Bio File Formats in R

David C. King

2025-03-01

File formats in biology

Many file formats in biology are plain text, but conform to a certain structure.

- Sequences- DNA, RNA, protein
- Sequence reads - fastq
- Sequence Alignments- MAF, SAM/BAM
- Phylogenetic trees - Newick, Phylip
- Genomic Coordinates and Annotations - BED, GTF, GFF
- Microscopy images - tiff, raw
- Protein Structure - pdb
- Mass Spectrography
- Analysis output – tidy data, csv
- Compressed files - gzip, tar, zip

DNA/RNA and Amino Acid Sequences

Purpose:

- Understand sequence content, properties.
- Domains/Structure/Function.
- Variation: SNPs, indels, transitions/transversions.
- Identity of a gene, relationship to other genes, other species (paralogs/orthologs).
- Next-gen sequencing- the presence and identity of the sequence is a measurement (a count of expressed transcripts, enriched fragments)
- Design primers for PCR

FASTA Format

Comes from FAST sequence aligner. Fast**n**: nucleotide; Fast**p**: protein: Fast**a**: all.

Definition:

https://en.wikipedia.org/wiki/FASTA_format

https://www.ncbi.nlm.nih.gov/genbank/fastaformat/

1. Header line starts with ">" followed by the name of the sequence. More information can be provided but must be on the same line.

2. Sequence follows on one or more lines.
3. Multiple header/sequence pairs can be in the same file (sometimes this is called “multi-fasta”)

Examples:

```
>gi|1817694395|ref|NZ_JAAGMU010000151.1| Streptomyces sp. SID7958 contig-52000002, whole genome shotgun
CCGGCTGGCGCGGCTGGCGCTGGCGGTGGGGCTGCGGCTGCTGGAGCTGGGGGTGGCGCTGGAGGCGCAC
GGCCAGAACCTGCTGGTGGTGCTGTCGCCGTCCGGGGAGCCGCGGCGGCTGGTCTACCGCGATCTGGCGG
ACATCCGGGTCTCCCCCGCGCGGCTGGCCCGGCACGGTATCCGGGTTCCGGACCTGCCGGCG

>gi|1643051563|gb|SZWM01000399.1| Citrobacter sp. TBCS-14 contig3128, whole genome shotgun sequence
GCACAGTGAGATCAGCATTCCGTTGGATCTACTGGTCAATCAAAACCTGACGCTGGGTACTGAATGGAAC
CAGCAGCGCATGAAGGACATGCTGTCTAACTCGCAGACCTTTATGGGCGGTAATATTCCAGGCTACAGCA
GCACCGATCGCAGCCCATATTCGAAAGCCGAGATCTTCTCTTTGTTTGCCGAAAACAACATG
```

These sequences have a complex header with multiple, |-delimited fields.

What’s missing? Sequence annotations and features.

- Example: Say there is a stem-loop at position 150-200. Have to record it separately.
- Exception: masking with uppercase/lowercase, or another symbol (i.e. N, X)

Masking Sequence aligners are often tripped up by short repeat fragments and low complexity regions. RepeatMasker is a common tool for finding repeats and *masking* them.

One convention is to use the character N for a hard-masked DNA sequence, or use lowercase letters for softmasking. *This doesn’t work if N is being used for an ambiguity (the exact base at that position is uncertain).*

Hard Masking (David made this example up)

In the Streptomyces example above, replace nucleotide letter with ‘N’ if sequence is in a repeat (found by RepeatMasker)

```
CCGGCTGGCGCGGCTGGCGCTGGCGGTGGGGCTGCGGCTGCTGGAGCTGGGGGNNNNNNNNNNNNNNNNN
NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNCTGGTCTACCGCGATCTGGCGG
ACATCCGGGTCTCCCCCGCGCGGCTGGCCCGGCACGGTATCCGGGTTCCGGACCTGCCGGCG
```

Software that deals with sequence analysis will be forced to ignore the N’s., but maintain the spacing.

