PH525.5x Section 4: Genomic annotation with Bioconductor

Lauren Kemperman

4/13/2021

Representing Reference Sequence

- Annotation concept hierarchy
- Base - reference genomic sequence for an organism
- Above this, organize the chromosomal sequence into regions of interest - i.e. genes, transcripts
- SNPs and CpG sites are also regions of interest
- SNPS are single nucleotide
- Other varients – indels, structural variants, fusions can constitute regions of interest but are more complicated to express + represent
- Within ROI, identify platform oriented annotation provided by assay manufacturer
- Once manufacturing happens, genomic annotation proceeds and annotations must be updated to account for ambiguities or updates for assay probe elements
- Above genomic sequence ROIs, annotations concerning groups with shared structural or functional properties
- Pathways with nodes being genes and paths being relationships between gene products, i.e. protein protein interaction, promotion, enhancement, repression (3rd level of hierarchy)
- Begin with reference genomes
- Biostrings package - **available.genomes** - packages that represent reference genomic sequences for many different organisms
- Homo sapiens reference - some have repeat masking and there are versions which include the masked regions
    - different numbers of sequences in the two builds due to contigs that haven't been placed on chromosomes yet
- Operations defined for BSGenome objects - substring, extract chromosomal information
- Bases in full sequence aren't completely resolved
- Application of iteration - count the number of bases in a number of chromosomes
- If you have enough RAM, it is possible to operate on chromosomes in parallel and performing operations using multicore programming

```
library(BSgenome)
```

```
## Loading required package: BiocGenerics
```

```
## Loading required package: parallel
```

```
## 
## Attaching package: 'BiocGenerics'

## The following objects are masked from 'package:parallel':
## 
##     clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##     clusterExport, clusterMap, parApply, parCapply, parLapply,
##     parLapplyLB, parRapply, parSapply, parSapplyLB

## The following objects are masked from 'package:stats':
## 
##     IQR, mad, sd, var, xtabs

## The following objects are masked from 'package:base':
## 
##     anyDuplicated, append, as.data.frame, basename, cbind, colnames,
##     dirname, do.call, duplicated, eval, evalq, Filter, Find, get, grep,
##     grepl, intersect, is.unsorted, lapply, Map, mapply, match, mget,
##     order, paste, pmax, pmax.int, pmin, pmin.int, Position, rank,
##     rbind, Reduce, rownames, sapply, setdiff, sort, table, tapply,
##     union, unique, unsplit, which.max, which.min

## Loading required package: S4Vectors

## Loading required package: stats4

## 
## Attaching package: 'S4Vectors'

## The following object is masked from 'package:base':
## 
##     expand.grid

## Loading required package: IRanges

## Loading required package: GenomeInfoDb

## Warning: package 'GenomeInfoDb' was built under R version 4.0.5

## Loading required package: GenomicRanges

## Loading required package: Biostrings

## Loading required package: XVector

## 
## Attaching package: 'Biostrings'

## The following object is masked from 'package:base':
## 
##     strsplit

## Loading required package: rtracklayer
```

```r
library(Biostrings)
ag = available.genomes()
grep("Scerev", ag, value=TRUE)
```

```
## [1] "BSgenome.Scerevisiae.UCSC.sacCer1" "BSgenome.Scerevisiae.UCSC.sacCer2"
## [3] "BSgenome.Scerevisiae.UCSC.sacCer3"
```

```r
grep("Hsap", ag, value=TRUE)
```

```
##  [1] "BSgenome.Hsapiens.1000genomes.hs37d5"
##  [2] "BSgenome.Hsapiens.NCBI.GRCh38"
##  [3] "BSgenome.Hsapiens.UCSC.hg17"
##  [4] "BSgenome.Hsapiens.UCSC.hg17.masked"
##  [5] "BSgenome.Hsapiens.UCSC.hg18"
##  [6] "BSgenome.Hsapiens.UCSC.hg18.masked"
##  [7] "BSgenome.Hsapiens.UCSC.hg19"
##  [8] "BSgenome.Hsapiens.UCSC.hg19.masked"
##  [9] "BSgenome.Hsapiens.UCSC.hg38"
## [10] "BSgenome.Hsapiens.UCSC.hg38.masked"
```

```r
# inspect the human genome
library(BSgenome.Hsapiens.UCSC.hg19)
Hsapiens
```

```
## Human genome:
## # organism: Homo sapiens (Human)
## # genome: hg19
## # provider: UCSC
## # release date: June 2013
## # 298 sequences:
## #   chr1                  chr2                  chr3
## #   chr4                  chr5                  chr6
## #   chr7                  chr8                  chr9
## #   chr10                 chr11                 chr12
## #   chr13                 chr14                 chr15
## #   ...                   ...                   ...
## #   chr19_gl949749_alt    chr19_gl949750_alt    chr19_gl949751_alt
## #   chr19_gl949752_alt    chr19_gl949753_alt    chr20_gl383577_alt
## #   chr21_gl383578_alt    chr21_gl383579_alt    chr21_gl383580_alt
## #   chr21_gl383581_alt    chr22_gl383582_alt    chr22_gl383583_alt
## #   chr22_kb663609_alt
## # (use 'seqnames()' to see all the sequence names, use the '$' or '[[' operator
## # to access a given sequence)
```

```r
length(Hsapiens)
```

```
## [1] 298
```

```r
class(Hsapiens)
```

```
## [1] "BSgenome"
## attr(,"package")
## [1] "BSgenome"
```

```r
methods(class="BSgenome")
```

```
##  [1] [[              $               as.list         bsgenomeName
##  [5] coerce          commonName      countPWM        export
##  [9] extractAt       getSeq          injectSNPs      length
## [13] masknames       matchPWM        metadata        metadata<-
## [17] mseqnames       names           organism        provider
## [21] providerVersion releaseDate     releaseName     seqinfo
## [25] seqinfo<-       seqnames        seqnames<-      show
## [29] snpcount        SNPlocs_pkgname snplocs         sourceUrl
## [33] vcountPattern   vcountPDict     Views           vmatchPattern
## [37] vmatchPDict
```

```
## see '?methods' for accessing help and source code
# inspect human genome
Hsapiens$chrX
```

```
## 155270560-letter DNAString object
## seq: NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN...NNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNNN
```

```
substr(Hsapiens$chrX, 5e6, 5.1e6)
```

```
## 100001-letter DNAString object
## seq: GCCTCAATGTCAGAATTATGCTGTTGCCCAAAATTG...TACTAAAAATACAAAAATTAGCTGGGCATGGTGGTG
```

```
nchar(Hsapiens$chrY)
```

```
## [1] 59373566
```

```
nchar(Hsapiens[[24]])
```

```
## [1] 59373566
```

```
library(parallel)
options(mc.cores=detectCores())

system.time(sum(unlist(mclapply(18:24, function(x) nchar(Hsapiens[[x]])))))
```

```
##    user  system elapsed
##   5.345   4.017  10.281
```

