Class 17 Plots

Trinity Lee A16639698

November 28, 2023

```
file_path <- "mm-second.x.zebrafish.tsv"

# Read the TSV file into a data frame
data <- read.table(file_path, header = TRUE, sep = "\t")
```

```
## Warning in scan(file = file, what = what, sep = sep, quote = quote, dec = dec,
## : number of items read is not a multiple of the number of columns
```

```
# Display the first few rows of the data
head(data)
```

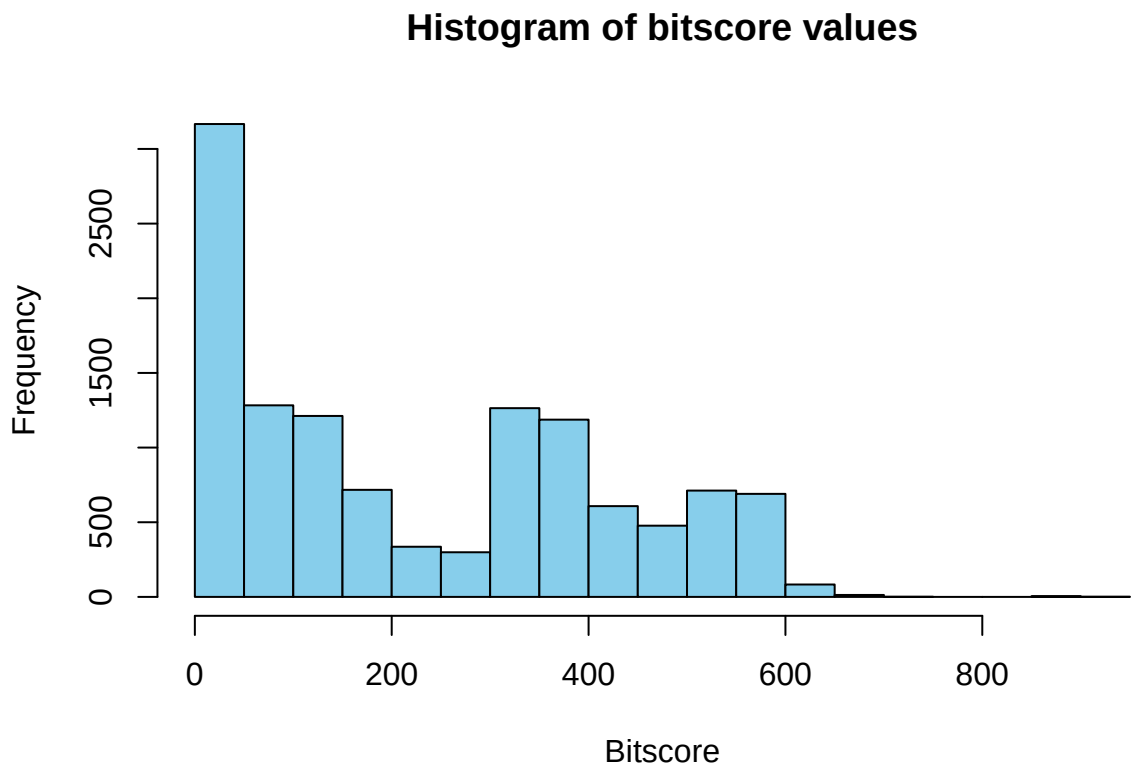
```
##   NP_598866.1 XP_009294521.1 X46.154 X273 X130 X6 X4 X267 X420 X684 X1.70e.63
## 1 NP_598866.1 NP_001313634.1  46.154  273  130  6  4  267  476  740  4.51e-63
## 2 NP_598866.1 XP_009294513.1  46.154  273  130  6  4  267  475  739  4.69e-63
## 3 NP_598866.1 NP_001186666.1  33.071  127   76  5  4  126  338  459  5.19e-12
## 4 NP_598866.1 NP_001003517.1  30.400  125   82  4  4  126  344  465  2.67e-11
## 5 NP_598866.1 NP_001003517.1  30.645   62   41  2 53  113   43  103  4.40e-01
## 6 NP_598866.1    NP_956073.2  34.444   90   56  3 40  126  527  616  1.70e-10
##    X214
## 1 214.0
## 2 214.0
## 3  67.8
## 4  65.5
## 5  33.9
## 6  63.2
```

```
colnames(data) <- c("qseqid", "sseqid", "pident", "length", "mismatch", "gapopen", "qstart", "qend", "ss

# Display the first few rows of the data
head(data)
```