Soft Masking Sometimes you want to retain the sequence, but convey that it is masked. This is usually done with the masked bases being converted to lowercase.

```
CCGGCTGGCGCGGCTGGCGCTGGCGGTGGGGCTGCGGCTGCTGGAGCTGGGGGtggcgctggaggcgcac
ggccagaacctgctggtggtgctgtcgccgtccggggagccgcggcggCTGGTCTACCGCGATCTGGCGG
ACATCCGGGTCTCCCCCGCGCGGCTGGCCCGGCACGGTATCCGGGTTCCGGACCTGCCGGCG
```

Software might not take case into account, it depends on the application, and behavior can be configured.

In R

```r
# Read from a fasta file
seqs = readDNAStringSet("seqs_for_markdown.fasta")
seqs
```

```
## DNAStringSet object of length 2:
##     width seq                                               names
## [1]   202 CCGGCTGGCGCGGCTGGCGCTGG...TCCGGGTTCCGGACCTGCCGGCG gi|1817694395|ref...
## [2]   202 GCACAGTGAGATCAGCATTCCGT...CTTTGTTTGCCGAAAACAACATG gi|1643051563|gb|...
```

```
# Save the names because they are long, and use shorter ones
long_names = names(seqs)
names(seqs) = c("Streptomyces","Citrobacter")
seqs
```

```
## DNAStringSet object of length 2:
##     width seq                                               names
## [1]   202 CCGGCTGGCGCGGCTGGCGCTGG...TCCGGGTTCCGGACCTGCCGGCG Streptomyces
## [2]   202 GCACAGTGAGATCAGCATTCCGT...CTTTGTTTGCCGAAAACAACATG Citrobacter
```

```
# Some functions

## Dinucleotide frequencies
oligonucleotideFrequency(seqs[1], 2)
```

```
##      AA AC AG AT CA CC CG CT GA GC GG GT TA TC TG TT
## [1,]  1  6  4  3  4 18 28 17  7 33 39 10  2  9 19  1
```

```
oligonucleotideFrequency(seqs[2], 1, as.prob=T)
```

```
##              A         C         G         T
## [1,] 0.2772277 0.2574257 0.2376238 0.2277228
```

```
# look at hexamers for the most common ones
hexamer_frequencies = oligonucleotideFrequency(seqs[1], 6) # matrix of one row, 4096 columns
hexamer_frequencies[1,1:10] # can't use head on a wide object like this
```

```
## AAAAAA AAAAAC AAAAAG AAAAAT AAAACA AAAACC AAAACG AAAACT AAAAGA AAAAGC
##      0      0      0      0      0      0      0      0      0      0
```

```
descending_order = order(hexamer_frequencies, decreasing = TRUE) # returns the INDEX of the the largest
descending_order[1:10] # top 10 hexamer indices
```

```
##  [1] 1695 1959 2472 2538 2683 1642 3435 3738  378  859
```

```
hexamer_frequencies[,descending_order[1:10]] # top ten hexamers (in a matrix)
```

```
## CGGCTG CTGGCG GCGGCT GCTGGC GGCTGG CGCGGC TCCGGG TGGCGC ACCTGC ATCCGG
##      5      4      4      4      4      3      3      3      2      2
```

```
top10 = names(hexamer_frequencies[,descending_order[1:10]]) # get the hexamers from the names


# turn the top10 into a sequence object
dict0 = DNAStringSet(top10)
pdict0 = PDict(dict0) # PDict for Pattern DICTionary

# find the occurences of the top 10 hexamers in the sequence
matches = matchPDict(pdict0, seqs[[1]])
