An Introduction to *Seqnames*

Martin Morgan, Herve Pages, Marc Carlson, Sonali Arora

Modified: 17 January, 2014. Compiled: February 11, 2014

Contents

1 Introduction

The *Seqnames* provides an interface to access SeqnameStyles (such as UCSC, NCBI, Ensembl) and their supported mappings for organisms. For instance, for Homo sapiens, SeqnameStyle "UCSC" maps to "chr1", "chr2", ... , "chrX","chrY". The section below introduces these functions with examples.

2 Seqname Functionality for all exsiting organisms

2.1 supportedStyles

The `supportedStyles` lists out for each organism, the SeqnameStyles and their mappings.

```
seqmap <- supportedStyles()
head(seqmap, n = 2)
```

```
## $Arabidopsis_thaliana
##   linear  auto   sex NCBI TAIR10
## 1   TRUE  TRUE FALSE    1      1
## 2   TRUE  TRUE FALSE    2      2
## 3   TRUE  TRUE FALSE    3      3
## 4   TRUE  TRUE FALSE    4      4
## 5   TRUE  TRUE FALSE    5      5
## 6  FALSE FALSE FALSE   MT     Mt
## 7   TRUE FALSE  TRUE Pltd     Pt
```

```
##
## $Caenorhabditis_elegans
##   linear  auto   sex NCBI   UCSC Ensembl
## 1   TRUE  TRUE FALSE    I   chrI       I
## 2   TRUE  TRUE FALSE   II  chrII      II
## 3   TRUE  TRUE FALSE  III chrIII     III
## 4   TRUE  TRUE FALSE   IV  chrIV      IV
## 5   TRUE  TRUE FALSE    V   chrV       V
## 6   TRUE FALSE  TRUE    X   chrX       X
## 7  FALSE  TRUE FALSE   MT   chrM   MtDNA
```

If one knows the organism one is interested in, then we can directly access the information for the given organism along. Each function accepts an argument called species which as "genus species", the default is "Homo sapiens". In the following example we list out only the first five entries returned by the code snippet.

```
head(supportedStyles("Homo sapiens"), 5)
```

```
##   linear auto   sex NCBI UCSC
## 1   TRUE TRUE FALSE    1 chr1
## 2   TRUE TRUE FALSE    2 chr2
## 3   TRUE TRUE FALSE    3 chr3
## 4   TRUE TRUE FALSE    4 chr4
## 5   TRUE TRUE FALSE    5 chr5
```

2.2 extractSeqnameSet

We can also extract the desired SeqnameStyle from a given organism using the `extractSeqnameSet`

```
extractSeqnameSet(species = "Arabidopsis thaliana", style = "NCBI")
```

```
## [1] "1"    "2"    "3"    "4"    "5"    "MT"   "Pltd"
```

2.3 extractSeqnameSetByGroup

We can also extract the desired SeqnameStyle from a given organism based on a group (Group - 'auto' denotes autosomes, 'linear' denotes linear chromosomes and 'sex' denotes sex chromosomes; the default is all chromosomes are returned).

```
extractSeqnameSetByGroup(species = "Arabidopsis thaliana", style = "NCBI", group = "auto")
```

```
## [1] "1" "2" "3" "4" "5"
```

2.4 seqnameStyle

We can find the seqname Style for a given character vector by using the `seqnameStyle`

```
seqnameStyle(paste0("chr", c(1:30)))
```

```
## [1] "UCSC"
```

```
seqnameStyle(c("2L", "2R", "X", "Xhet"))
```

```
## [1] "NCBI"
```

2.5 seqnamesInGroup

We can also subset a given character vector containing seqnames using the `seqnamesInGroup`. We currently support 3 groups: 'auto' for autosomes, 'sex' for allosomes/sex chromosomes and linear for 'linear' chromosomes. The user can also prvoide the style and species they are working with. In the following example, we extract the sex chromosomes for Homo sapiens

```
newchr <- paste0("chr", c(1:22, "X", "Y", "M", "1_gl000192_random", "4_ctg9_hap1"))
seqnamesInGroup(newchr, group = "sex")
```

```
## [1] "chrX" "chrY"
```

```
seqnamesInGroup(newchr, group = "auto")
```

```
##  [1] "chr1"  "chr2"  "chr3"  "chr4"  "chr5"  "chr6"  "chr7"  "chr8"  "chr9"  "chr10"
## [11] "chr11" "chr12" "chr13" "chr14" "chr15" "chr16" "chr17" "chr18" "chr19" "chr20"
## [21] "chr21" "chr22"
```

```
seqnamesInGroup(newchr, group = "linear")
```

```
##  [1] "chr1"  "chr2"  "chr3"  "chr4"  "chr5"  "chr6"  "chr7"  "chr8"  "chr9"  "chr10"
## [11] "chr11" "chr12" "chr13" "chr14" "chr15" "chr16" "chr17" "chr18" "chr19" "chr20"
## [21] "chr21" "chr22" "chrX"  "chrY"
```

```
seqnamesInGroup(newchr, group = "sex", "Homo sapiens", "UCSC")
```

```
## [1] "chrX" "chrY"
```

2.6 seqnamesOrder

The `seqnamesOrder` can return the order of a given character vector which contains seqnames.In the following example, we show how you can find the order for a given seqnames character vector.

```
seqnames <- c("chr1", "chr9", "chr2", "chr3", "chr10")
seqnamesOrder(seqnames)
```

```
## [1] 1 3 4 2 5
```

2.7 findSequenceRenamingMaps

Returns a matrix with 1 column per supplied sequence name and 1 row per sequence renaming map compatible with the specified style. If `best.only` is `TRUE` (the default), only the "best" renaming maps (i.e. the rows with less NAs) are returned.

```
findSequenceRenamingMaps(c("chrII", "chrIII", "chrM"), "NCBI")
```

```
##  chrII chrIII   chrM
##   "II"  "III"   "MT"
```