BSgenome.Cfamiliaris.UCSC.canFam2.masked

April 26, 2017

Description

Full genome sequences for Canis lupus familiaris (Dog) as provided by UCSC (canFam2, May 2005) and stored in Biostrings objects. The sequences are the same as in BSgenome.Cfamiliaris.UCSC.canFam2, except that each of them has the 4 following masks on top: (1) the mask of assembly gaps (AGAPS mask), (2) the mask of intra-contig ambiguities (AMB mask), (3) the mask of repeats from RepeatMasker (RM mask), and (4) the mask of repeats from Tandem Repeats Finder (TRF mask). Only the AGAPS and AMB masks are "active" by default.

Note

The masks in this BSgenome data package were made from the following source data files:

AGAPS masks: all the chr*_gap.txt.gz files from ftp://hgdownload.cse.ucsc.edu/goldenPath/canFam2/database/
RM masks: http://hgdownload.cse.ucsc.edu/goldenPath/canFam2/bigZips/chromOut.tar.gz
TRF masks: http://hgdownload.cse.ucsc.edu/goldenPath/canFam2/bigZips/chromTrf.tar.gz

See ?BSgenome.Cfamiliaris.UCSC.canFam2 in the BSgenome.Cfamiliaris.UCSC.canFam2 package for information about how the sequences were obtained.

See ?BSgenomeForge and the BSgenomeForge vignette (vignette("BSgenomeForge")) in the BSgenome software package for how to make a BSgenome data package.

Author(s)

The Bioconductor Dev Team
See Also

- BSgenome.Cfamiliaris.UCSC.canFam2 in the BSgenome.Cfamiliaris.UCSC.canFam2 package for information about how the sequences were obtained.
- BSgenome objects and the the available.genomes function in the BSgenome software package.
- MaskedDNAString objects in the Biostrings package.
- The BSgenomeForge vignette (vignette("BSgenomeForge")) in the BSgenome software package for how to make a BSgenome data package.

Examples

BSgenome.Cfamiliaris.UCSC.canFam2.masked
genome <- BSgenome.Cfamiliaris.UCSC.canFam2.masked
seqlengths(genome)
genome$chr1 # a MaskedDNAString object!
## To get rid of the masks altogether:
unmasked(genome$chr1) # same as BSgenome.Cfamiliaris.UCSC.canFam2$chr1

if ("AGAPS" %in% masknames(genome)) {

## Check that the assembly gaps contain only Ns:
checkOnlyNsInGaps <- function(seq)
{
  ## Replace all masks by the inverted AGAPS mask
  masks(seq) <- gaps(masks(seq)["AGAPS"])
  unique_letters <- uniqueLetters(seq)
  if (any(unique_letters != "N"))
    stop("assembly gaps contain more than just Ns")
}

## A message will be printed each time a sequence is removed
## from the cache:
options(verbose=TRUE)

for (seqname in seqnames(genome)) {
  cat("Checking sequence", seqname, "...
")
  seq <- genome[[seqname]]
  checkOnlyNsInGaps(seq)
  cat("OK\n")
}

## See the GenomeSearching vignette in the BSgenome software
## package for some examples of genome-wide motif searching using
## Biostrings and the BSgenome data packages:
if (interactive())
  vignette("GenomeSearching", package="BSgenome")
Index

*Topic data
  BSgenome.Cfamiliaris.UCSC.canFam2.masked, 1

*Topic package
  BSgenome.Cfamiliaris.UCSC.canFam2.masked, 1

available.genomes, 2

BSgenome, 2
BSgenome.Cfamiliaris.UCSC.canFam2, /, /2
BSgenome.Cfamiliaris.UCSC.canFam2.masked, 1
BSgenome.Cfamiliaris.UCSC.canFam2.masked-package
  (BSgenome.Cfamiliaris.UCSC.canFam2.masked), 1
BSgenomeForge, /
MaskedDNAString, 2