for (i in 1:10) {print(top10[i]); print(replaceAt(seqs[[1]], matches[[i]], value = 'NNNNNN'))}
```

```
## [1] "CGGCTG"
## 202-letter DNAString object
## seq: CNNNNNNGCGNNNNNNGCGCTGGCGGTGGGGCTGNN...GCCCGGCACGGTATCCGGGTTCCGGACCTGCCGGCG
## [1] "CTGGCG"
## 202-letter DNAString object
## seq: CCGGNNNNNNCGGNNNNNNNNNNNNGTGGGGCTGCG...GCCCGGCACGGTATCCGGGTTCCGGACCTGCCGGCG
## [1] "GCGGCT"
## 202-letter DNAString object
## seq: CCGGCTGGCNNNNNNGGCGCTGGCGGTGGGGCTNNN...GCCCGGCACGGTATCCGGGTTCCGGACCTGCCGGCG
## [1] "GCTGGC"
## 202-letter DNAString object
## seq: CCGNNNNNNGCGNNNNNNNNNNNNGGTGGGGCTGCG...NNCCGGCACGGTATCCGGGTTCCGGACCTGCCGGCG
## [1] "GGCTGG"
## 202-letter DNAString object
## seq: CCNNNNNNCGCNNNNNNCGCTGGCGGTGGGGCTGCG...NCCCGGCACGGTATCCGGGTTCCGGACCTGCCGGCG
## [1] "CGCGGC"
## 202-letter DNAString object
## seq: CCGGCTGGNNNNNNTGGCGCTGGCGGTGGGGCTGCG...GCCCGGCACGGTATCCGGGTTCCGGACCTGCCGGCG
## [1] "TCCGGG"
## 202-letter DNAString object
## seq: CCGGCTGGCGCGGCTGGCGCTGGCGGTGGGGCTGCG...GCCCGGCACGGTANNNNNNTTCCGGACCTGCCGGCG
## [1] "TGGCGC"
## 202-letter DNAString object
## seq: CCGGCNNNNNNGGCNNNNNNTGGCGGTGGGGCTGCG...GCCCGGCACGGTATCCGGGTTCCGGACCTGCCGGCG
## [1] "ACCTGC"
## 202-letter DNAString object
## seq: CCGGCTGGCGCGGCTGGCGCTGGCGGTGGGGCTGCG...GCCCGGCACGGTATCCGGGTTCCGGNNNNNNCGGCG
## [1] "ATCCGG"
## 202-letter DNAString object
## seq: CCGGCTGGCGCGGCTGGCGCTGGCGGTGGGGCTGCG...GCCCGGCACGGTNNNNNNGTTCCGGACCTGCCGGCG
```

```
expectedDinucleotideFrequency <- function(x)
{
    # Individual base frequencies.
    bf <- alphabetFrequency(x, baseOnly=TRUE)[DNA_BASES]
    (as.matrix(bf) %*% t(bf) - diag(bf)) / length(x)
}

## On Celegans chrI:
library(BSgenome.Celegans.UCSC.ce11)
```

```
## Loading required package: BSgenome

## Loading required package: GenomicRanges

## Loading required package: BiocIO

## Loading required package: rtracklayer

##
## Attaching package: 'rtracklayer'

## The following object is masked from 'package:BiocIO':
##
##     FileForFormat
```

```
chrI <- Celegans$chrI
obs_df <- dinucleotideFrequency(chrI, as.matrix=TRUE)
obs_df  # CG has the lowest frequency
```

```
##         A      C      G       T
## A 2049736 704043 758861 1323299
## C  909649 521372 503518  761350
## G  953429 515117 518679  704930
## T  923125 955358 911096 2058871
```

```r
exp_df <- expectedDinucleotideFrequency(chrI)
## A sanity check:
stopifnot(as.integer(sum(exp_df)) == sum(obs_df))

## Ratio of observed frequency to expected frequency:
obs_df / exp_df  # TA has the lowest ratio, not CG!
```

```
##           A         C         G         T
## A 1.3210516 0.8139535 0.8785465 0.8506632
## C 1.0516573 1.0812520 1.0456736 0.8779356
## G 1.1038012 1.0697617 1.0786538 0.8140037
## T 0.5934173 1.1016520 1.0520698 1.3200998
```