Assessment: Reference Genomes

```
library(BSgenome)
library(Biostrings)
ag = available.genomes()
library(BSgenome)
grep("mask", grep("Drerio", available.genomes(), value=TRUE), invert=TRUE, value=TRUE) # exclude masked
```

```
## [1] "BSgenome.Drerio.UCSC.danRer10" "BSgenome.Drerio.UCSC.danRer11"
## [3] "BSgenome.Drerio.UCSC.danRer5"  "BSgenome.Drerio.UCSC.danRer6"
## [5] "BSgenome.Drerio.UCSC.danRer7"
```

```
library(BSgenome.Hsapiens.UCSC.hg19.masked)
c17m = BSgenome.Hsapiens.UCSC.hg19.masked$chr17


c22m = BSgenome.Hsapiens.UCSC.hg19.masked$chr22
round(100*sum(width(masks(c22m)$AGAPS))/length(c22m),0)
```

```
## [1] 32
```

Gene, Transcript and Exon Databases

- Can find information about reference genome regions such as genes, transcripts and exons on annotation packages
- UCSC Genome Browser - major source of reference genome structure annotation
- **TxDb.Hsapiens.UCSC.hg19** - collection of well documented protein coding genes, transcripts and exons on the hg19 build of the human genome. Additional TxDb packages exist for other organisms and genome builds

- Introduction to TxDb package architecture

```
# Import TxDb transcript database
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
```

```
## Loading required package: GenomicFeatures

## Warning: package 'GenomicFeatures' was built under R version 4.0.4

## Loading required package: AnnotationDbi

## Loading required package: Biobase

## Welcome to Bioconductor
##
##     Vignettes contain introductory material; view with
##     'browseVignettes()'. To cite Bioconductor, see
##     'citation("Biobase")', and for packages 'citation("pkgname")'.
```

```
txdb = TxDb.Hsapiens.UCSC.hg19.knownGene
class(txdb)
```

```
## [1] "TxDb"
## attr(,"package")
## [1] "GenomicFeatures"
```

```
methods(class="TxDb")
```

```
##  [1] $                    $<-                  annotatedDataFrameFrom
##  [4] as.list              asBED                asGFF
##  [7] assayData            assayData<-          cds
## [10] cdsBy                cdsByOverlaps        coerce
## [13] columns              combine              contents
## [16] dbconn               dbfile               dbInfo
## [19] dbmeta               dbschema             disjointExons
## [22] distance             exons                exonsBy
## [25] exonsByOverlaps      ExpressionSet        extractUpstreamSeqs
## [28] featureNames         featureNames<-       fiveUTRsByTranscript
## [31] genes                initialize           intronsByTranscript
## [34] isActiveSeq          isActiveSeq<-        isNA
## [37] keys                 keytypes             mapIds
## [40] mapIdsToRanges       mappedkeys           mapRangesToIds
## [43] mapToTranscripts     metadata             microRNAs
## [46] nhit                 organism             promoters
## [49] revmap               sample               sampleNames
## [52] sampleNames<-        saveDb               select
## [55] seqinfo              seqinfo<-            seqlevels<-
## [58] seqlevels0           show                 species
## [61] storageMode          storageMode<-        taxonomyId
## [64] threeUTRsByTranscript transcripts         transcriptsBy
## [67] transcriptsByOverlaps tRNAs               updateObject
## see '?methods' for accessing help and source code
```

```
# extract and inspect genes from TxDb
genes(txdb)
```

```
##   403 genes were dropped because they have exons located on both strands
##   of the same reference sequence or on more than one reference sequence,
##   so cannot be represented by a single genomic range.
```

```
##   Use 'single.strand.genes.only=FALSE' to get all the genes in a
##   GRangesList object, or use suppressMessages() to suppress this message.

## GRanges object with 23056 ranges and 1 metadata column:
##         seqnames              ranges strand |     gene_id
##            <Rle>           <IRanges>  <Rle> | <character>
##       1    chr19   58858172-58874214      - |           1
##      10     chr8   18248755-18258723      + |          10
##     100    chr20   43248163-43280376      - |         100
##    1000    chr18   25530930-25757445      - |        1000
##   10000     chr1 243651535-244006886      - |       10000
##     ...      ...                 ...    ... .         ...
##    9991     chr9 114979995-115095944      - |        9991
##    9992    chr21   35736323-35743440      + |        9992
##    9993    chr22   19023795-19109967      - |        9993
##    9994     chr6   90539619-90584155      + |        9994
##    9997    chr22   50961997-50964905      - |        9997
##   -------
##   seqinfo: 93 sequences (1 circular) from hg19 genome
```

```r
table(strand(genes(txdb)))
```

```
##   403 genes were dropped because they have exons located on both strands
##   of the same reference sequence or on more than one reference sequence,
##   so cannot be represented by a single genomic range.
##   Use 'single.strand.genes.only=FALSE' to get all the genes in a
##   GRangesList object, or use suppressMessages() to suppress this message.

##
##     +     -     *
## 11737 11319     0
```

```r
summary(width(genes(txdb)))
```

```
##   403 genes were dropped because they have exons located on both strands
##   of the same reference sequence or on more than one reference sequence,
##   so cannot be represented by a single genomic range.
##   Use 'single.strand.genes.only=FALSE' to get all the genes in a
##   GRangesList object, or use suppressMessages() to suppress this message.

##    Min. 1st Qu.  Median    Mean 3rd Qu.     Max.
##      20    5666   20116   60660   58175 24187703
```

```r
# inspect larges gene in genome
id = which.max(width(genes(txdb)))
```

```
##   403 genes were dropped because they have exons located on both strands
##   of the same reference sequence or on more than one reference sequence,
##   so cannot be represented by a single genomic range.
##   Use 'single.strand.genes.only=FALSE' to get all the genes in a
##   GRangesList object, or use suppressMessages() to suppress this message.
```

```r
genes(txdb)[id]
```

```
##   403 genes were dropped because they have exons located on both strands
##   of the same reference sequence or on more than one reference sequence,
##   so cannot be represented by a single genomic range.
##   Use 'single.strand.genes.only=FALSE' to get all the genes in a
```

```
##   GRangesList object, or use suppressMessages() to suppress this message.

## GRanges object with 1 range and 1 metadata column:
##         seqnames            ranges strand |     gene_id
##            <Rle>         <IRanges>  <Rle> | <character>
##   286297     chr9 42844370-67032072      - |      286297
##   -------
##   seqinfo: 93 sequences (1 circular) from hg19 genome
```

```r
library(org.Hs.eg.db)
```

```
##
```

```r
select(org.Hs.eg.db, keys="286297", keytype="ENTREZID", columns=c("SYMBOL", "GENENAME"))
```

```
## 'select()' returned 1:1 mapping between keys and columns

##   ENTREZID    SYMBOL
## 1   286297 LOC286297
##                                                            GENENAME
## 1 methylenetetrahydrofolate dehydrogenase (NADP+ dependent) 1 like pseudogene
```

```r
# compare total size of exons to total size of genes
ex = exons(txdb)
rex = reduce(ex)
ex_width = sum(width(rex)) # bases in exons
gene_width = sum(width(genes(txdb))) # bases in genes
```

```
##   403 genes were dropped because they have exons located on both strands
##   of the same reference sequence or on more than one reference sequence,
##   so cannot be represented by a single genomic range.
##   Use 'single.strand.genes.only=FALSE' to get all the genes in a
##   GRangesList object, or use suppressMessages() to suppress this message.
```

```r
ex_width/gene_width
```

```
## [1] 0.06380062
```

