PepBed examples (Mouse)

November 19, 2017

Parsing bigbed files

```
library(PepBed)

# path to bigbed file(s)
bb_path <- '/home/enrique/temp/mouse/pride_cluster_peptides_10090_Mouse_pogo.bb'
bb_mod_path <- '/home/enrique/temp/mouse/pride_cluster_peptides_10090_Mouse_pogo_ptm.bb'

# convert bigbed to bed file (output bed file in the same directory)
bigbed2bed(inputFile = bb_path, compress = FALSE)
bigbed2bed(inputFile = bb_mod_path, compress = FALSE)

# getting basic information (output description file in the same directory)
getBigBedInfo(inputFile = bb_path)
getBigBedInfo(inputFile = bb_mod_path)

# getting field names if available
fieldNames <- getBigBedFieldNames(inputFile = bb_path, only.names = TRUE)

print(fieldNames)
```

```
##  [1] "chrom"       "chromStart"  "chromEnd"    "name"        "score"
##  [6] "strand"      "thickStart"  "thickEnd"    "reserved"    "blockCount"
## [11] "blockSizes"  "chromStarts"
```

Parsing Bed file

```
# path to bed file(s)
bed_path <- '/home/enrique/temp/mouse/pride_cluster_peptides_10090_Mouse_pogo.bed'
bed_mod_path <- '/home/enrique/temp/mouse/pride_cluster_peptides_10090_Mouse_pogo_ptm.bed'

# import bed file as  dataframe
bed_df <- readBedFile(inputFile = bed_path)
bed_mod_df <- readBedFile(inputFile = bed_mod_path)

# set column name to bed file
names(bed_df) <- fieldNames
names(bed_mod_df) <- fieldNames

# convert dataframe to GRanges
# all non-modified peptides
granges_peptide <- buildGRangesFromData(data = bed_df,
                                        chrColName = "chrom",
                                        startColName = "chromStart",
                                        endColName = "chromEnd")

# all modified peptides
granges_mod_peptide <- buildGRangesFromData(data = bed_mod_df,
                                            chrColName = "chrom",
                                            startColName = "chromStart",
                                            endColName = "chromEnd")
```

Computing some basic stats from the data

```
# getting number of features(peptides) by chromosome
counts <- countsByChromosome(gr = granges_peptide, colName = 'Peptides')
counts_mod <- countsByChromosome(gr = granges_mod_peptide, colName = 'Peptides_mod')

# merging dfs
merged_counts <- merge.data.frame(counts, counts_mod, by = 'Chromosome')
# ordering by chromosome
merged_counts <- orderByChromosome(df = merged_counts, colName = 'Chromosome')

print(merged_counts)
```

```
##    Chromosome Peptides Peptides_mod
## 1           1     2403          898
## 12          2     2914         1105
## 13          3     1960          742
## 14          4     2382         1020
## 15          5     2336         1208
## 16          6     2124          952
## 17          7     2613         1069
## 18          8     1653          685
## 19          9     2195          933
## 2          10     1926          735
## 3          11     4776         2046
## 4          12     1619          688
## 5          13     2347         1029
## 6          14     1818          700
## 7          15     2135          937
## 8          16      986          366
## 9          17     1814          799
## 10         18     1050          461
## 11         19     1932          706
## 21          X     1298          549
## 22          Y       58           34
## 20          M       20            7
```

Getting stats for unique peptides

```
# removing duplicated entries from original granges_peptide
unique_pep <- getUniqueFeatures(granges_peptide, colFeatures = 'name')
unique_pep_mod <- getUniqueFeatures(granges_mod_peptide, colFeatures = 'name')

# getting unique number of features(peptides) by chromosome
counts_unique <- countsByChromosome(gr = unique_pep, colName = 'Peptides')
counts_mod_unique <- countsByChromosome(gr = unique_pep_mod, colName = 'Peptides_mod')

# merging dfs
merged_counts_unique <- merge.data.frame(counts_unique,
                                         counts_mod_unique,
                                         by = 'Chromosome')

# ordering by chromosome
merged_counts_unique <- orderByChromosome(df = merged_counts_unique,
                                          colName = 'Chromosome')

print(merged_counts_unique)
```

```
##    Chromosome Peptides Peptides_mod
## 1           1     2232          782
## 12          2     2290          713
## 13          3     1429          467
## 14          4     1703          643
## 15          5     1660          644
## 16          6     1484          506
## 17          7     2011          760
## 18          8     1207          348
## 19          9     1619          621
## 2          10     1681          625
## 3          11     3797         1470
## 4          12     1305          504
## 5          13     1060          372
## 6          14     1447          531
## 7          15     1515          546
## 8          16      863          285
## 9          17     1271          489
## 10         18      834          307
## 11         19     1423          420
## 21          X      910          329
## 20          M       20            7
```

