Starling-May18
Projects/Katarina Stuart/KStuart.Starling-Aug18/At1_Genome/Figures/2023-02-07.Circos

PDF Version generated by

Katarina Stuart (z5188231@ad.unsw.edu.au)

on

Aug 23, 2023 @09:45 AM NZST

Table of Contents

Circos Plot for *A. tristis*

Genome scaffold lengths

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos
```

```
GENOME=/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/analysis/curation/step4_scaffolding/ragtag_atris_synteny/renamed/AcTris_vAus2.0.fasta
```

```
module load SAMtools/1.16.1-GCC-11.3.0
```

```
samtools faidx $GENOME
```

```
cut -f1,2 /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/analysis/curation/step4_scaffolding/ragtag_atris_synteny/renamed/AcTris_vAus2.0.fasta.fai > sizes.genome
```

```
module load BEDTools/2.30.0-GCC-11.3.0
WIDTH=1000000
```

```
bedtools makewindows -g sizes.genome -w ${WIDTH} > atristis_${WIDTH}bps.bed
head -n 37 sizes.genome | awk -v OFS=',' '{print $1, 1, $2 , 0, "gneg"}' > cyto_columns.txt
cat cyto_header.txt cyto_columns.txt > cyto_init.csv
```

Variant Density Track

```
module load BEDTools/2.30.0-GCC-11.3.0
```

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos
VCF=/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/analysis/recombination/LDhat/vcf_split/recombination_IND_MP_TN.recode.vcf
```

```
bedtools coverage -a atristis_${WIDTH}bps.bed -b $VCF -counts > variantcoverage_${WIDTH}bps.txt
sed 's/\t/,/g' variantcoverage_${WIDTH}bps.txt > variantcoverage_${WIDTH}bps.csv
```

Repeat Content

```
module load BEDTools/2.30.0-GCC-11.3.0
```

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos
```

```
REPEATS=/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/analysis/repeats/repeatmasker/AcTris_vAus2.0repeatlib_hardmask/AcTris_vAus2.0.fasta.out
awk -v OFS="\t" '$1=$1' ${REPEATS}| cut -f5,6,7 | tail -n +3 > repeat_masker.bed
```

```
bedtools coverage -a atristis_${WIDTH}bps.bed -b repeat_masker.bed -counts > repeats_${WIDTH}bps.txt
sed 's/\t/,/g' repeats_${WIDTH}bps.txt > repeats_${WIDTH}bps.csv
```

Methylation

```
module load BEDTools/2.30.0-GCC-11.3.0
```

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos
```

```
METH=/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/analysis/methylation/merged/AcTris_vAus_ALL_modbam2bed.out

#for info on input file column values refer to modbam2bed output
#create column with % of reads methylated fo each site
#filter for CPG sites with some methylation, and for sites with minimum methylated+unmethylated reads of 5
awk '$13>0' ${METH} | awk '($13+$12)>5' | awk -v OFS="\t" '{print $0, $13 / ($12+$13) }'  > methylation_CPGsite_density.txt

#pull out all CPG sites with methylation, and those with 80%+ methylation
awk '$16>0.75 '  methylation_CPGsite_density.txt | awk -v OFS="\t" '{print $1,$2,$3}' > methylation_CPG_density_10.bed
awk -v OFS="\t" '{print $1,$2,$3}'  methylation_CPGsite_density.txt > methylation_CPG_density_100.bed

#all of this is needed for the overlap visualisation to make sure the grey bit captures top 10%, and white bit the rest of the 80% quantile
bedtools coverage -a atristis_${WIDTH}bps.bed -b methylation_CPG_density_10.bed -counts > methylation10sites_${WIDTH}bps.txt
bedtools coverage -a atristis_${WIDTH}bps.bed -b methylation_CPG_density_100.bed -counts > methylation100sites_${WIDTH}bps.txt

paste methylation100sites_1000000bps.txt <(cut -f4 methylation10sites_1000000bps.txt) > methylationBOTHsites_1000000bps.txt
awk -v OFS="\t" '{print $1, $2, $3, $4, $4-$5}'  methylationBOTHsites_1000000bps.txt >  methylationBOTHsites2_1000000bps.txt
awk '{print $1,$2,$3,($4>20000)? 20000: $4,($5>20000)? 20000: $5}' methylationBOTHsites2_1000000bps.txt > methylationBOTHsites3_1000000bps.txt  #this line for maxing out peak valu
sed 's/ /,/g'  methylationBOTHsites3_1000000bps.txt > methylationBOTHsites_1000000bps.csv
```