ensembldb, EnsDb: annotation from EMBL

- European initiative for annotating genome called ensembl
- Ensemble-based representations managed in package called EmsembleDb
- Different packages representing different builds of ensembl annotation for different organisms
- More direct relationship to database and database tables - gene, transcipt, transcript to exon mapping tables.
- More details provided to user through Ensembl transcripts method - get info on transcripts but also associated proteins, genes and biotype

```r
# inspect data available from Ensembl
library(ensembldb)
```

```
## Loading required package: AnnotationFilter
```

```
##
## Attaching package: 'ensembldb'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
library(EnsDb.Hsapiens.v75)
names(listTables(EnsDb.Hsapiens.v75))
```

```
##  [1] "gene"            "tx"              "tx2exon"         "exon"
##  [5] "chromosome"      "protein"         "uniprot"         "protein_domain"
##  [9] "entrezgene"      "metadata"
```

```
# extract Ensembl transcripts
edb = EnsDb.Hsapiens.v75 # abbreviate
txs <- transcripts(edb, filter = GeneNameFilter("ZBTB16"),
                   columns = c("protein_id", "uniprot_id", "tx_biotype"))
txs
```

```
## GRanges object with 20 ranges and 5 metadata columns:
##                   seqnames              ranges strand |        protein_id
##                      <Rle>           <IRanges>  <Rle> |       <character>
##   ENST00000335953     11 113930315-114121398      + | ENSP00000338157
##   ENST00000335953     11 113930315-114121398      + | ENSP00000338157
##   ENST00000335953     11 113930315-114121398      + | ENSP00000338157
##   ENST00000335953     11 113930315-114121398      + | ENSP00000338157
##   ENST00000335953     11 113930315-114121398      + | ENSP00000338157
##               ...    ...                 ...    ... .               ...
##   ENST00000392996     11 113931229-114121374      + | ENSP00000376721
##   ENST00000539918     11 113935134-114118066      + | ENSP00000445047
##   ENST00000545851     11 114051488-114118018      + |              <NA>
##   ENST00000535379     11 114107929-114121279      + |              <NA>
##   ENST00000535509     11 114117512-114121198      + |              <NA>
##                   uniprot_id           tx_biotype           tx_id
##                  <character>          <character>     <character>
##   ENST00000335953  ZBT16_HUMAN       protein_coding ENST00000335953
##   ENST00000335953 Q71UL7_HUMAN       protein_coding ENST00000335953
##   ENST00000335953 Q71UL6_HUMAN       protein_coding ENST00000335953
##   ENST00000335953 Q71UL5_HUMAN       protein_coding ENST00000335953
##   ENST00000335953 F5H6C3_HUMAN       protein_coding ENST00000335953
##               ...          ...                  ...             ...
##   ENST00000392996 F5H5Y7_HUMAN       protein_coding ENST00000392996
##   ENST00000539918         <NA> nonsense_mediated_de.. ENST00000539918
##   ENST00000545851         <NA>   processed_transcript ENST00000545851
##   ENST00000535379         <NA>   processed_transcript ENST00000535379
##   ENST00000535509         <NA>       retained_intron ENST00000535509
##                     gene_name
##                   <character>
##   ENST00000335953       ZBTB16
##   ENST00000335953       ZBTB16
##   ENST00000335953       ZBTB16
##   ENST00000335953       ZBTB16
##   ENST00000335953       ZBTB16
##               ...          ...
##   ENST00000392996       ZBTB16
##   ENST00000539918       ZBTB16
##   ENST00000545851       ZBTB16
##   ENST00000535379       ZBTB16
##   ENST00000535509       ZBTB16
##   -------
##   seqinfo: 1 sequence from GRCh37 genome
```

```
# compare Ensembl and UCSC transcripts
alltx = transcripts(edb) # Ensembl is larger
utx = transcripts(txdb) # UCSC is smaller

# table of biological types of transcripts
table(alltx$tx_biotype)
```

```
##
##              3prime_overlapping_ncrna                          antisense
##                                    29                              10058
##                             IG_C_gene                    IG_C_pseudogene
##                                    31                                 13
##                             IG_D_gene                          IG_J_gene
##                                    64                                 24
##                       IG_J_pseudogene                          IG_V_gene
##                                     6                                185
##                       IG_V_pseudogene                            lincRNA
##                                   264                              12101
##                              LRG_gene                              miRNA
##                                   477                               3424
##                              misc_RNA                            Mt_rRNA
##                                  2190                                  2
##                               Mt_tRNA                     non_stop_decay
##                                    22                                 63
##              nonsense_mediated_decay             polymorphic_pseudogene
##                                 13812                                 70
##                 processed_pseudogene               processed_transcript
##                                 11321                              31417
##                        protein_coding                         pseudogene
##                                 90273                                664
##                        retained_intron                              rRNA
##                                 28579                                570
##                        sense_intronic                   sense_overlapping
##                                   827                                342
##                                snoRNA                              snRNA
##                                  1621                               2074
##                             TR_C_gene                          TR_D_gene
##                                     6                                  3
##                             TR_J_gene                    TR_J_pseudogene
##                                    82                                  4
##                             TR_V_gene                    TR_V_pseudogene
##                                   150                                 40
##   transcribed_processed_pseudogene transcribed_unprocessed_pseudogene
##                                   476                                986
##     translated_processed_pseudogene                  unitary_pseudogene
##                                     1                                189
##                 unprocessed_pseudogene
##                                  3187
```

