all_strain_genome_stats

2023-07-12

Assembly stats for 13 Emiliania huxleyi genomes

```
genstats <- list()
gc <- list()
stats <- list()
strains <- as.character(read.csv('data/strains.csv',header=FALSE))
for (strain in strains){
  genstats[[strain]] <- read.table(paste0("data/stats_after_pilon_round_2_decontam_rmdup/"
                                          ,strain,".genstats.txt"),sep="\t",header=F)

  colnames(genstats[[strain]]) <- c(    'Contig_name','Avg_fold',   'Length',   'Ref_GC',
                                    'Covered_percent', 'Covered_bases','Plus_reads',
                                    'Minus_reads', 'Read_GC',  'Median_fold', 'Std_Dev')

    gc[[strain]] <- read.table(paste0("data/stats_after_pilon_round_2_decontam_rmdup/"
                                      ,strain,".gcscaffold.txt"),sep="\t",header=F)
  colnames(gc[[strain]]) <- c('Contig_name',    'Length',   'A',    'C',    'G',    'T',    'N',
                              'IUPAC','Other',  'GC')
print(length(rownames(genstats[[strain]])))
    #genstats[[strain]]$Contig_name <- str_split(genstats[[strain]]$Contig_name,regex("_1$"),simplify =

  stats[[strain]] <- inner_join(genstats[[strain]],gc[[strain]],by="Contig_name")
  print(length(rownames(stats[[strain]])))

}
```

Read in assembly stats data

```
## [1] 3749
## [1] 3749
## [1] 7179
## [1] 7179
## [1] 8513
## [1] 8513
## [1] 6028
## [1] 6028
## [1] 383
## [1] 383
## [1] 7788
## [1] 7788
## [1] 9374
## [1] 9374
```

```
## [1] 2174
## [1] 2174
## [1] 9915
## [1] 9915
## [1] 8474
## [1] 8474
## [1] 4042
## [1] 4042
## [1] 1770
## [1] 1770
## [1] 8184
## [1] 8184
```

```
n50s <- c()
lengths <- c()
contig_counts <- c()
mins <- c()
maxs <- c()
l50s <- c()
for (strain in strains){
contig_lengths <- stats[[strain]]$Length.x
contig_counts <- c(contig_counts,length(contig_lengths))
total_assembled_length=sum(contig_lengths)
contig_lengths <- sort(contig_lengths,decreasing=TRUE)

sum <- 0
count <- 0
for (length in contig_lengths){
  sum <- sum+length
  count <- count+1
  if (sum>=total_assembled_length/2){
    l50s <- c(l50s,count)
    n50s <- c(n50s,as.numeric(length))
    lengths <- c(lengths,total_assembled_length)
    maxs <- c(maxs,contig_lengths[1])
    mins <- c(mins,contig_lengths[length(contig_lengths)])
    break
  }
}

}
global_stats <- data.frame(n50s,l50s,lengths,contig_counts,mins,maxs)
rownames(global_stats) <- strains
colnames(global_stats) <- c('N50','L50',  'Total assembled length','Contig count',
                            'Min contig length','Max contig length')
```

Calculate globals assembly stats including N50 and total assembled length

```
name_translation <- read.table('data/genomescope/illumina-run-conversions.txt',sep='')
temp <- name_translation$V2
names(temp) <- name_translation$V1
name_translation <- temp
```