Variant Density Track

```
module load BEDTools/2.30.0-GCC-11.3.0
```

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos
RECOMB=/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/analysis/recombination/LDhat/IND_chroms/Superscaffold_chr4A/recombination_IND_MP_TN.Superscaffold_chr4A
```

```
bedtools coverage -a atristis_${WIDTH}bps.bed -b $VCF -counts > variantcoverage_${WIDTH}bps.txt
sed 's/\t/,/g' variantcoverage_${WIDTH}bps.txt > variantcoverage_${WIDTH}bps.csv
```

GFF Annotation

```
module load BEDTools/2.30.0-GCC-11.3.0
```

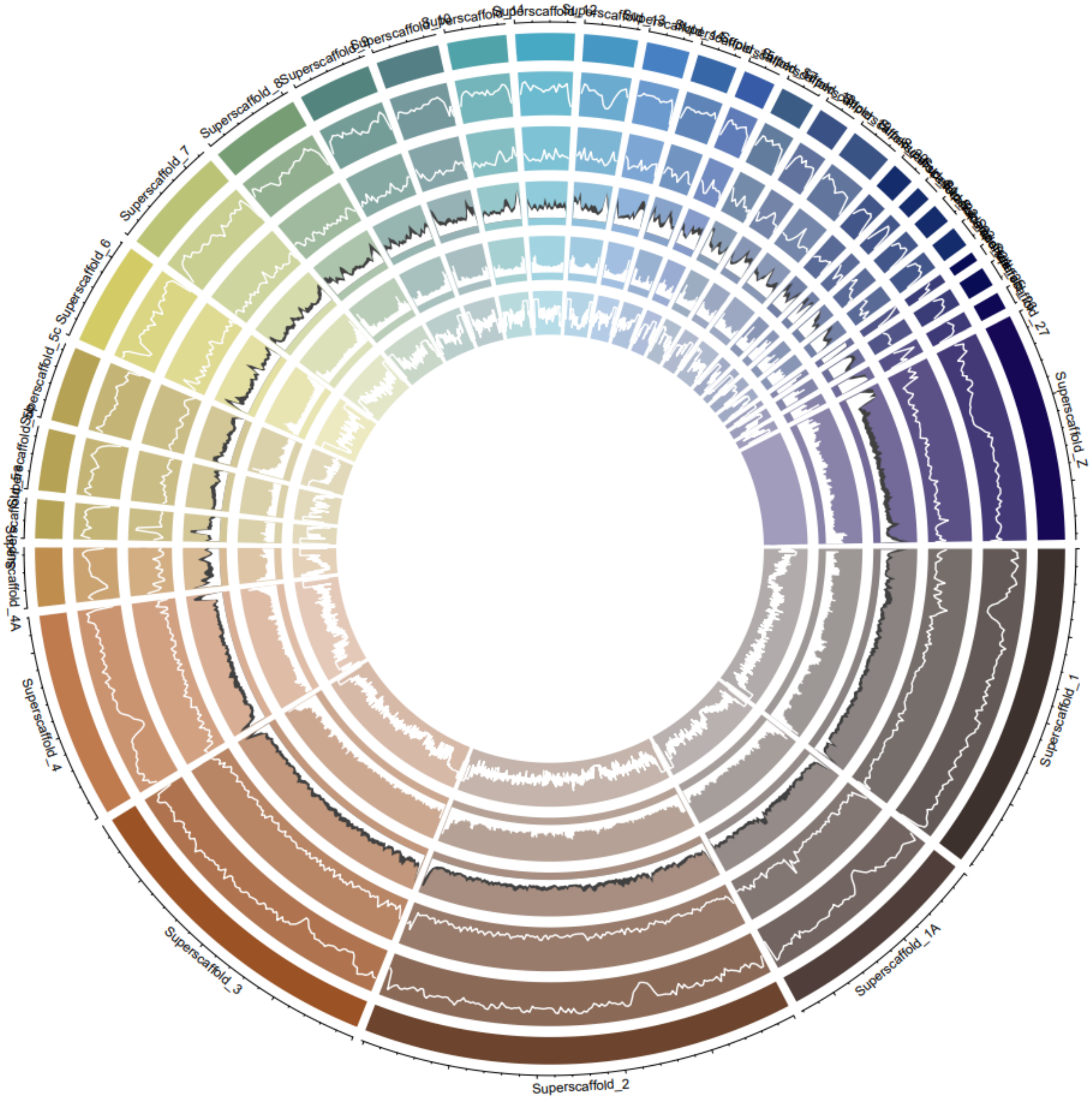
```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos
```