Assessment: Gene and transcript model

```
library(devtools)
```

```
## Loading required package: usethis
```

```
install_github("genomicsclass/ph525x")
```

```
## Skipping install of 'ph525x' from a github remote, the SHA1 (e83c0d57) has not changed since last in
##   Use `force = TRUE` to force installation
```

```
library(ph525x)
```

```
## Loading required package: png
```

```
## Loading required package: grid
```

```
## Loading required package: Homo.sapiens
```

```
## Loading required package: OrganismDbi
```

```
## Loading required package: GO.db
```

```
##
```

```
stopifnot(packageVersion("ph525x") >= "0.0.16") # do over if fail
modPlot("ESR1", useGeneSym=FALSE, collapse=FALSE)
```

```
## Loading required package: Gviz
```

```
## Warning: package 'Gviz' was built under R version 4.0.4
```

```
##
## Attaching package: 'Gviz'
```

```
## The following object is masked from 'package:AnnotationFilter':
##
##     feature
```

```
## 'select()' returned 1:many mapping between keys and columns
```
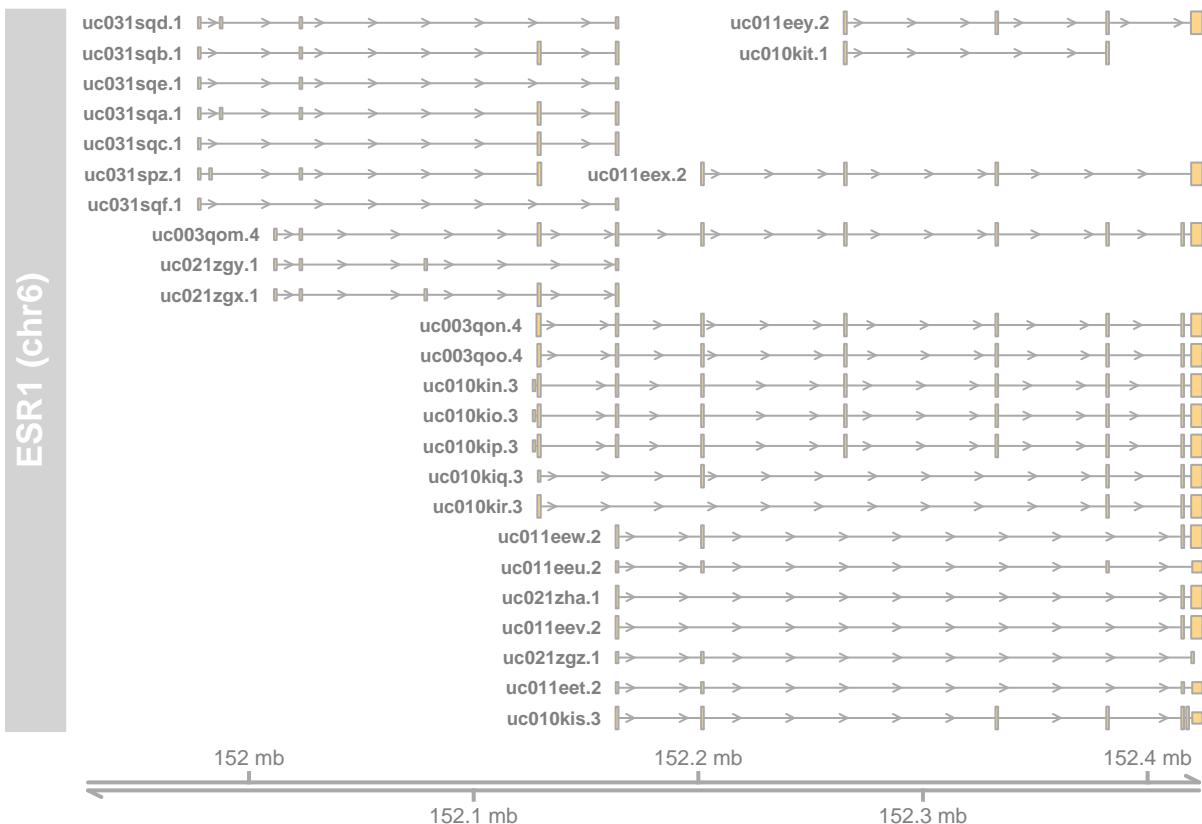
```
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
txdb = TxDb.Hsapiens.UCSC.hg19.knownGene
e_id <- select(edb, keys="ESR1", keytype="GENENAME", columns=c("ENTREZID"))[1, "ENTREZID"]
n_transcripts <- length(transcripts(txdb, filter=list(gene_id=e_id)))
paste("Number of transcripts comprimising model of ESR1: ", n_transcripts)
```

```
## [1] "Number of transcripts comprimising model of ESR1:  27"
```