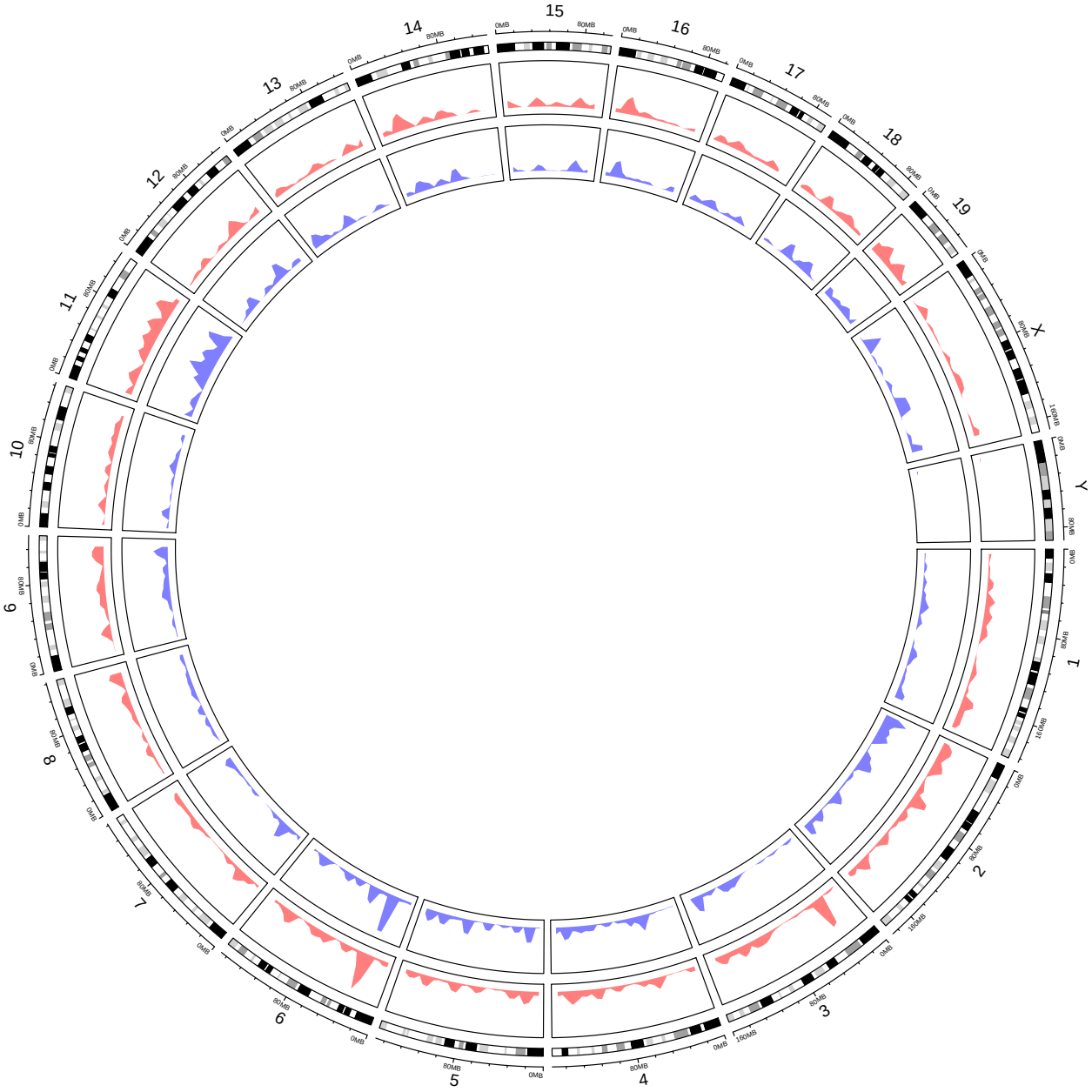
Computing % coverage

```
## compute coverage of query (peptide evidences) on subject (transcripts) by crhomosome
data("protein_coding_transcript_mm10") # load protein coding transcript as GRanges object

coverage <- computeCoverageByChromosome(query = granges_peptide,
                                        subject = transcripts_mm10,
                                        colName = 'Coverage')

coverage_mod <- computeCoverageByChromosome(query = granges_mod_peptide,
                                            subject = transcripts_mm10,
                                            colName = 'Coverage_mod')

# merging dfs
merged_coverage <- merge.data.frame(coverage, coverage_mod, by = 'Chromosome')

# ordering by chromosome
merged_coverage <- orderByChromosome(df = merged_coverage, colName = 'Chromosome')

print(merged_coverage)
```