```
GFF=/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/tsebra/braker_gemoma_combined_katmanual_brakerforce.gtf
bedtools coverage -a atristis_${WIDTH}bps.bed -b <(awk '$3=="gene"' $GFF) -counts > annotationcoverage_${WIDTH}bps.txt
sed 's/\t/,/g' annotationcoverage_${WIDTH}bps.txt > annotationcoverage_${WIDTH}bps.csv
```

Plot

```
module load R/4.1.0-gimkl-2020a
R
setwd("/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos")
#install.packages("circlize")
library(circlize)

cytoband.df = read.csv("cyto_init.csv", colClasses = c("character", "numeric", "numeric", "character", "character"), sep = ",", na.strings='NULL')
str(cytoband.df)

vd.df = read.csv("variantcoverage_1000000bps.csv", sep = ",", na.strings='NULL',header=FALSE )
head(vd.df)

#mt.df = read.csv("methylationBOTH_1000000bps.csv", sep = ",", na.strings='NULL',header=FALSE )
mt.df = read.csv("methylationBOTHsites_1000000bps.csv", sep = ",", na.strings='NULL',header=FALSE )

rp.df = read.csv("repeats_1000000bps.csv", sep = ",", na.strings='NULL')

ts.df = read.csv("repeat_masker.bed.csv", sep = ",", na.strings='NULL')

rc.df2 = read.csv("/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/analysis/recombination/LDhat/IND_chroms/recombination_rho_allchroms_thin100.txt", sep = ",",
na.strings='NULL',header=FALSE )
rc.df2$V5 <- log(rc.df2$V4+1) #log transform the data
rc.df<-rc.df2[,c(1:3,5)]

gn.df = read.csv("annotationcoverage_1000000bps.csv", sep = ",", na.strings='NULL',header=FALSE )



pal <- c("#3D312D", "#4f3e39", "#6d442d", "#9b5224", "#BF7A4E","#BF8D4E", "#b6a254","#b6a254" ,"#b6a254" ,"#D3CC66", "#BBC476","#779D75", "#53857e","#537F85", "#50A3A9", "#4
"#4797C4", "#4781C4", "#3767a7","#375BA7","#3A5C85", "#3a5185", "#3A5385","#142c6c","#142c6c", "#142c6c", "#142c6c","#080C55","#080C55", "#0E0855", "#160855")
```

```
pal20 <- add_transparency(pal, transparency = 0.2)
pal30 <- add_transparency(pal, transparency = 0.3)
pal40 <- add_transparency(pal, transparency = 0.4)
pal50 <- add_transparency(pal, transparency = 0.5)
pal60 <- add_transparency(pal, transparency = 0.6)


pdf("At1_circular_plot.pdf", width=15, height=15)

#initialise
circos.initializeWithIdeogram(cytoband.df,
chromosome.index = paste0("Superscaffold_chr", c(1,"1A",2,3,4,"4A","5a","5b","5c",6,7,8,9,10,11,12,13,14,15,17,18,19,20,21,22,23,24,25,26,27,"Z")),
                labels.cex = 1, axis.labels.cex = 0.01)

#band colours and thick band
circos.track(ylim = c(0, 1), panel.fun = function(x, y) {
  chr = CELL_META$sector.index
  xlim = CELL_META$xlim
  ylim = CELL_META$ylim},
bg.col = pal,
  track.height = 0.05, bg.border = NA)