Add in estimated genome size stats, calculated using Genomescope

```
global_stats$genome_haploid_length <- seq(1,nrow(global_stats))
global_stats$genome_unique_length <- seq(1,nrow(global_stats))
for (folder in list.dirs('data/genomescope/')){
  if (grepl('HA',folder)){
  key <- str_split(folder,'_',simplify=TRUE)[,1]
  key <- str_split(key,'/',simplify=TRUE)[,4]
  if (sum(grepl(name_translation[key],rownames(global_stats)))==1){
  temp <- read.csv(paste0(folder,"/summary.txt_fixed.csv"))
  global_stats[name_translation[key],'genome_haploid_length'] <-
    mean(temp[2,'min'],temp[2,'max'])
  global_stats[name_translation[key],'genome_unique_length'] <-
    mean(temp[4,'min'],temp[4,'max'])
  }
  }
}
#print(temp)
print(global_stats)
```

```
##              N50  L50 Total assembled length Contig count Min contig length
## CCMP371   175654  255              178311892         3749               671
## CCMP375    87072  535              184716260         7179               505
## CCMP377    78015  871              252888127         8513               493
## CCMP1280  113435  361              176639739         6028               510
## RCC874   3369442   15              152306371          383               522
## RCC914     77641  462              147848163         7788               508
## RCC1222    55027  677              145640905         9374               160
## RCC1239   370740  138              170164584         2174               506
## RCC1256    44965 1147              198119043         9915               393
## RCC3492   106475  366              171317621         8474               496
## RCC3963   158060  295              183446030         4042               544
## RCC6071   546189  101              192603649         1770               545
## RCC6856    83052  795              238298182         8184               528
##          Max contig length genome_haploid_length genome_unique_length
## CCMP371            1476961              98806191             72440020
## CCMP375            1947962             103218139             69423426
## CCMP377             606376             138388017             76045285
## CCMP1280           4348365             111401730             64523269
## RCC874             7853977             118376237             91959391
## RCC914              653764              86242315             53701350
## RCC1222            1451258             107508536             66175986
```

```
## RCC1239            1376033            116873978            83660661
## RCC1256             579311             93127971            69271484
## RCC3492            1245722             77328042            62355972
## RCC3963            1638904            107212419            77275117
## RCC6071            2885505            131960871            81986586
## RCC6856             509900            123410186            76317769
```

```
global_stats$Strain <- rownames(global_stats)
ggplot(global_stats,aes(x=N50,y=genome_haploid_length,label=Strain))+
  geom_point()+
  theme_bw()+
  scale_x_log10()+
  geom_smooth(method=lm,colour="black")+
  ylab("Haploid genome length")+
  xlab("Contig N50")+
  labs(caption=str_wrap("Figure 1.  Relationship between predicted haploid genome
                        length and contig N50 for 13 Emiliania huxleyi
                        genomes.",75))+
    geom_text_repel(size=3)+
  theme(plot.caption = element_text(hjust = 0,size=12))
```

Genomescope predicted haploid genome length vs N50

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: The following aesthetics were dropped during statistical transformation: label
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a `group` aesthetic or to convert a numerical
##   variable into a factor?
```

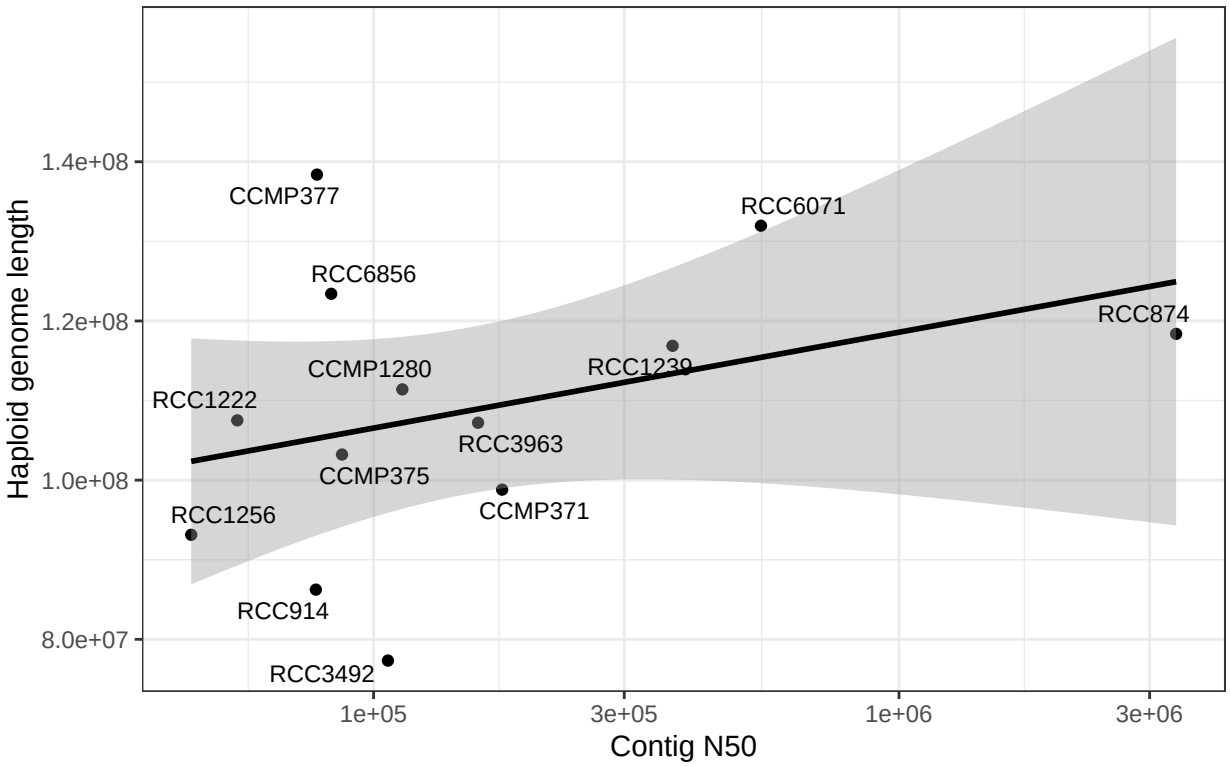

Figure 1. Relationship between predicted haploid genome length and contig N50 for 13 Emiliania huxleyi genomes.

