**Perl script for Contact Map Analysis**

#!/usr/bin/perl

$residuenumber=168;

$nsnapshots=50000;

for ($i=1;$i<$residuenumber;$i++){

for ($j=1;$j<$residuenumber;$j++){

$contact[$i][$j]=0;

}

}

for($k=1;$k<=50000;$k++){

$file="abctmol.pdb.$k";

for ($l=1;$l<=$residuenumber;$l++){

$sumx[$l]=0;

$sumy[$l]=0;

$sumz[$l]=0;

$natom[$l]=0;

}

open (FILE,"<$file"); # read the r.pdb

while (<FILE>) {

chomp;

# print "$\_\n"; #print the line itself

#print "$.\n"; #print the line number; \n means the line finished

@line=split (undef, $\_); #array generate

if ($line[0].$line[1].$line[2].$line[3] eq "ATOM"){

$n=$line[6].$line[7].$line[8].$line[9].$line[10]; #atom number

$pdbatom[$n][1]=$line[32].$line[33].$line[34].$line[35].$line[36].$line[37]; # x axis $pdbatom[$n][2]=$line[40].$line[41].$line[42].$line[43].$line[44].$line[45]; # y axis $pdbatom[$n][3]=$line[48].$line[49].$line[50].$line[51].$line[52].$line[53]; # z axis $pdbatom[$n][4]=$line[22].$line[23].$line[24].$line[25]; # Residue number

$totalatom+=1;

}

}

close FILE;

# Subroutine to calculate the distance between two atoms

sub dist () {

$distance=sqrt(($pdbatom[$\_[0]][1]-$pdbatom[$\_[1]][1])\*\*2+($pdbatom[$\_[0]][2]-$pdbatom[$\_[1]][2])\*\*2+($pdbatom[$\_[0]][3]-$pdbatom[$\_[1]][3])\*\*2);

print "Distance between atom $\_[0] and atom $\_[1] is: $distance\n";

}

############################################################

sub com {

for (my $i=1;$i<=664;$i++){

$sumx[$pdbatom[$i][4]]+=$pdbatom[$i][1]; #xaxis

$sumy[$pdbatom[$i][4]]+=$pdbatom[$i][2]; #yaxis

$sumz[$pdbatom[$i][4]]+=$pdbatom[$i][3]; #zaxis

$natom[$pdbatom[$i][4]]+=1;

}

}

&com;

for ($i=1;$i<$residuenumber;$i++){

for ($j=1;$j<$residuenumber;$j++){

$var1=$i;

$var2=$j;

$x=$sumx[$var1]/$natom[$var1];

$y=$sumy[$var1]/$natom[$var1];

$z=$sumz[$var1]/$natom[$var1];

$x1=$sumx[$var2]/$natom[$var2];

$y1=$sumy[$var2]/$natom[$var2];

$z1=$sumz[$var2]/$natom[$var2];

$distance = sqrt(($x-$x1)\*\*2+($y-$y1)\*\*2+($z-$z1)\*\*2);

if($distance<=6.5)

{

$contact[$i][$j]++;

}

}

}

}

##################################################################

open (RESULT,">contmapabmolCT.dat");

for ($i=1; $i<$residuenumber; $i++) {

for ($j=1; $j<$residuenumber; $j++) {

my $t=$contact[$i] [$j]/$nsnapshots\*100;

print RESULT "$i $j $t\n";

}

}

close RESULT;

**Perl Script for Free Energy Landscape Analysis**

#!/usr/bin/perl

$file="ab42rr.dat";

$nx=200;

$ny=200;

$fileout="pm3dscript1\_d200\_200.dat";

open (IN,"<$file");

$xmax=0;$xmin=1000;

$ymax=0;$ymin=1000;

while (<IN>){

chomp;

my @line=split;

if ($line[0] > $xmax ){$xmax =$line[0];}

if ($line[0] < $xmin ){$xmin =$line[0];}

if ($line[1] > $ymax ){$ymax =$line[1];}

if ($line[1] < $ymin ){$ymin =$line[1];}

}

close IN;

$dx=($xmax-$xmin)/$nx;

$dy=($ymax-$ymin)/$ny;

print "

xmax: $xmax

xmin: $xmin

dx: $dx

ymax: $ymax

ymin: $ymin

dy: $dy

";

$B=0;

open (OUT,">$fileout");

for (my $x=0;$x<$nx;$x++){

for (my $y=0;$y<$ny;$y++){

open (IN,"<$file");

while (<IN>){

chomp;

@line=split;

if ($line[0] >= $xmin+($x\*$dx) && $line[0] < $xmin+($x+1)\*$dx){

if ($line[1] >= $ymin+($y\*$dy) && $line[1] < $ymin+($y+1)\*$dy){

$B++;

}

}

}

close IN;

printf OUT ("%10.8f %10.8f %5s\n", $xmin+$x\*$dx, $ymin+$y\*$dy, $B);

$B=0;

}

print OUT "\n";

}

close OUT;

#########################################

#!/usr/bin/perl

$max\_population=276;

$T=300;

open (PO,"<pm3dscrip2\_200\_200.dat");

while (<PO>){

chomp;

@line=split;

if (/^\s\*$/){print "\n";}

elsif ($line[2] eq 0) {printf ("%10.4f %10.4f %10.4f\n",$line[0],$line[1],-0.0019872041\*$T\*(log(0.0001)-log($max\_population)));}

else {printf ("%10.4f %10.4f %10.4f\n",$line[0],$line[1],-0.0019872041\*$T\*(log($line[2])-log($max\_population)));}

}

close PO;