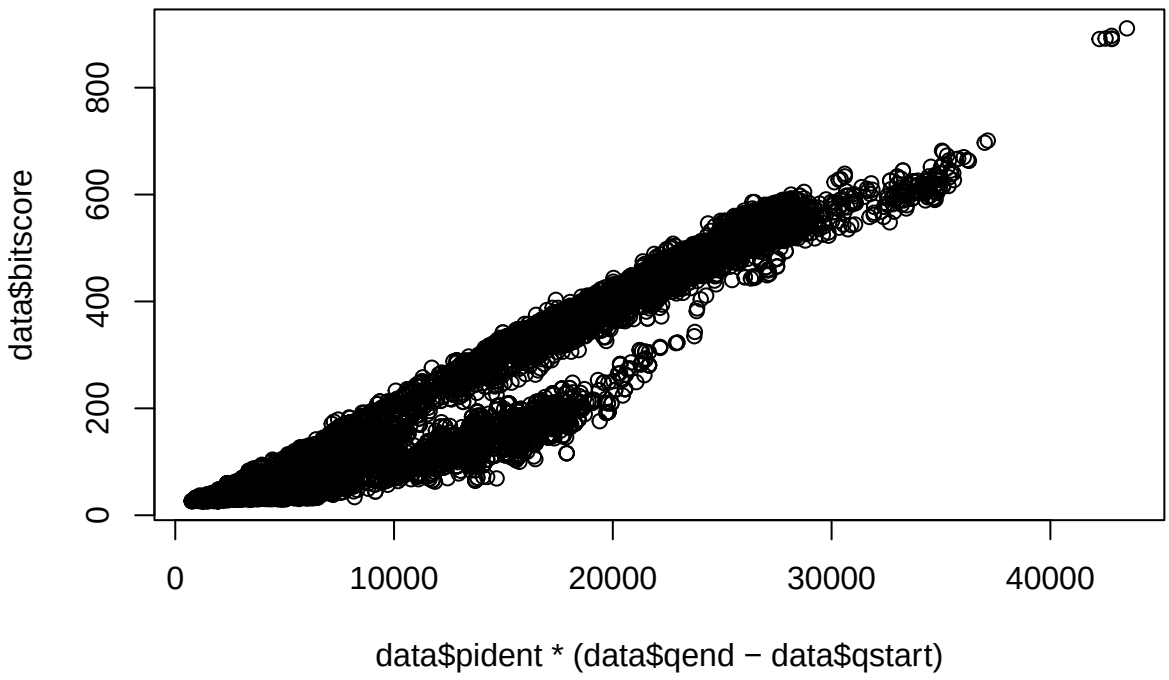
```
##        qseqid         sseqid pident length mismatch gapopen qstart qend sstart
## 1 NP_598866.1 NP_001313634.1 46.154    273      130       6      4  267    476
## 2 NP_598866.1 XP_009294513.1 46.154    273      130       6      4  267    475
## 3 NP_598866.1 NP_001186666.1 33.071    127       76       5      4  126    338
## 4 NP_598866.1 NP_001003517.1 30.400    125       82       4      4  126    344
## 5 NP_598866.1 NP_001003517.1 30.645     62       41       2     53  113     43
## 6 NP_598866.1    NP_956073.2 34.444     90       56       3     40  126    527
##   send   evalue bitscore
## 1  740 4.51e-63    214.0
## 2  739 4.69e-63    214.0
## 3  459 5.19e-12     67.8
## 4  465 2.67e-11     65.5
## 5  103 4.40e-01     33.9
```

```
## 6  616 1.70e-10     63.2
```

```r
bitscore_values <- data$bitscore

# Create a histogram with breaks=30
hist(bitscore_values, breaks = 30, col = "skyblue", main = "Histogram of bitscore values",
     xlab = "Bitscore", ylab = "Frequency")
```