#SNP track #or manually alter in to the area under ??
circos.genomicTrackPlotRegion(vd.df, bg.col = pal20, ylim = c(0, 35000),
                panel.fun = function(region, value, ...) {
                  circos.genomicLines(region, value, ...,  lwd = 2 , col = "#FFFFFF")
                }, track.height = 0.08,bg.border = NA)

#Genes
circos.genomicTrackPlotRegion(gn.df, bg.col = pal30, ylim = c(0, 100),
                panel.fun = function(region, value, ...) {
                  circos.genomicLines(region, value, ...,  lwd = 2 , col = "#FFFFFF")
                }, track.height = 0.08,bg.border = NA)

# Track methylation
circos.genomicTrack(mt.df, bg.col = pal40, ylim = c(0, 20000), panel.fun = function(region, value, ...) {
  circos.genomicLines(region, value, ..., col = c("#404040", "#FFFFFF"), lwd = 0.1 , area = TRUE ,  border = NA )
}, track.height = 0.08, bg.border = NA)

#repeat track
circos.genomicTrackPlotRegion(rp.df, bg.col = pal50, ylim = c(0, 1500),
                panel.fun = function(region, value, ...) {
                  circos.genomicLines(region, value, ..., type ='s', area = TRUE, col = "#FFFFFF", border = NA)
                }, track.height = 0.08, bg.border = NA )

#Recombination track
circos.genomicTrackPlotRegion(rc.df, bg.col = pal60,
                panel.fun = function(region, value, ...) {
                  circos.genomicLines(region, value, ...,  lwd = 2 , col = "#FFFFFF")
                }, track.height = 0.08,bg.border = NA)


dev.off()
circos.clear()




###OLD TRACKS

#SNP track
circos.genomicTrackPlotRegion(vd.df, bg.col = pal20, ylim = c(0, 25000),
                panel.fun = function(region, value, ...) {
                  circos.genomicLines(region, value, ..., area = TRUE, col = "#FFFFFF", border = "#FFFFFF",)
                }, track.height = 0.12,bg.border = NA)

circos.genomicTrackPlotRegion(vd.df, ylim = c(0, 24900),
                panel.fun = function(region, value, ...) {
                  circos.genomicLines(region, value, ..., area = TRUE, col = "#F2D7D5")
                }, track.height = 0.10)



# Track Repeats
circos.genomicDensity(ts.df, bg.col = pal50, window.size = 5e5,
            track.height = 0.08, col = "white", bg.border = NA)
```



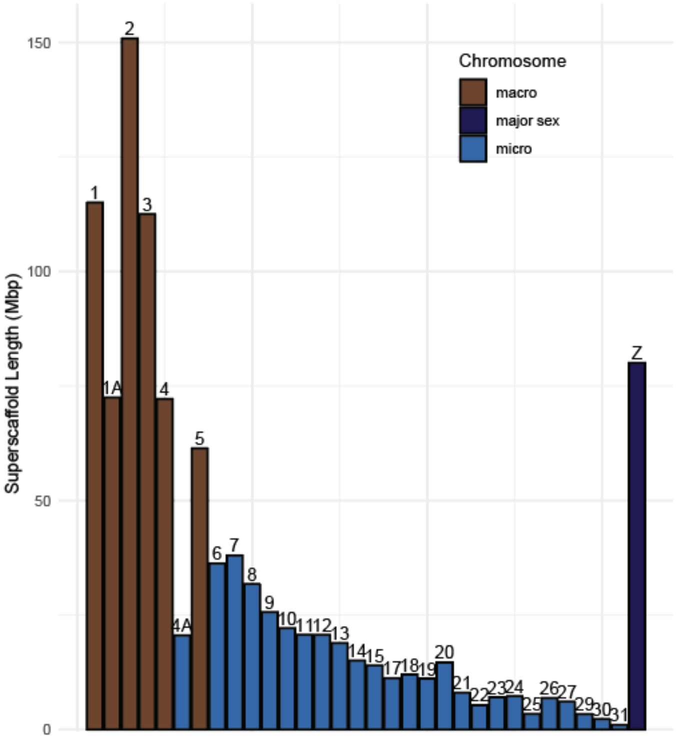
Chrom Sizes: micro and macro