```
ggsave("plots/haploid_genome_length_vs_N50.png")
```

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: The following aesthetics were dropped during statistical transformation: label
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a `group` aesthetic or to convert a numerical
##   variable into a factor?
```

```
global_stats$Strain <- rownames(global_stats)
ggplot(global_stats,aes(x=N50,y=genome_unique_length,label=Strain))+
  geom_point()+
  theme_bw()+
  scale_x_log10()+
  geom_smooth(method=lm,colour="black")+
  ylab("Unique genome length")+
  xlab("Contig N50")+
  labs(caption=str_wrap("Figure 2.  Relationship between predicted unique genome
                        length and contig N50 for 13 Emiliania huxleyi
                        genomes.",75))+
```

```
    geom_text_repel(size=3)+
  theme(plot.caption = element_text(hjust = 0,size=12))
```

Genomescope predicted unique genome length vs N50

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: The following aesthetics were dropped during statistical transformation: label
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a `group` aesthetic or to convert a numerical
##   variable into a factor?
```

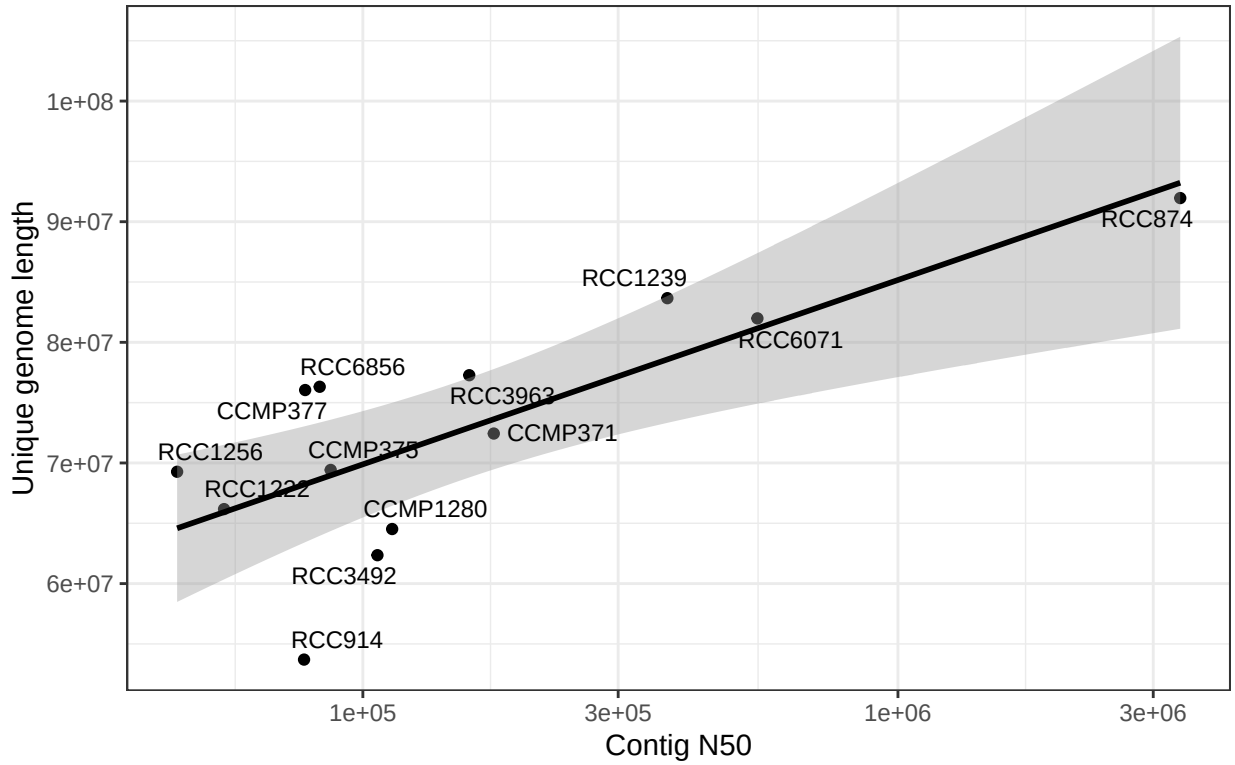

Figure 2. Relationship between predicted unique genome length and contig N50 for 13 Emiliania huxleyi genomes.