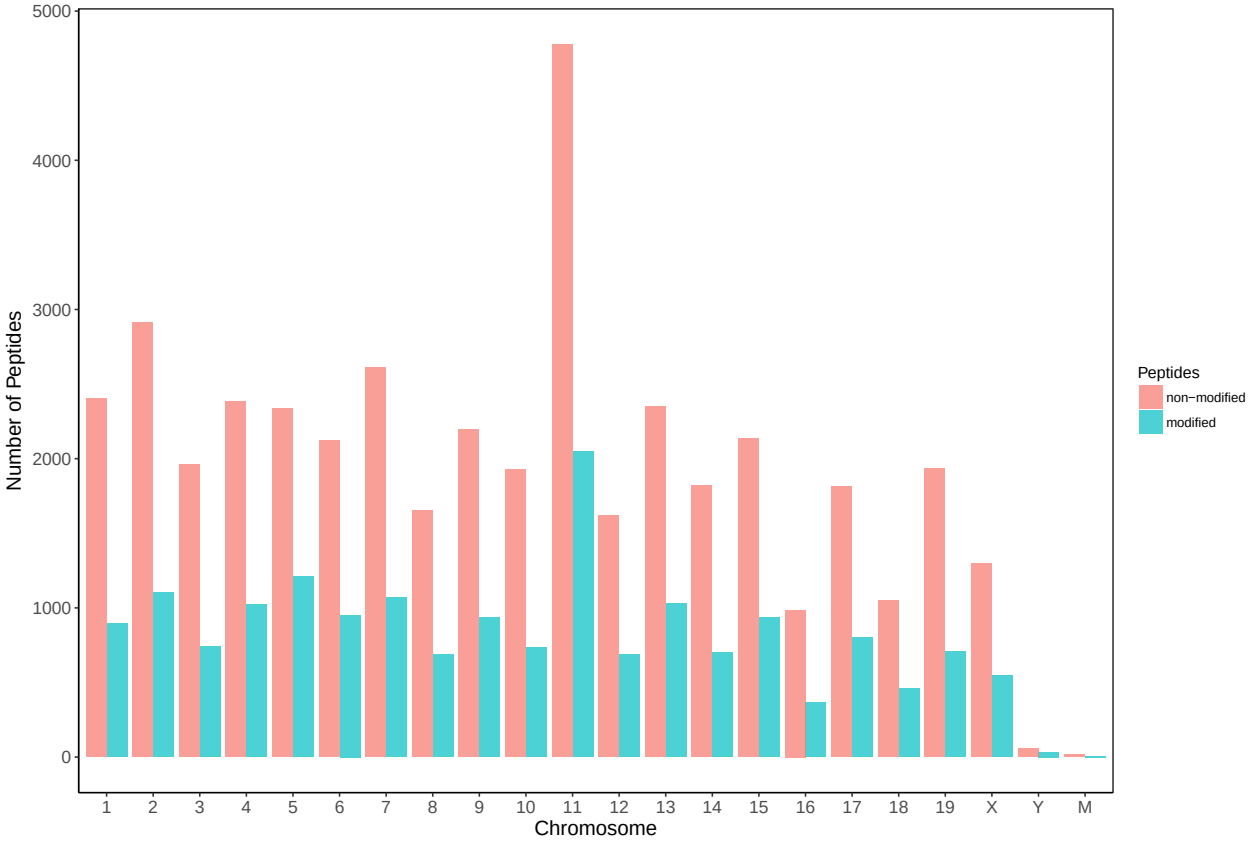
```
##    Chromosome Coverage Coverage_mod
## 1           1   40.872       40.596
## 12          2   42.836       42.224
## 13          3   40.543       39.848
## 14          4   36.648       35.801
## 15          5   38.697       38.117
## 16          6   41.383       41.489
## 17          7   40.177       39.785
## 18          8   35.847       34.937
## 19          9   33.356       33.101
## 2          10   31.369       30.935
## 3          11   34.397       34.236
## 4          12   36.366       36.066
## 5          13   28.457       27.412
## 6          14   33.502       32.295
## 7          15   27.665       26.862
## 8          16   37.278       36.890
## 9          17   26.583       26.082
## 10         18   34.695       34.167
## 11         19   33.501       31.681
## 21          X   38.360       38.157
## 22          Y   29.430       29.290
## 20          M    6.262        3.184
```