AnnotationHub: finding and caching important information

- Central hub for genomic annotation files maintained by Bioconductor community
- Includes annotation files from UCSC, ENSEMBL, and the Broad Institute
- **AnnotationHub** allows you to search and download resources from inside R session

```
library(AnnotationHub)
```

```
## Loading required package: BiocFileCache
```

```
## Loading required package: dbplyr
```

```
##
## Attaching package: 'AnnotationHub'
```

```
## The following object is masked from 'package:Biobase':
##
##     cache
```

```
ah <- AnnotationHub()
```

```
## snapshotDate(): 2020-10-27
ah
```

```
## AnnotationHub with 57231 records
## # snapshotDate(): 2020-10-27
## # $dataprovider: Ensembl, BroadInstitute, UCSC, ftp://ftp.ncbi.nlm.nih.gov/g...
## # $species: Homo sapiens, Mus musculus, Drosophila melanogaster, Bos taurus,...
## # $rdataclass: GRanges, TwoBitFile, BigWigFile, EnsDb, Rle, OrgDb, ChainFile...
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass, tags,
## #   rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH5012"]]'
##
##             title
##   AH5012  | Chromosome Band
##   AH5013  | STS Markers
##   AH5014  | FISH Clones
##   AH5015  | Recomb Rate
##   AH5016  | ENCODE Pilot
##   ...       ...
##   AH91566 | Zonotrichia_albicollis.Zonotrichia_albicollis-1.0.1.ncrna.2bit
##   AH91567 | Zosterops_lateralis_melanops.ASM128173v1.cdna.all.2bit
##   AH91568 | Zosterops_lateralis_melanops.ASM128173v1.dna_rm.toplevel.2bit
##   AH91569 | Zosterops_lateralis_melanops.ASM128173v1.dna_sm.toplevel.2bit
##   AH91570 | Zosterops_lateralis_melanops.ASM128173v1.ncrna.2bit
```

```
length(unique(ah$species))
```

```
## [1] 2643
```

```
ah_human <- subset(ah, species == "Homo sapiens")
ah_human
```

```
## AnnotationHub with 26461 records
## # snapshotDate(): 2020-10-27
## # $dataprovider: BroadInstitute, UCSC, Ensembl, GENCODE, UWashington, Stanfo...
## # $species: Homo sapiens
## # $rdataclass: GRanges, BigWigFile, Rle, ChainFile, TwoBitFile, list, data.f...
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass, tags,
## #   rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH5012"]]'
##
##             title
##   AH5012  | Chromosome Band
##   AH5013  | STS Markers
##   AH5014  | FISH Clones
##   AH5015  | Recomb Rate
##   AH5016  | ENCODE Pilot
##   ...       ...
##   AH83216 | Ensembl 101 EnsDb for Homo sapiens
##   AH83362 | Sequences of snoRNA targets of Homo sapiens hg38
##   AH84122 | org.Hs.eg.db.sqlite
##   AH89180 | Ensembl 102 EnsDb for Homo sapiens
##   AH89426 | Ensembl 103 EnsDb for Homo sapiens
```

```r
query(ah, "HepG2")
```

```
## AnnotationHub with 440 records
## # snapshotDate(): 2020-10-27
## # $dataprovider: UCSC, BroadInstitute, Pazar
## # $species: Homo sapiens, NA
## # $rdataclass: GRanges, BigWigFile
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass, tags,
## #   rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH22246"]]'
##
##             title
##   AH22246 | pazar_CEBPA_HEPG2_Schmidt_20120522.csv
##   AH22249 | pazar_CTCF_HEPG2_Schmidt_20120522.csv
##   AH22273 | pazar_HNF4A_HEPG2_Schmidt_20120522.csv
##   AH22309 | pazar_STAG1_HEPG2_Schmidt_20120522.csv
##   AH22348 | wgEncodeAffyRnaChipFiltTransfragsHepg2CytosolLongnonpolya.broadP...
##   ...       ...
##   AH41564 | E118-H4K5ac.imputed.pval.signal.bigwig
##   AH41691 | E118-H4K8ac.imputed.pval.signal.bigwig
##   AH41818 | E118-H4K91ac.imputed.pval.signal.bigwig
##   AH46971 | E118_15_coreMarks_mnemonics.bed.gz
##   AH49484 | E118_RRBS_FractionalMethylation.bigwig
```

```r
query(ah, c("HepG2", "H3K4me3"))
```

```
## AnnotationHub with 11 records
## # snapshotDate(): 2020-10-27
## # $dataprovider: BroadInstitute, UCSC
## # $species: Homo sapiens
## # $rdataclass: GRanges, BigWigFile
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass, tags,
## #   rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH23311"]]'
##
##             title
##   AH23311 | wgEncodeBroadHistoneHepg2H3k4me3StdPk.broadPeak.gz
##   AH27201 | wgEncodeUwHistoneHepg2H3k4me3StdHotspotsRep1.broadPeak.gz
##   AH27202 | wgEncodeUwHistoneHepg2H3k4me3StdHotspotsRep2.broadPeak.gz
##   AH27203 | wgEncodeUwHistoneHepg2H3k4me3StdPkRep1.narrowPeak.gz
##   AH27204 | wgEncodeUwHistoneHepg2H3k4me3StdPkRep2.narrowPeak.gz
##   ...       ...
##   AH30771 | E118-H3K4me3.narrowPeak.gz
##   AH31712 | E118-H3K4me3.gappedPeak.gz
##   AH32893 | E118-H3K4me3.fc.signal.bigwig
##   AH33925 | E118-H3K4me3.pval.signal.bigwig
##   AH40296 | E118-H3K4me3.imputed.pval.signal.bigwig
```

```r
hepg2 <- query(ah, "HepG2")
hepg2_h3k4me3 <- query(hepg2, c("H3k4me3"))
hepg2_h3k4me3
```

```
## AnnotationHub with 11 records
```

```
## # snapshotDate(): 2020-10-27
## # $dataprovider: BroadInstitute, UCSC
## # $species: Homo sapiens
## # $rdataclass: GRanges, BigWigFile
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass, tags,
## #   rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH23311"]]'
##
##             title
##   AH23311 | wgEncodeBroadHistoneHepg2H3k4me3StdPk.broadPeak.gz
##   AH27201 | wgEncodeUwHistoneHepg2H3k4me3StdHotspotsRep1.broadPeak.gz
##   AH27202 | wgEncodeUwHistoneHepg2H3k4me3StdHotspotsRep2.broadPeak.gz
##   AH27203 | wgEncodeUwHistoneHepg2H3k4me3StdPkRep1.narrowPeak.gz
##   AH27204 | wgEncodeUwHistoneHepg2H3k4me3StdPkRep2.narrowPeak.gz
##   ...       ...
##   AH30771 | E118-H3K4me3.narrowPeak.gz
##   AH31712 | E118-H3K4me3.gappedPeak.gz
##   AH32893 | E118-H3K4me3.fc.signal.bigwig
##   AH33925 | E118-H3K4me3.pval.signal.bigwig
##   AH40296 | E118-H3K4me3.imputed.pval.signal.bigwig
```

```r
hepg2_h3k4me3$tags
```

```
##  [1] "wgEncode, ChipSeq, broadPeak, HepG2 cell, Bernstein grant"
##  [2] "wgEncode, ChipSeq, broadPeak, HepG2 cell, Stam grant"
##  [3] "wgEncode, ChipSeq, broadPeak, HepG2 cell, Stam grant"
##  [4] "wgEncode, ChipSeq, narrowPeak, HepG2 cell, Stam grant"
##  [5] "wgEncode, ChipSeq, narrowPeak, HepG2 cell, Stam grant"
##  [6] "EpigenomeRoadMap, peaks, consolidated, broadPeak, E118, ENCODE2012, LIV.HEPG2.CNCR, HepG2 Hepa
##  [7] "EpigenomeRoadMap, peaks, consolidated, narrowPeak, E118, ENCODE2012, LIV.HEPG2.CNCR, HepG2 Hepa
##  [8] "EpigenomeRoadMap, peaks, consolidated, gappedPeak, E118, ENCODE2012, LIV.HEPG2.CNCR, HepG2 Hepa
##  [9] "EpigenomeRoadMap, signal, consolidated, macs2signal, E118, ENCODE2012, LIV.HEPG2.CNCR, HepG2 He
## [10] "EpigenomeRoadMap, signal, consolidated, macs2signal, E118, ENCODE2012, LIV.HEPG2.CNCR, HepG2 He
## [11] "EpigenomeRoadMap, signal, consolidatedImputed, H3K4me3, E118, ENCODE2012, LIV.HEPG2.CNCR, HepG2
```

```r
# display(query(ah, "HepG2"))

e118_broadpeak <- query(hepg2_h3k4me3, c("E118", "broadPeak"))
id <- e118_broadpeak$ah_id
id
```

```
## [1] "AH29728"
```

```r
hepg2_h3k4me3_broad <- ah[["AH29728"]]
```

```
## loading from cache
```

```r
hepg2_h3k4me3_broad
```

```
## GRanges object with 60638 ranges and 5 metadata columns:
##           seqnames            ranges strand |        name     score signalValue
##              <Rle>         <IRanges>  <Rle> | <character> <numeric>   <numeric>
##       [1]    chr14 24614467-24618166      * |      Rank_1       850     20.3233
##       [2]    chr20   3183140-3185609      * |      Rank_2       830     25.7534
##       [3]    chr14 24700096-24704098      * |      Rank_3       811     17.2931
##       [4]    chr14 24766070-24770499      * |      Rank_4       763     18.9677
```

```
##     [5]     chr20 44420138-44421910       * |      Rank_5       755      24.0763
##     ...       ...                  ...     ... .       ...       ...         ...
## [60634]      chr2 11928736-11929617       * |  Rank_60634         0      1.73093
## [60635]     chr10 97229724-97230412       * |  Rank_60635         0      1.73015
## [60636]      chr2 39896310-39896946       * |  Rank_60636         0      1.73014
## [60637]      chr6   3978391-3978677       * |  Rank_60637         0      1.73015
## [60638]      chr6 49433554-49434110       * |  Rank_60638         0      1.73014
##            pValue    qValue
##         <numeric> <numeric>
##     [1]   88.3475   85.0287
##     [2]   86.2138   83.0301
##     [3]   84.3213   81.1706
##     [4]   79.3876   76.3449
##     [5]   78.6304   75.5947
##     ...       ...       ...
## [60634]   1.00441         0
## [60635]   1.00357         0
## [60636]   1.00357         0
## [60637]   1.00357         0
## [60638]   1.00357         0
##   -------
##   seqinfo: 298 sequences (2 circular) from hg19 genome
```

```
alt_format <- ah[[id]]
```

```
## loading from cache
```

```
identical(hepg2_h3k4me3_broad, alt_format)
```

```
## [1] TRUE
```