```
ggsave("plots/unique_genome_length_vs_N50.png")
```

```
## Saving 6.5 x 4.5 in image
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: The following aesthetics were dropped during statistical transformation: label
## i This can happen when ggplot fails to infer the correct grouping structure in
##   the data.
## i Did you forget to specify a `group` aesthetic or to convert a numerical
##   variable into a factor?
```

```
stats_joined <- ldply(stats,rbind)
ggplot(stats_joined,aes(x=Length.x))+geom_histogram()+scale_x_log10()+
          theme_bw()+
  xlab("Contig size")+
  ylab("Count")+facet_wrap(vars(.id))+
  labs(caption=str_wrap("Figure 3. Contig length distribution for 13 Emiliania
                        huxleyi genomes.",75))+
  theme(plot.caption = element_text(hjust = 0,size=12))
```

Contig length distribution for each strain

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

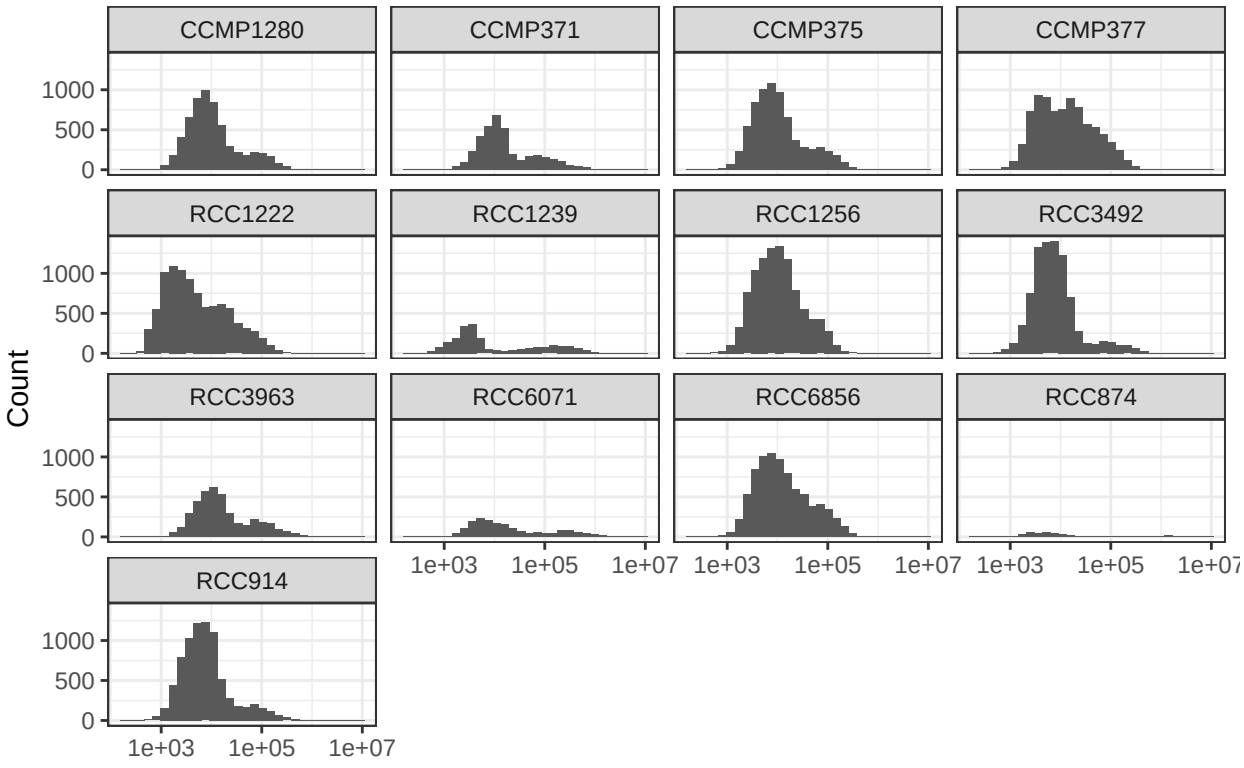

Figure 3. Contig length distribution for 13 Emiliania huxleyi genomes.