```
sed -e 's/Superscaffold_chr5a/Superscaffold_chr5/g' -e 's/Superscaffold_chr5b/Superscaffold_chr5/g' -e 's/Superscaffold_chr5c/Superscaffold_chr5/g' sizes.genome > sizes.genome.plotting

sed 's/Superscaffold_chr//g'  sizes.genome.plotting.manual > sizes.genome.plotting.manual2
```

```
module load R/4.1.0-gimkl-2020a
R
library("ggplot2")
library("dplyr")

setwd("/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos")

genome<- read.table("sizes.genome.plotting.manual2", header=FALSE)
genome2<-genome[c(1:17,19:32,35),]

genome2$order <- seq.int(nrow(genome2))

genome2$lab <- c(rep("macro",5),"micro","macro",rep("micro",24),"major sex")

genome2$length <- genome2$V2/1000000

#for use eslewhere
as.data.frame( genome2 %>% group_by(lab) %>% summarise(sum = sum(V2), n = n()) )
```

```
 lab sum n
1     macro 584476776  6
2 major sex  80029777  1
3 micro 363460020 25
```

```
#plot

pdf("At1_genome_chroms.pdf")
ggplot(genome2, aes(x=order, y=length, fill=lab))+geom_bar(stat="identity", color="black")+scale_fill_manual(values=c("#6d442d", "#160855","#3767a7"))+ theme_minimal()+geom_text(aes
position=position_dodge(width=0.9), vjust=-0.25)+ylab("Superscaffold Length (Mbp)")+xlab("")
dev.off()
```

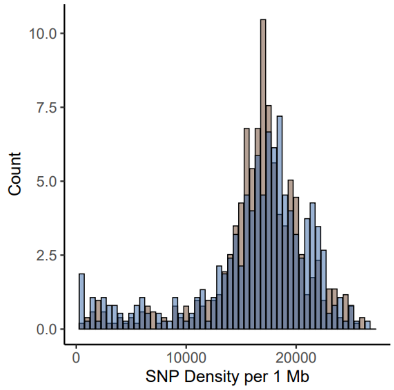


Panel A

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos/panels
```

```
WIDTH=1000000
grep -w -F -f macro_superscaffolds.txt ../variantcoverage_${WIDTH}bps.txt > variantcoverage_${WIDTH}bps_macro.txt
grep -w -F -f micro_superscaffolds.txt ../variantcoverage_${WIDTH}bps.txt > variantcoverage_${WIDTH}bps_micro.txt
```

```
module load R/4.1.0-gimkl-2020a
R
library("ggplot2")
library(dplyr)
setwd("/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos/panels")

macro.density <- read.table(file="variantcoverage_1000000bps_macro.txt", header=FALSE, sep="\t")
macro.density <-macro.density %>% mutate(windows = cut(V4, breaks=seq(0, 27000, 500)))
macro_counts <- macro.density %>% count(windows, .drop=FALSE)
macro_counts2 <- as.data.frame( cbind(macro_counts$n,  seq(500, 27000, 500) ) )
macro_counts2 <- macro_counts2 %>% mutate(corrected = (V1 / 516 )*100)  # sum(macro_counts2$corrected) is 100

micro.density <- read.table(file="variantcoverage_1000000bps_micro.txt", header=FALSE, sep="\t")
micro.density <-micro.density %>% mutate(windows = cut(V4, breaks=seq(0, 27000, 500)))
micro_counts <- micro.density %>% count(windows, .drop=FALSE)
micro_counts2 <- as.data.frame( cbind(micro_counts$n,  seq(500, 27000, 500) ) )
micro_counts2 <- micro_counts2 %>% mutate(corrected = (V1 / 375) *100)  # sum(micro_counts2$corrected) is 100

macro_counts2$chrom <- c("macro")
micro_counts2$chrom <- c("micro")

chrom_density <- rbind(macro_counts2[,c(2,3,4)],micro_counts2[,c(2,3,4)])

pdf("At1_snpdensitybins.pdf", width=6, height=6)
ggplot() + geom_col(data=macro_counts2, aes(x = V2, y = corrected, fill=chrom), fill='#6d442d', color="black", alpha = 0.5) +
geom_col(data=micro_counts2, mapping=aes(x = V2, y = corrected, fill=chrom), fill='#3767a7', color="black", alpha = 0.5) +
ylab("Count Proportion") + xlab("SNP Density per 1 Mb") + theme_classic(base_size = 18) + theme(axis.text=element_text(size=16))
dev.off()
```