Assessment: AnnotationHub

```
library(AnnotationHub)
ah = AnnotationHub()
```

```
## snapshotDate(): 2020-10-27
```

```
mah = mcols(ah)
names(mah)
```

```
##  [1] "title"             "dataprovider"      "species"
##  [4] "taxonomyid"        "genome"            "description"
##  [7] "coordinate_1_based" "maintainer"       "rdatadateadded"
## [10] "preparerclass"     "tags"              "rdataclass"
## [13] "rdatapath"         "sourceurl"         "sourcetype"
```

```
sort(table(mah$species), decreasing=TRUE)[1:10]
```

```
##
##           Homo sapiens            Mus musculus Drosophila melanogaster
##                  26461                    1617                     422
##             Bos taurus         Pan troglodytes       Rattus norvegicus
##                    318                     306                     305
##            Danio rerio           Gallus gallus   Monodelphis domestica
##                    297                     265                     242
```

```
##            Felis catus
##                     235
n_ctcf_binding_hepg2 <- length(names(query(query(ah, "HepG2"), "CTCF")))
paste("Number of entries addressing CTCF binding in HepG2: ", n_ctcf_binding_hepg2)

## [1] "Number of entries addressing CTCF binding in HepG2:  13"
```

liftOver: Translating between reference builds

- Genomic annotations typically defined for fixed genome build
- Human is often hg19
- When analysis is performed on different genome build, annotations must be translated to the coordinates of the new build before use
- Process of translating called **lifting**
- Implemented in **liftOver()** function of **rtracklayer** Bioconductor package
- Tutorial will move features from genome build hg38 -> hg19

```
# liftOver from rtracklayer
library(rtracklayer)
?liftOver

# chromosome 1 gene locations in hg38
library(TxDb.Hsapiens.UCSC.hg38.knownGene)
tx38 <- TxDb.Hsapiens.UCSC.hg38.knownGene
seqlevels(tx38, pruning.mode="coarse") = "chr1"
g1_38 <- genes(tx38)
```

```
##   12 genes were dropped because they have exons located on both strands
##   of the same reference sequence or on more than one reference sequence,
##   so cannot be represented by a single genomic range.
##   Use 'single.strand.genes.only=FALSE' to get all the genes in a
##   GRangesList object, or use suppressMessages() to suppress this message.
# Download hg38 to hg19 chain file
library(AnnotationHub)
ah <- AnnotationHub()
```

```
## snapshotDate(): 2020-10-27
ah.chain <- subset(ah, rdataclass == "ChainFile" & species == "Homo sapiens")
query(ah.chain, c("hg19", "hg38"))
```

```
## AnnotationHub with 4 records
## # snapshotDate(): 2020-10-27
## # $dataprovider: UCSC, NCBI
## # $species: Homo sapiens
## # $rdataclass: ChainFile
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass, tags,
## #   rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH14108"]]'
##
##            title
##   AH14108 | hg38ToHg19.over.chain.gz
##   AH14150 | hg19ToHg38.over.chain.gz
```

```
##    AH78915 | Chain file for Homo sapiens rRNA hg19 to hg38
##    AH78916 | Chain file for Homo sapiens rRNA hg38 to hg19
```

```r
ch <- ah [["AH14108"]]
```

```
## loading from cache
```

```r
# perform the liftOver
g1_19L <- liftOver(g1_38, ch)
g1_19L
```

```
## GRangesList object of length 2696:
## $`10000`
## GRanges object with 1 range and 1 metadata column:
##       seqnames              ranges strand |     gene_id
##          <Rle>           <IRanges>  <Rle> | <character>
##   [1]     chr1 243651535-244014381      - |       10000
##   -------
##   seqinfo: 19 sequences from an unspecified genome; no seqlengths
##
## $`100034743`
## GRanges object with 3 ranges and 1 metadata column:
##       seqnames              ranges strand |     gene_id
##          <Rle>           <IRanges>  <Rle> | <character>
##   [1]     chr1 147466094-147484530      - |   100034743
##   [2]     chr1 147484532-147484551      - |   100034743
##   [3]     chr1 147484553-147487188      - |   100034743
##   -------
##   seqinfo: 19 sequences from an unspecified genome; no seqlengths
##
## $`100126331`
## GRanges object with 1 range and 1 metadata column:
##       seqnames              ranges strand |     gene_id
##          <Rle>           <IRanges>  <Rle> | <character>
##   [1]     chr1 117637265-117637350      + |   100126331
##   -------
##   seqinfo: 19 sequences from an unspecified genome; no seqlengths
##
## ...
## <2693 more elements>
```

Assessment: liftOver

```r
if(!file.exists("hg19ToHg38.over.chain")){
  download.file("http://hgdownload.cse.ucsc.edu/goldenPath/hg19/liftOver/hg19ToHg38.over.chain.gz", "hg
  library(R.utils)
  gunzip("hg19ToHg38.over.chain.gz")
}

library(ERBS)
data(HepG2)
library(rtracklayer)
ch = import.chain("hg19ToHg38.over.chain")
nHepG2 = liftOver(HepG2, ch)
```

```r
s1 <- start(HepG2[1])
s2 <- start(nHepG2[1])[[1]]

abs_diff_bases <- abs(s2 - s1)
paste("Number of bases moved upstream in first range of HepG2 to hg38: ", abs_diff_bases)
```

```
## [1] "Number of bases moved upstream in first range of HepG2 to hg38:  199761"
```

