-norm.fasta -o output5-frhit-p2-norm.txt
Total number of reads recruited: 25099 (14%)

> /diag/software/fr-hit-v0.7-x86_64/fr-hit -g 1 -c 50 -d reference5.fasta -a sample-p7-norm.fasta -o output5-frhit-p7-norm.txt
Total number of reads recruited: 28649 (12%)

#**Parse FR-HIT output**
#**Parse through output and print stats out to a single tab-delimited file**
> perl parse_frhit.pl

```
#!/usr/bin/perl
use warnings;
use strict;
my $query;
my $frhitoutput = "output-frhit-af-norm.txt";
#my $frhitoutput = "output-frhit-a2-norm.txt";
#my $frhitoutput = "output-frhit-a7-norm.txt";
#my $frhitoutput = "output-frhit-pf-norm.txt";
#my $frhitoutput = "output-frhit-p2-norm.txt";
#my $frhitoutput = "output-frhit-p7-norm.txt";

my $seq_fh;
#open FR-HIT output file
open($seq_fh, $frhitoutput) || die "failed to read input file: $!";
while (my $line = <$seq_fh>) {
    open(OUT, '>>2norm-output.txt');
    chomp $line;
    next if $line =~ /^#/;
    my @line = split(/\s+/, $line);
    if($line =~ m/\t(\S+)%/){
        #1:af, 2:a2, 3:a7, 4:pf, 5:p2, 6:p7
        #Print out to file: global id \t base pair position, group number
        print OUT "$1\t$line[9]\t6\n";
    }
    close(OUT);
}
```

#**Plotting in R – FRHIT output**
#**Read in tab-delimited file and create matrix**

```
> m <-
read.table("/home/steven/Documents/Metagenomics/Normalization/2norm-output.txt",
sep="\t")

#Split by group number aka day0/2/7
> pf <- m[m[,3]==4, c(1,2,3)]
> froz <- rbind(af,pf)
> a2 <- m[m[,3]==2, c(1,2,3)]
> p2 <- m[m[,3]==5, c(1,2,3)]
> dtwo <- rbind(a2,p2)
> a7 <- m[m[,3]==3, c(1,2,3)]
> p7 <- m[m[,3]==6, c(1,2,3)]
> dsev <- rbind(a7,p7)
#Set base pair position to x
> x.f <- froz[,2]
> x.2 <- dtwo[,2]
> x.7 <- dsev[,2]
#Set percent identity to y
> y.f <- froz[,1]
> y.2 <- dtwo[,1]
> y.7 <- dsev[,1]
#Set percent identity to z
> z.f <- froz[,3]
> z.2 <- dtwo[,3]
> z.7 <- dsev[,3]
#Set colors to each group
> c.f <- cut(z.f, 2, labels = c("black", "red"))
> c.2 <- cut(z.2, 2, labels = c("blue", "green"))
> c.7 <- cut(z.7, 2, labels = c("purple", "pink"))
#Plot
> par(mfrow=c(3,1))
> plot(x.f, y.f, main= "FRP - FR-HIT(normalized): frozen (day 0)\n Reference genome:
Polaromonas sp. JS666 (NC_007948)", ylab = "Percent identity", xlab = "Base pair
position", col = as.character(c.f), pch=20, cex=.05)
> legend("bottomright", c("active frozen", "permafrost frozen"), col=par(as.character(z)), fill
= c("black", "red"), bty="o", bg="white", cex=.8)
> plot(x.2, y.2, main= "FRP - FR-HIT(normalized): day 2\n Reference genome:
Polaromonas sp. JS666 (NC_007948)", ylab = "Percent identity", xlab = "Base pair
position", col = as.character(c.2), pch=20, cex=.05)
> legend("bottomright", c("active day 2", "permafrost day 2"),  col=par(as.character(ztwo)),
fill = c("green", "blue"), bty="o", bg="white", cex=.8)
> plot(x.7, y.7, main= "FRP - FR-HIT(normalized): day 7\n Reference genome:
Polaromonas sp. JS666 (NC_007948)", ylab = "Percent identity", xlab = "Base pair
position", col = as.character(c.7), pch=20, cex=.05)
> legend("bottomright", c("active day 7", "permafrost day 7"), col=par(as.character(zthr)),
fill = c("purple", "pink"), bty="o", bg="white", cex=.8)


> m <-
read.table("/home/steven/Documents/Metagenomics/Normalization/5norm-output.txt
", sep="\t")
#Split by group number aka day0/2/7
```

```
> af <- m[m[,3]==1, c(1,2,3)]
> pf <- m[m[,3]==4, c(1,2,3)]
> froz <- rbind(af,pf)
> a2 <- m[m[,3]==2, c(1,2,3)]
> p2 <- m[m[,3]==5, c(1,2,3)]
> dtwo <- rbind(a2,p2)
> a7 <- m[m[,3]==3, c(1,2,3)]
> p7 <- m[m[,3]==6, c(1,2,3)]
> dsev <- rbind(a7,p7)
#Set base pair position to x
> x.f <- froz[,2]
> x.2 <- dtwo[,2]
> x.7 <- dsev[,2]
#Set percent identity to y
> y.f <- froz[,1]
> y.2 <- dtwo[,1]
> y.7 <- dsev[,1]
#Set percent identity to z
> z.f <- froz[,3]
> z.2 <- dtwo[,3]
> z.7 <- dsev[,3]
#Set colors to each group
> c.f <- cut(z.f, 2, labels = c("black", "red"))
> c.2 <- cut(z.2, 2, labels = c("blue", "green"))
> c.7 <- cut(z.7, 2, labels = c("purple", "pink"))
#Plot
> par(mfrow=c(3,1))
> plot(x.f, y.f, main= "FRP - FR-HIT(normalized): frozen (day 0)\n Reference genome:
Janthinobacterium sp. Marseille (NC_009659)", ylab = "Percent identity", xlab = "Base pair
position", col = as.character(c.f), pch=20, cex=.05)
> legend("bottomright", c("active frozen", "permafrost frozen"), col=par(as.character(z)), fill
= c("black", "red"), bty="o", bg="white", cex=.8)
> plot(x.2, y.2, main= "FRP - FR-HIT(normalized): day 2\n Reference genome:
Janthinobacterium sp. Marseille (NC_009659)", ylab = "Percent identity", xlab = "Base pair
position", col = as.character(c.2), pch=20, cex=.05)
> legend("bottomright", c("active day 2", "permafrost day 2"),  col=par(as.character(ztwo)),
fill = c("green", "blue"), bty="o", bg="white", cex=.8)
> plot(x.7, y.7, main= "FRP – FR-HIT(normalized): day 7\n Reference genome:
Janthinobacterium sp. Marseille (NC_009659)", ylab = "Percent identity", xlab = "Base pair
position", col = as.character(c.7), pch=20, cex=.05)
> legend("bottomright", c("active day 7", "permafrost day 7"), col=par(as.character(zthr)),
fill = c("purple", "pink"), bty="o", bg="white", cex=.8)
```