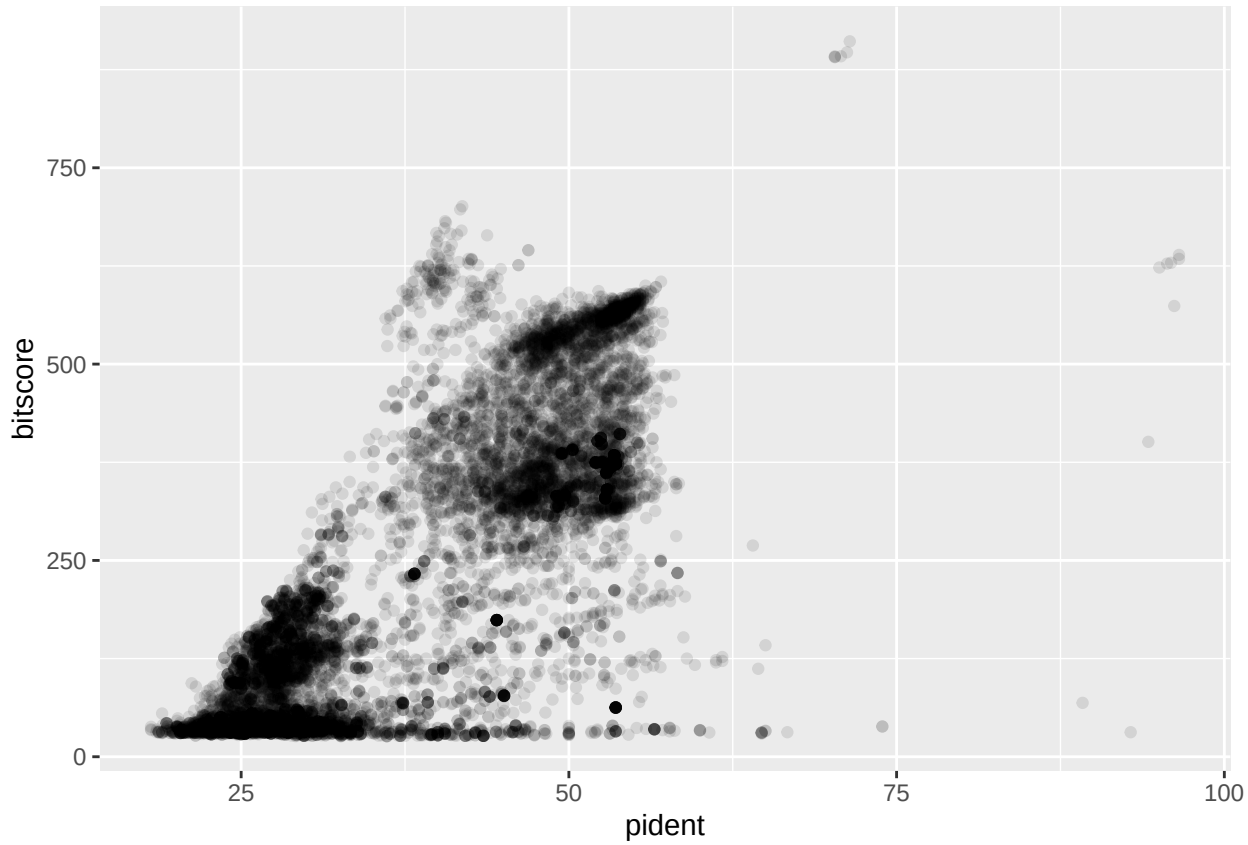



```r
plot(data$pident  * (data$qend - data$qstart), data$bitscore)
```




```
library(ggplot2)
ggplot(data, aes(pident, bitscore)) + geom_point(alpha=0.1)
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```




```
ggplot(data, aes((data$pident * (data$qend - data$qstart)), bitscore)) + geom_point(alpha=0.1) + geom_sm
```

```
## Warning: Use of `data$pident` is discouraged.
## i Use `pident` instead.
```

```
## Warning: Use of `data$qend` is discouraged.
## i Use `qend` instead.
```

```
## Warning: Use of `data$qstart` is discouraged.
## i Use `qstart` instead.
```