- **rtracklayer** package parses data into common formats so they can easily be used as annotations in future analysis

```r
library(devtools)
install_github("genomicsclass/ERBS") # install ERBS package
```

```
## Skipping install of 'ERBS' from a github remote, the SHA1 (9f16eb6a) has not changed since last insta
##   Use `force = TRUE` to force installation
```

```r
f1 = dir(system.file("extdata",package="ERBS"), full=TRUE)[1] # access dat a
readLines(f1, 4) # preview a few lines
```

```
## [1] "chrX\t1509354\t1512462\t5\t0\t.\t157.92\t310\t32.000000\t1991"
## [2] "chrX\t26801421\t26802448\t6\t0\t.\t147.38\t310\t32.000000\t387"
## [3] "chr19\t11694101\t11695359\t1\t0\t.\t99.71\t311.66\t32.000000\t861"
## [4] "chr19\t4076892\t4079276\t4\t0\t.\t84.74\t310\t32.000000\t1508"
```

```r
library(rtracklayer)
imp = import(f1, format="bedGraph") # import as bedGraph format
imp
```

```
## GRanges object with 1873 ranges and 7 metadata columns:
##            seqnames              ranges strand |     score      NA.      NA.1
##               <Rle>           <IRanges>  <Rle> | <numeric> <integer> <logical>
##      [1]       chrX   1509355-1512462      * |         5         0      <NA>
##      [2]       chrX 26801422-26802448      * |         6         0      <NA>
##      [3]      chr19 11694102-11695359      * |         1         0      <NA>
##      [4]      chr19   4076893-4079276      * |         4         0      <NA>
##      [5]       chr3 53288568-53290767      * |         9         0      <NA>
##      ...        ...                 ...    ... .       ...       ...       ...
##   [1869]      chr19 11201120-11203985      * |      8701         0      <NA>
##   [1870]      chr19   2234920-2237370      * |       990         0      <NA>
##   [1871]       chr1 94311336-94313543      * |      4035         0      <NA>
##   [1872]      chr19 45690614-45691210      * |     10688         0      <NA>
##   [1873]      chr19   6110100-6111252      * |      2274         0      <NA>
##                NA.2      NA.3      NA.4      NA.5
##           <numeric> <numeric> <numeric> <integer>
##      [1]    157.92   310.000        32      1991
##      [2]    147.38   310.000        32       387
##      [3]     99.71   311.660        32       861
##      [4]     84.74   310.000        32      1508
##      [5]     78.20   299.505        32      1772
##      ...       ...       ...       ...       ...
##   [1869]      8.65     7.281   0.26576      2496
##   [1870]      8.65    26.258   1.99568      1478
##   [1871]      8.65    12.511   1.47237      1848
##   [1872]      8.65     6.205   0.00000       298
##   [1873]      8.65    17.356   2.01323       496
```

```
##    -------
##   seqinfo: 23 sequences from an unspecified genome; no seqlengths
genome(imp) # genome identifier tag not set, but can be set manually

##  chrX chr19  chr3 chr17  chr8 chr11 chr16  chr1  chr2  chr6  chr9  chr7  chr5
##    NA    NA    NA    NA    NA    NA    NA    NA    NA    NA    NA    NA    NA
## chr12 chr20 chr21 chr22 chr18 chr10 chr14 chr15  chr4 chr13
##    NA    NA    NA    NA    NA    NA    NA    NA    NA    NA

genome(imp) = "hg19"
genome(imp)

##    chrX   chr19    chr3   chr17    chr8   chr11   chr16    chr1    chr2    chr6    chr9
## "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"
##    chr7    chr5   chr12   chr20   chr21   chr22   chr18   chr10   chr14   chr15    chr4
## "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"  "hg19"
##   chr13
## "hg19"

export(imp, "demoex.bed") # export as BED format
cat(readLines("demoex.bed", n=5), sep="\n") # check output file

## chrX 1509354 1512462 .   5    .
## chrX 26801421    26802448    .   6    .
## chr19    11694101    11695359    .   1    .
## chr19    4076892 4079276 .   4    .
## chr3 53288567    53290767    .   9    .
```

Assessment: Import/export

```
library(rtracklayer)
data(targets)
c_targets <- class(targets)
paste("Class of targets: ", c_targets)

## [1] "Class of targets:  data.frame"

library(GenomicRanges)
mtar <- with(targets,
GRanges(chrom, IRanges(start,end), strand=strand,
targets=target, mirname=name))

cat(export(mtar[1:5], format="bed"), sep="\n")

## chr12    8985196 8985217 .   0    -
## chr7 117095439   117095461   .   0    +
## chr17    23750063    23750088    .   0    +
## chr7 27187934    27187957    .   0    -
## chr17    43458622    43458643    .   0    -

cat("\n")

cat(export(mtar[1:5], format="gff3"), sep="\n")

## ##gff-version 3
## ##source-version rtracklayer 1.50.0
```

```
## ##date 2021-04-14
## chr12    rtracklayer sequence_feature    8985197 8985217 .   -   .   targets=ENST00000000412;mirname=
## chr7 rtracklayer sequence_feature    117095440   117095461   .   +   .   targets=ENST00000003084;mir
## chr17    rtracklayer sequence_feature    23750064    23750088    .   +   .   targets=ENST00000003834
## chr7 rtracklayer sequence_feature    27187935    27187957    .   -   .   targets=ENST00000006015;mir
## chr17    rtracklayer sequence_feature    43458623    43458643    .   -   .   targets=ENST00000006101
```