```
ggsave("plots/contig_length_distributions.png")
```

```
## Saving 6.5 x 4.5 in image
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
ggplot(data = stats_joined, aes(x = GC,y=Avg_fold)) +
  geom_point(size=0.5)+
```

```
  ylim(0,300)+
  #scale_y_log10()+
  facet_wrap(vars(.id))+
             theme_bw()+
  xlab("Contig GC%")+
  ylab("Average Fold Coverage")+
  labs(caption=str_wrap("Figure 4. Contig Fold coverage vs GC percentage for 13
                        Emiliana huxleyi genomes.",75))+
  theme(plot.caption = element_text(hjust = 0,size=12))
```

Scatterplots of fold coverage vs GC percentage (each point represents a contig)

```
## Warning: Removed 174 rows containing missing values (‘geom_point()‘).
```

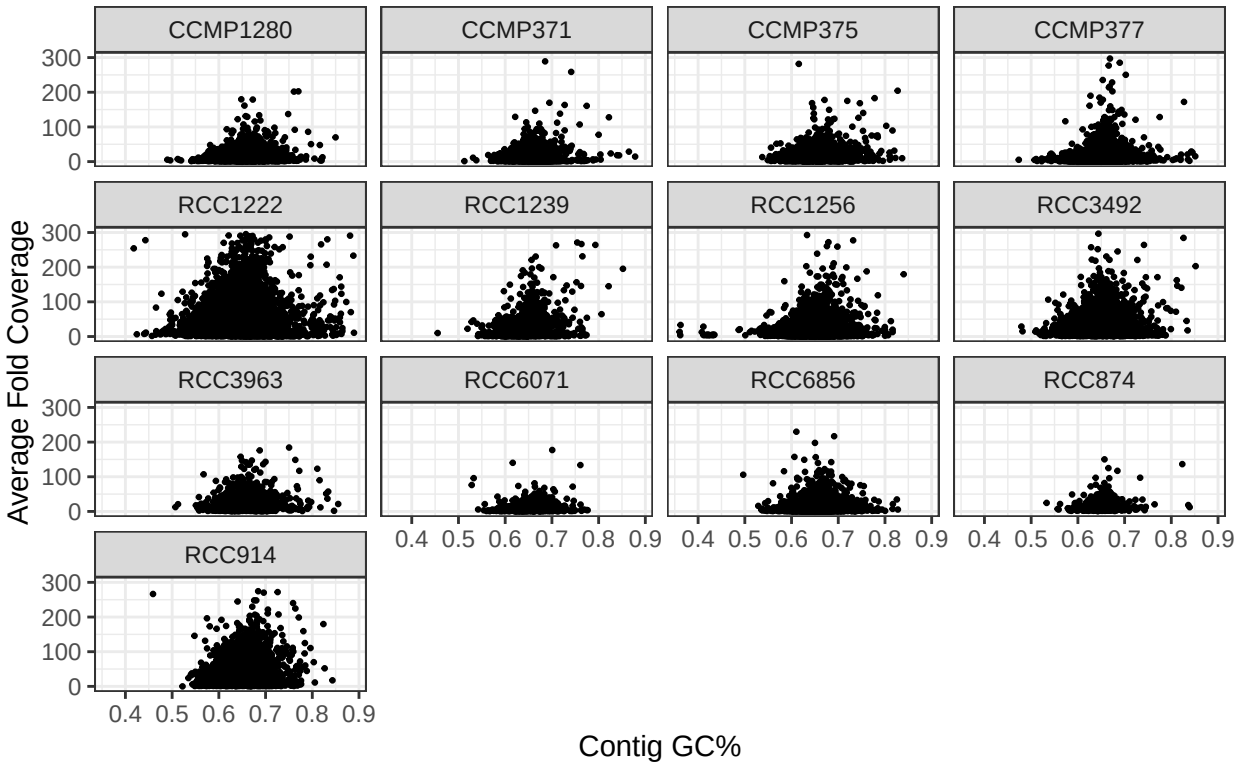

Figure 4. Contig Fold coverage vs GC percentage for 13 Emiliana huxleyi genomes.