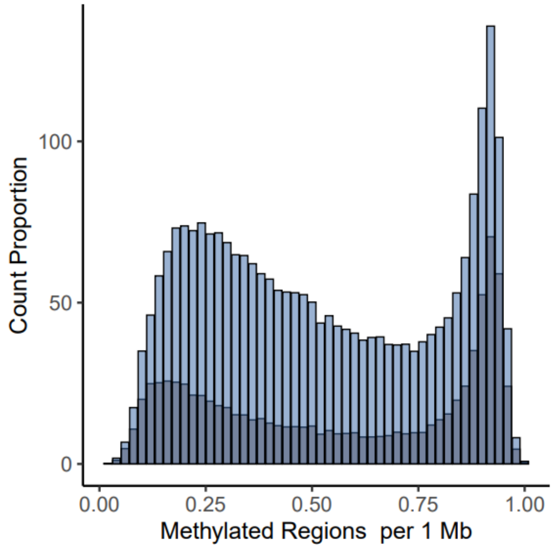


Panel B

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos/panels
```

```
METH=/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/analysis/methylation/merged/methylation_regions_density.bed

grep -w -F -f macro_superscaffolds.txt $METH > methylation_CPGsite_macro.txt
grep -w -F -f micro_superscaffolds.txt $METH > methylation_CPGsite_micro.txt

#working with regions, not sites

module load R/4.1.0-gimkl-2020a
R
library("ggplot2")
library("dplyr")
setwd("/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos/panels")

macro.meth <- read.table(file="methylation_CPGsite_macro.txt", header=FALSE, sep="\t")
#work out for each window which 'break' group is belongs to
macro.meth <-macro.meth %>% mutate(windows = cut(V7, breaks=seq(0, 1, 0.02)))
#count across groups
macro.meth_counts <- macro.meth %>% count(windows, .drop=TRUE)
#assign 'break' group labels to the new data frame
macro.meth_counts2 <- as.data.frame( cbind(macro.meth_counts$n,  seq(0.02, 1, 0.02) ) )
#correct the total counts of each group so the y-axis is scaled based on how much of the genome the micro vs marco chroms cover
macro.meth_counts2 <- macro.meth_counts2 %>% mutate(corrected = (V1 / 58.448 ))   # sum(macro.meth_counts2$corrected) is 100
```

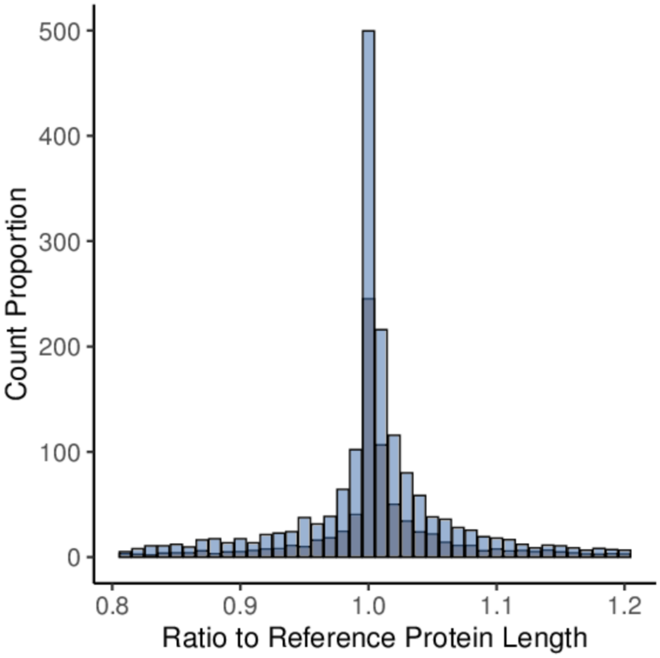
```
micro.meth <- read.table(file="methylation_CPGsite_micro.txt", header=FALSE, sep="\t")
#work out for each window which 'break' group is belongs to
micro.meth <-micro.meth %>% mutate(windows = cut(V7, breaks=seq(0, 1, 0.02)))
#count across groups
micro.meth_counts <- micro.meth %>% count(windows, .drop=FALSE)
#assign 'break' group labels to the new data frame
micro.meth_counts2 <- as.data.frame( cbind(micro.meth_counts$n,  seq(0.02, 1, 0.02) ) )
#correct the total counts of each group so the y-axis is scaled based on how much of the genome the micro vs marco chroms cover
micro.meth_counts2 <- micro.meth_counts2 %>% mutate(corrected = (V1 / 36.346 ))   # sum(micro.meth_counts2$corrected) is 100