Visualizing the data

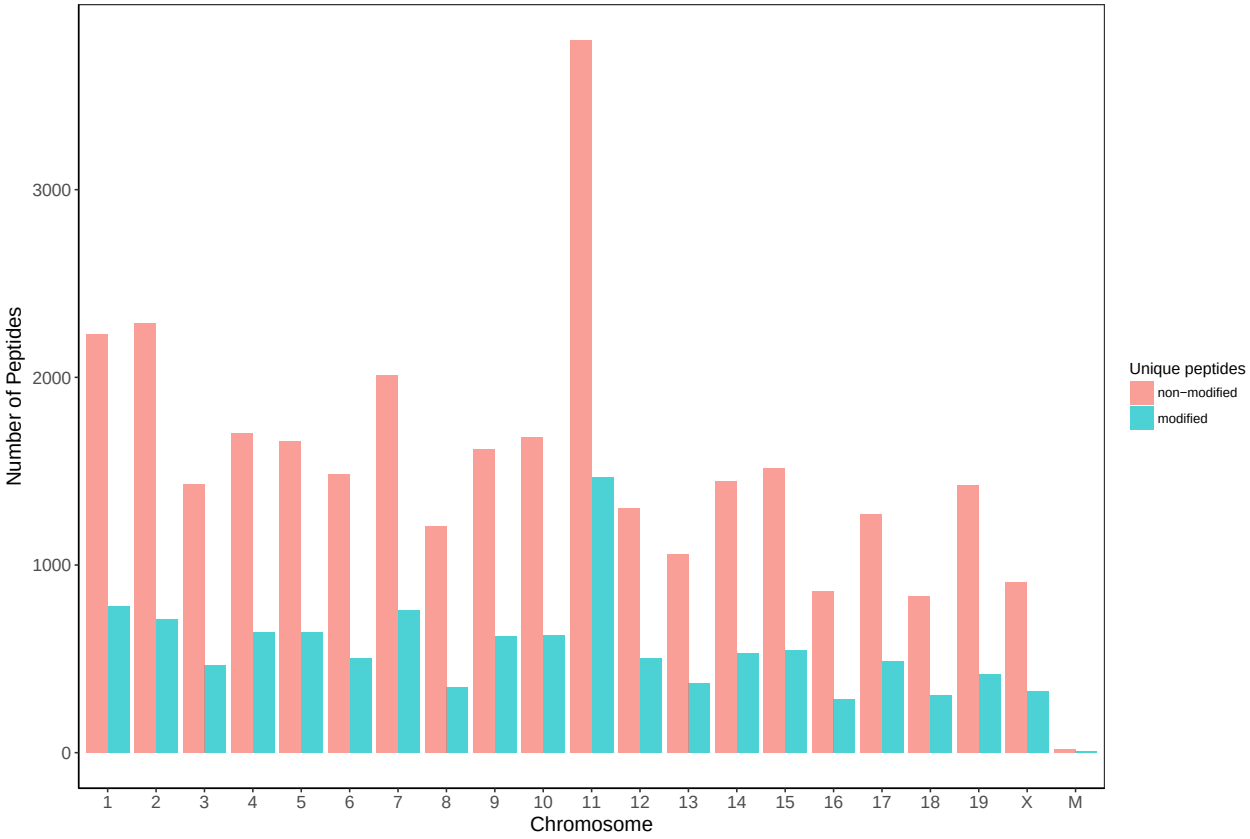
- The distribution of peptides by chromosome. (blue_track: modified peptide; red_track: non-modified)



- barplot with number of peptides (modified and non-modified) by chromosome



- barplot with number of unique peptides (modified and non-modified) by chromosome



- barplot with coverage (all peptides) by chromosome