```
ggsave("plots/coverage_vs_GC.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
## Warning: Removed 174 rows containing missing values (‘geom_point()‘).
```

```
global_stats$total_assembled_length <- global_stats[,"Total assembled length"]
ggplot(global_stats,aes(x=total_assembled_length,y=genome_unique_length,label=Strain))+
  geom_point()+
  theme_bw()+
  geom_abline(color="black",slope=1,linetype = "dashed")+
    geom_abline(color="black",slope=0.5,linetype = "dashed")+
  xlim(0,3e+8)+
  ylim(0,1.5e+8)+
  ylab("Unique genome length")+
  xlab("Total assembled length")+
  labs(caption=str_wrap("Figure 5.  Relationship between predicted unique genome
                        length and total assembled length for 13 Emiliania huxleyi
                        genomes.",75))+
    geom_text_repel(size=3)+
    annotate("text",x=1e8,y=1e8,label="y=x",color="red",size=5)+
      annotate("text",x=2.5e8,y=1e8,label="y=0.5x",color="red",size=5)+
  theme(plot.caption = element_text(hjust = 0,size=12))
```

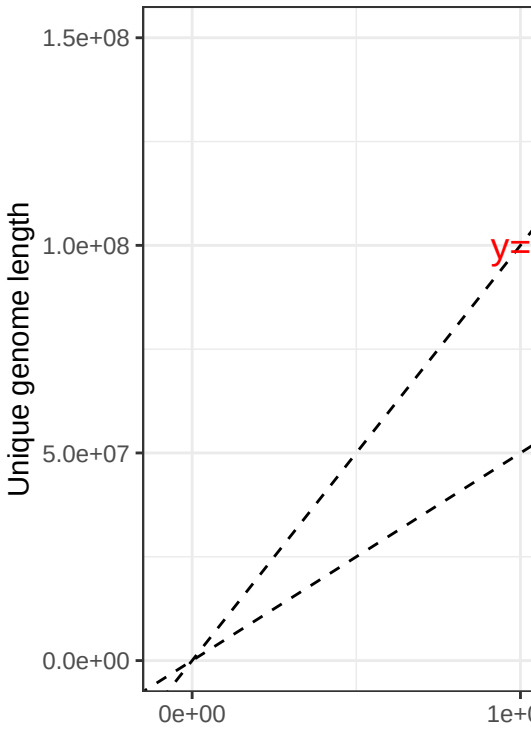

Figure 5. Relationship betw
assembled length for 13 Em

Genomescope predicted unique genome length vs assembled length

```
ggsave("plots/unique_genome_length_vs_total_assembled_length.png")
```

```
## Saving 6.5 x 4.5 in image
```

```
ggplot(global_stats,aes(x=total_assembled_length,y=genome_haploid_length,label=Strain))+
  geom_point()+
```