#correction vals grabbed from above section Chrom Sizes: micro and macro. e.g,. for macro 584476776 + micro 363460020)

macro.meth_counts2$chrom <- c("macro")
micro.meth_counts2$chrom <- c("micro")

pdf("At1_methylation.pdf", width=6, height=6)
ggplot() + geom_col(data=macro.meth_counts2, aes(x = V2, y = corrected, fill=chrom), fill='#6d442d', color="black", alpha = 0.5) +
geom_col(data=micro.meth_counts2, mapping=aes(x = V2, y = corrected, fill=chrom), fill='#3767a7', color="black", alpha = 0.5) +
ylab("Count Proportion") + xlab("Methylated Regions  per 1 Mb") + theme_classic(base_size = 18) + theme(axis.text=element_text(size=16))
dev.off()

CHECK GENES
```



Panel F

```
cd /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos/panels
```

```
grep -w -F -f macro_superscaffolds.txt <(grep "ENSG" /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/saaga/saaga.proteins.tdt) > saaga_macro.txt
grep -w -F -f micro_superscaffolds.txt <(grep "ENSG" /nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/annotation/saaga/saaga.proteins.tdt) > saaga_micro.txt


module load R/4.1.0-gimkl-2020a
R
library("ggplot2")
library("dplyr")
setwd("/nesi/nobackup/uoa02613/kstuart_projects/At1_MynaGenome/figures/circos/panels")


macro.saaga <- read.table(file="saaga_macro.txt", header=FALSE, sep="\t")
macro.saaga  <-macro.saaga %>% mutate(windows = cut(V16, breaks=seq(0.8, 1.2, 0.01)))
macro.saaga_counts <- macro.saaga  %>% count(windows, .drop=FALSE)
macro.saaga_counts2 <- as.data.frame( cbind(macro.saaga_counts$n,  seq(0.81, 1.2, 0.01) ) )
macro.saaga_counts2  <- macro.saaga_counts2  %>% mutate(corrected = (V1 / 58.448))   # sum(macro.saaga_counts2$corrected) is 100

micro.saaga <- read.table(file="saaga_micro.txt", header=FALSE, sep="\t")
micro.saaga  <-micro.saaga %>% mutate(windows = cut(V16, breaks=seq(0.8, 1.2, 0.01 )))
micro.saaga_counts <- micro.saaga  %>% count(windows, .drop=FALSE)
micro.saaga_counts2 <- as.data.frame( cbind(micro.saaga_counts$n,  seq(0.81, 1.2, 0.01) ) )
micro.saaga_counts2  <- micro.saaga_counts2  %>% mutate(corrected = (V1 /36.346 ))   # sum(macro.saaga_counts2$corrected) is 100

#correction vals grabbed from above section Chrom Sizes: micro and macro. e.g,. for macro 584476776 + micro 363460020)

macro.saaga_counts2$chrom <- c("macro")
micro.saaga_counts2$chrom <- c("micro")

pdf("At1_saaga.pdf", width=6, height=6)
ggplot() + geom_col(data=macro.saaga_counts2[1:40,], aes(x = V2, y = corrected, fill=chrom), fill='#6d442d', color="black", alpha = 0.5) +
geom_col(data=micro.saaga_counts2[1:40,] , mapping=aes(x = V2, y = corrected, fill=chrom), fill='#3767a7', color="black", alpha = 0.5) +
ylab("Count Proportion") + xlab("Ratio to Reference Protein Length") + theme_classic(base_size = 18) + theme(axis.text=element_text(size=16))
dev.off()
```


Count Proportion
Ratio to Reference Protein Length