```
## Warning: Use of `data$pident` is discouraged.
## i Use `pident` instead.
```

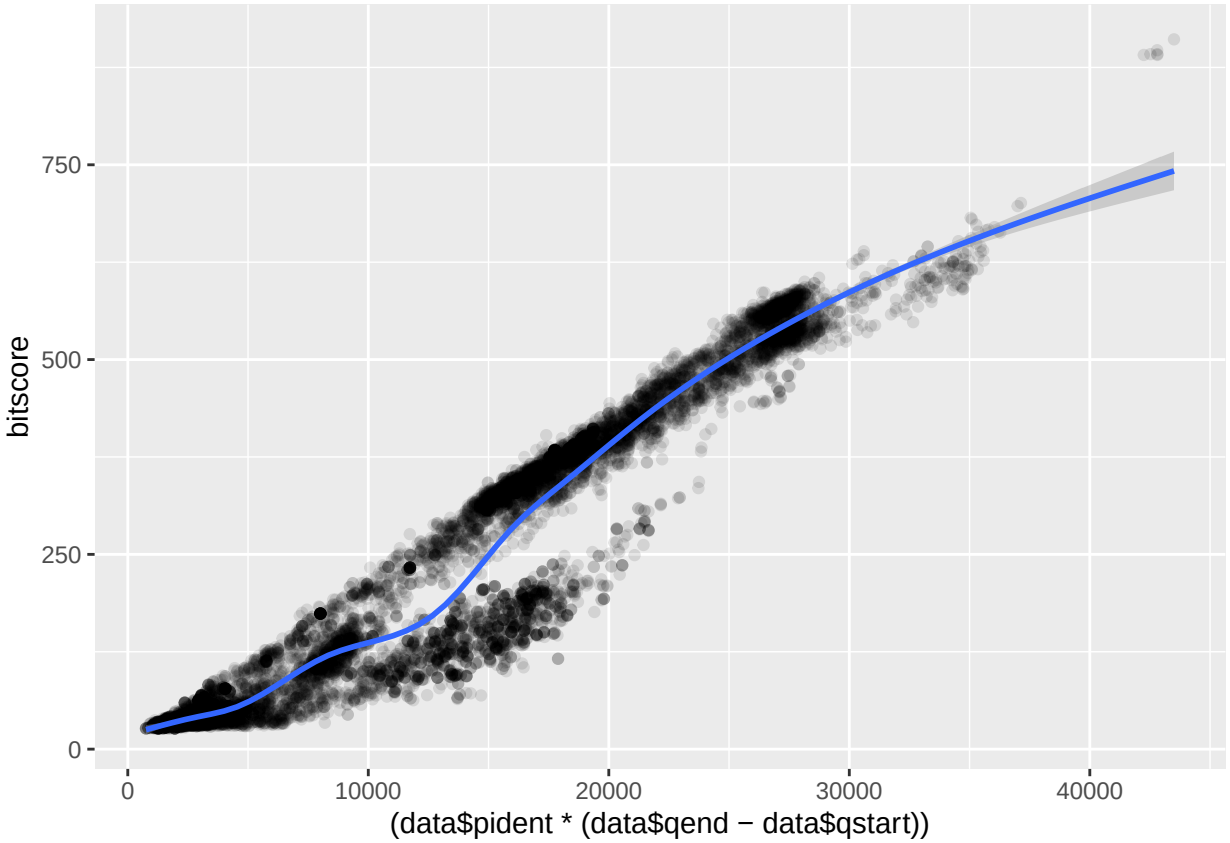
```
## Warning: Use of `data$qend` is discouraged.
## i Use `qend` instead.
```

```
## Warning: Use of `data$qstart` is discouraged.
## i Use `qstart` instead.
```

```
## `geom_smooth()` using method = 'gam' and formula = 'y ~ s(x, bs = "cs")'
```

```
## Warning: Removed 1 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 1 rows containing missing values (`geom_point()`).
```


bitscore
750
500
250
0
0
10000
20000
30000
40000
(data$pident * (data$qend − data$qstart))