```
  theme_bw()+
  geom_abline(color="black",slope=1,linetype = "dashed")+
    geom_abline(color="black",slope=0.5,linetype = "dashed")+
  xlim(0,3.5e+8)+
  ylim(0,2e+8)+
  ylab("Haploid genome length")+
  xlab("Total assembled length")+
  labs(caption=str_wrap("Figure 6.  Relationship between predicted haploid genome
                        length and total assembled length for 13 Emiliania huxleyi
                        genomes.",75))+
    geom_text_repel(size=3)+
  annotate("text",x=2e8,y=1.8e8,label="y=x",color="red",size=5)+
    annotate("text",x=3e8,y=1.7e8,label="y=0.5x",color="red",size=5)+
  theme(plot.caption = element_text(hjust = 0,size=12))
```

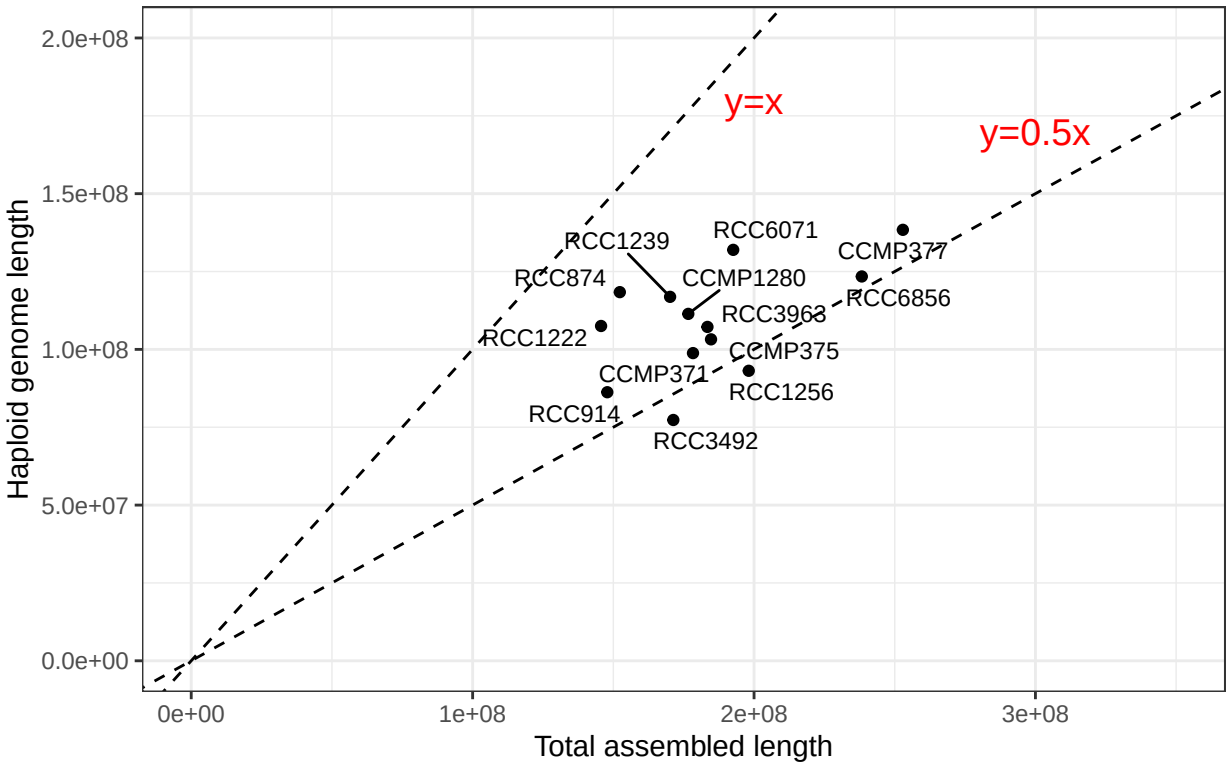

Figure 6. Relationship between predicted haploid genome length and total assembled length for 13 Emiliania huxleyi genomes.

```
ggsave("plots/haploid_genome_length_vs_total_assembled_length.png")
```

```
## Saving 6.5 x 4.5 in image
```