Annotation Databases

Gene annotations, sequences, other data that can be mapped to genes from a variety of sources/experiments.

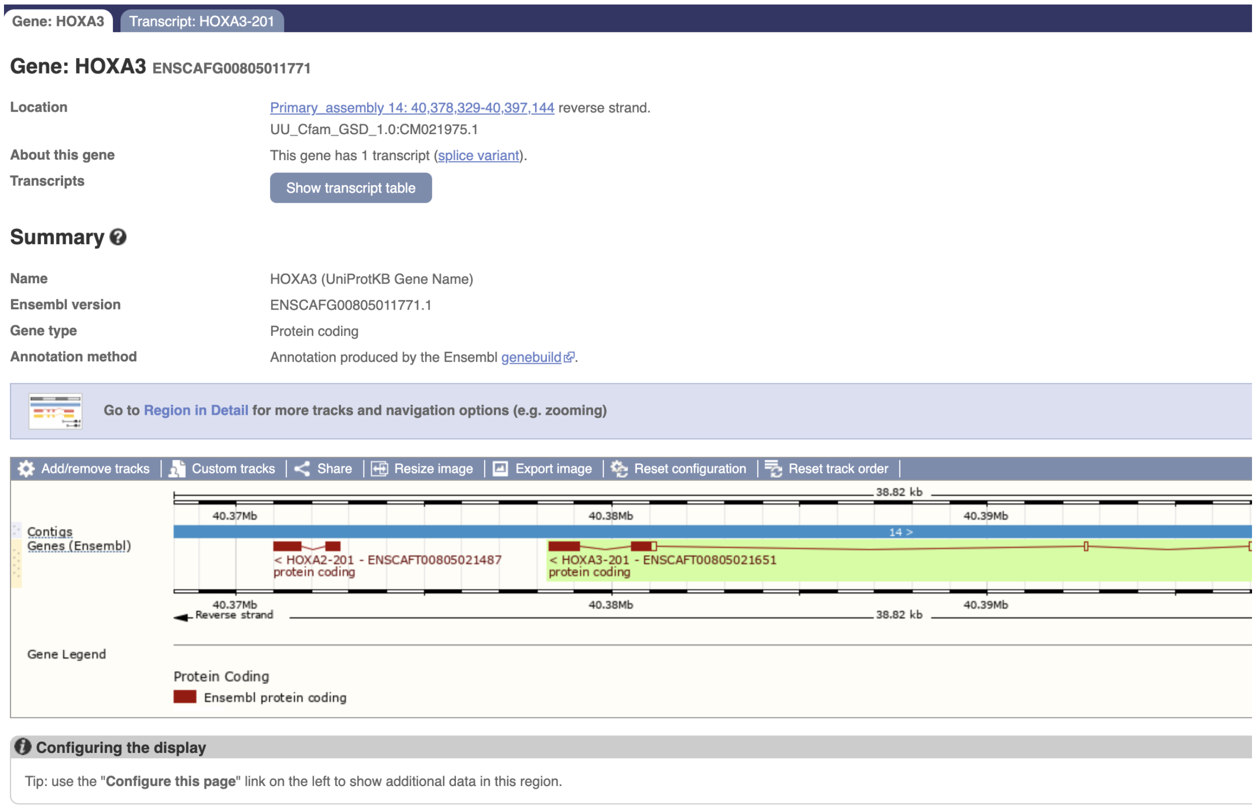
NCBI - National Center for Biotechnology Information

UCSC Genome Browser - University of California Santa Cruz

Ensembl - European Bioinformatics Institute

Ensembl Genome Browser

Ensembl *Canis lupus familiaris* HOXA4 (for German Shephard)



Gene: HOXA3 | Transcript: HOXA3-201

Gene: HOXA3 ENSCAFG00805011771

Location	Primary_assembly 14: 40,378,329-40,397,144 reverse strand. UU_Cfam_GSD_1.0:CM021975.1
About this gene	This gene has 1 transcript (splice variant).
Transcripts	Show transcript table

Summary

Name	HOXA3 (UniProtKB Gene Name)
Ensembl version	ENSCAFG00805011771.1
Gene type	Protein coding
Annotation method	Annotation produced by the Ensembl genebuild.

Go to Region in Detail for more tracks and navigation options (e.g. zooming)

Configuring the display

Tip: use the "**Configure this page**" link on the left to show additional data in this region.

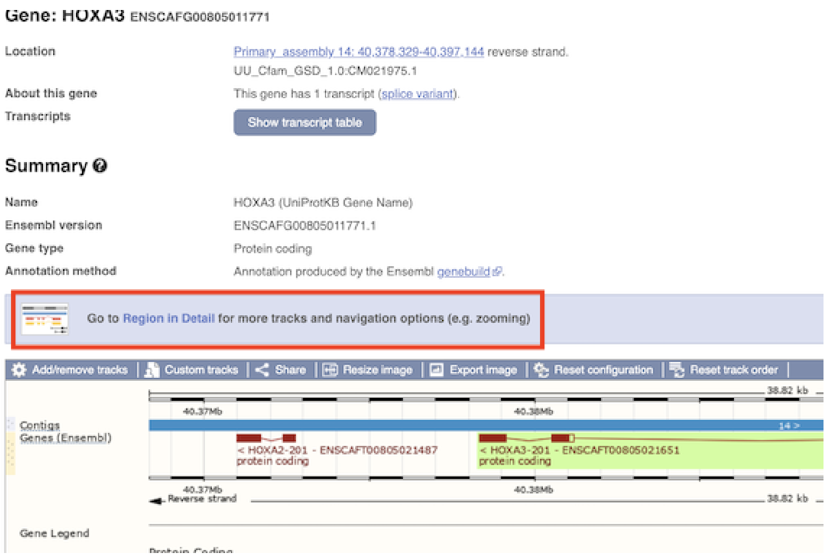
Tracks and Configuration

- Tracks displayed in screenshot

  - Contigs - sequence assembly
  - Genes (Ensembl) - The Enbsembl gene annotations; others may be included

- Configuration
  - Tabs: Gene/Transcript
  - Add/remove tracks
  - Legend, Strand, scale
  - Export options

Adding tracks Click on *Region in Detail*



Gene: HOXA3 ENSCAFG00805011771

Location	Primary_assembly 14: 40,378,329-40,397,144 reverse strand. UU_Cfam_GSD_1.0:CM021975.1
About this gene	This gene has 1 transcript (splice variant).
Transcripts	Show transcript table

Summary

Name	HOXA3 (UniProtKB Gene Name)
Ensembl version	ENSCAFG00805011771.1
Gene type	Protein coding
Annotation method	Annotation produced by the Ensembl genebuild.

Go to Region in Detail for more tracks and navigation options (e.g. zooming)

Figure 1: HOX A4, German Shephard

Chose some tracks to show, exported the image.

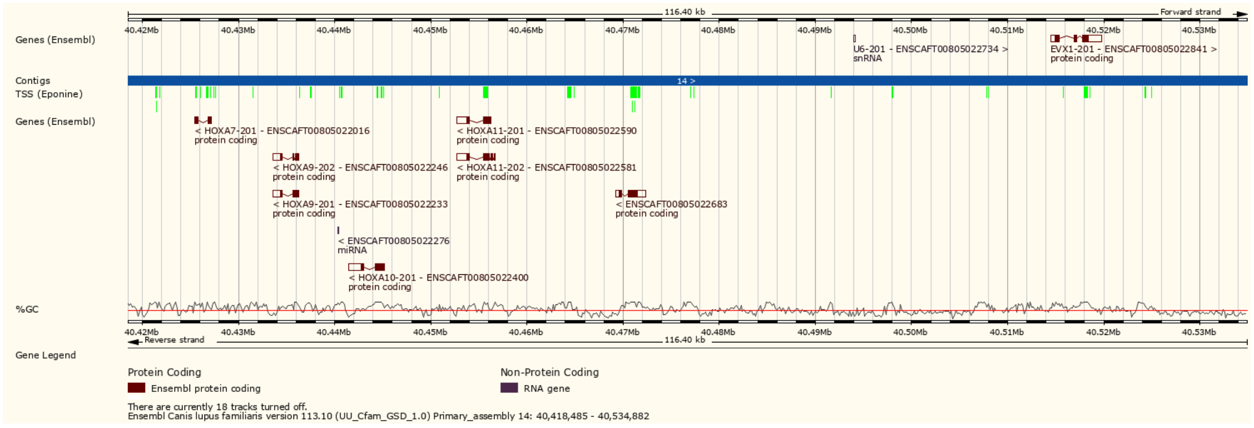


Figure 2: HOX A4, German Shephard

Changes:

- Zoomed out
- TSS (eponine) - Transciption start sites
- GC% Continuous, numeric data

Experiment with adding/hiding tracks:

HOX A4, German Shephard

HOXA4 at UCSC https://genome.ucsc.edu/s/davidcking/canFam4_hoxA10

They don't have the rest of the gene cluster!?!?!