OrgDb: unified organism-specific annotation for systems biology

- Approach to annotation in Bioconductor
- Org packages have the form Org - two letter abbreviation of organism
- Two-letter abbreviation of organization - entrezgene is the resource
- Sqlite-based package
- If you know something about a RefSeq or UniProt ID you can learn about what genes have been
  annotated to it

```r
# load human OrgDb and inspect available keys
library(org.Hs.eg.db)
org.Hs.eg.db
```

```
## OrgDb object:
## | DBSCHEMAVERSION: 2.1
## | Db type: OrgDb
## | Supporting package: AnnotationDbi
## | DBSCHEMA: HUMAN_DB
## | ORGANISM: Homo sapiens
## | SPECIES: Human
## | EGSOURCEDATE: 2020-Sep23
## | EGSOURCENAME: Entrez Gene
## | EGSOURCEURL: ftp://ftp.ncbi.nlm.nih.gov/gene/DATA
## | CENTRALID: EG
## | TAXID: 9606
## | GOSOURCENAME: Gene Ontology
## | GOSOURCEURL: http://current.geneontology.org/ontology/go-basic.obo
## | GOSOURCEDATE: 2020-09-10
## | GOEGSOURCEDATE: 2020-Sep23
## | GOEGSOURCENAME: Entrez Gene
## | GOEGSOURCEURL: ftp://ftp.ncbi.nlm.nih.gov/gene/DATA
## | KEGGSOURCENAME: KEGG GENOME
## | KEGGSOURCEURL: ftp://ftp.genome.jp/pub/kegg/genomes
## | KEGGSOURCEDATE: 2011-Mar15
## | GPSOURCENAME: UCSC Genome Bioinformatics (Homo sapiens)
## | GPSOURCEURL:
## | GPSOURCEDATE: 2020-Aug27
## | ENSOURCEDATE: 2020-Aug18
## | ENSOURCENAME: Ensembl
## | ENSOURCEURL: ftp://ftp.ensembl.org/pub/current_fasta
## | UPSOURCENAME: Uniprot
## | UPSOURCEURL: http://www.UniProt.org/
## | UPSOURCEDATE: Mon Oct  5 00:18:02 2020
##
## Please see: help('select') for usage information
```

```r
keytypes(org.Hs.eg.db)
```

```
##  [1] "ACCNUM"      "ALIAS"       "ENSEMBL"     "ENSEMBLPROT" "ENSEMBLTRANS"
##  [6] "ENTREZID"    "ENZYME"      "EVIDENCE"    "EVIDENCEALL" "GENENAME"
## [11] "GO"          "GOALL"       "IPI"         "MAP"         "OMIM"
## [16] "ONTOLOGY"    "ONTOLOGYALL" "PATH"        "PFAM"        "PMID"
## [21] "PROSITE"     "REFSEQ"      "SYMBOL"      "UCSCKG"      "UNIGENE"
## [26] "UNIPROT"
```

```r
# load GO.db and inspect available terms
library(GO.db)
allterms = keys(GO.db, keytype="TERM")
allterms[1:5]
```

```
## [1] "mitochondrion inheritance"
## [2] "mitochondrial genome maintenance"
## [3] "reproduction"
## [4] "high-affinity zinc transmembrane transporter activity"
## [5] "low-affinity zinc ion transmembrane transporter activity"
```

```r
# find GOID (gene ontology tag) for ribosome biogenesis
goid <- select(GO.db,  keys = "ribosome biogenesis", keytype="TERM", columns="GOID")[,"GOID"]
```

```
## 'select()' returned 1:1 mapping between keys and columns
```

```r
# find symbols for genes involved in ribosome biogenesis
select(org.Hs.eg.db, keys=goid, keytype="GO", columns="SYMBOL")
```

```
## 'select()' returned 1:many mapping between keys and columns
```

```
##             GO EVIDENCE ONTOLOGY  SYMBOL
## 1  GO:0042254      ISS       BP    BYSL
## 2  GO:0042254      IMP       BP    GLUL
## 3  GO:0042254      IBA       BP     NVL
## 4  GO:0042254      IMP       BP     NVL
## 5  GO:0042254      IMP       BP   RPS28
## 6  GO:0042254      IMP       BP    XPO1
## 7  GO:0042254      IMP       BP   CUL4B
## 8  GO:0042254      IMP       BP   CUL4A
## 9  GO:0042254      IBA       BP    URB2
## 10 GO:0042254      IEA       BP   CEBPZ
## 11 GO:0042254      IEA       BP MYBBP1A
## 12 GO:0042254      IEA       BP    PWP1
## 13 GO:0042254       IC       BP    BOP1
## 14 GO:0042254      NAS       BP    BOP1
## 15 GO:0042254      IMP       BP  ZNF658
## 16 GO:0042254      IEA       BP    MTG2
## 17 GO:0042254      IEA       BP    AATF
## 18 GO:0042254      IEA       BP  DROSHA
## 19 GO:0042254      IEA       BP    GNL2
## 20 GO:0042254      IEA       BP   GNL3L
## 21 GO:0042254      IEA       BP    RRN3
## 22 GO:0042254      IDA       BP   DHX37
## 23 GO:0042254      IBA       BP   DDX31
## 24 GO:0042254      IMP       BP   DDX31
## 25 GO:0042254      IBA       BP  MRPL36
## 26 GO:0042254      IEA       BP    RIOX2
```

```
## 27 GO:0042254      IEA      BP GTPBP10
## 28 GO:0042254      IBA      BP   NAF1
## 29 GO:0042254      IDA      BP   NAF1
## 30 GO:0042254      IEA      BP MRPL10
## 31 GO:0042254      IBA      BP   RBIS
## 32 GO:0042254      IMP      BP   RBIS
# you can pull out multiple columns at once
e_id <- select(org.Hs.eg.db, keys = "GO:0042254", keytype="GO", columns=c("SYMBOL", "ENTREZID"))
```

```
## 'select()' returned 1:many mapping between keys and columns
```

```
entrezid <- unlist(e_id[e_id["SYMBOL"] == "ZNF658", "ENTREZID"])
```

```
# find gene ontology tags for related to ZNF658, which has the specified ENTREZID
select(org.Hs.eg.db, keys=entrezid, keytype="ENTREZID", columns="GO")
```

```
## 'select()' returned 1:many mapping between keys and columns
```

```
##    ENTREZID         GO EVIDENCE ONTOLOGY
## 1     26149 GO:0000976      IDA       MF
## 2     26149 GO:0000978      IBA       MF
## 3     26149 GO:0001228      IBA       MF
## 4     26149 GO:0005634      IBA       CC
## 5     26149 GO:0005634      IEA       CC
## 6     26149 GO:0006357      IBA       BP
## 7     26149 GO:0042254      IMP       BP
## 8     26149 GO:0045892      IMP       BP
## 9     26149 GO:0046872      IEA       MF
## 10    26149 GO:0071294      IMP       BP
# save GO tags to a character vector
select(org.Hs.eg.db, keys=entrezid, keytype="ENTREZID", columns="GO")$"GO"
```

```
## 'select()' returned 1:many mapping between keys and columns
```

```
##  [1] "GO:0000976" "GO:0000978" "GO:0001228" "GO:0005634" "GO:0005634"
##  [6] "GO:0006357" "GO:0042254" "GO:0045892" "GO:0046872" "GO:0071294"
```

```
myk = unlist(.Last.value)
```

```
# identify biological processes ZNF658 is involved in
#select(GO.db, keys=myk, columns="TERM")
```

Assessment: orgDb

```
library(org.Hs.eg.db)
org.Hs.eg.db
```

```
## OrgDb object:
## | DBSCHEMAVERSION: 2.1
## | Db type: OrgDb
## | Supporting package: AnnotationDbi
## | DBSCHEMA: HUMAN_DB
## | ORGANISM: Homo sapiens
## | SPECIES: Human
```

```
## | EGSOURCEDATE: 2020-Sep23
## | EGSOURCENAME: Entrez Gene
## | EGSOURCEURL: ftp://ftp.ncbi.nlm.nih.gov/gene/DATA
## | CENTRALID: EG
## | TAXID: 9606
## | GOSOURCENAME: Gene Ontology
## | GOSOURCEURL: http://current.geneontology.org/ontology/go-basic.obo
## | GOSOURCEDATE: 2020-09-10
## | GOEGSOURCEDATE: 2020-Sep23
## | GOEGSOURCENAME: Entrez Gene
## | GOEGSOURCEURL: ftp://ftp.ncbi.nlm.nih.gov/gene/DATA
## | KEGGSOURCENAME: KEGG GENOME
## | KEGGSOURCEURL: ftp://ftp.genome.jp/pub/kegg/genomes
## | KEGGSOURCEDATE: 2011-Mar15
## | GPSOURCENAME: UCSC Genome Bioinformatics (Homo sapiens)
## | GPSOURCEURL:
## | GPSOURCEDATE: 2020-Aug27
## | ENSOURCEDATE: 2020-Aug18
## | ENSOURCENAME: Ensembl
## | ENSOURCEURL: ftp://ftp.ensembl.org/pub/current_fasta
## | UPSOURCENAME: Uniprot
## | UPSOURCEURL: http://www.UniProt.org/
## | UPSOURCEDATE: Mon Oct  5 00:18:02 2020

##
## Please see: help('select') for usage information
```

```r
keytypes(org.Hs.eg.db)
```

```
##  [1] "ACCNUM"       "ALIAS"        "ENSEMBL"      "ENSEMBLPROT"  "ENSEMBLTRANS"
##  [6] "ENTREZID"     "ENZYME"       "EVIDENCE"     "EVIDENCEALL"  "GENENAME"
## [11] "GO"           "GOALL"        "IPI"          "MAP"          "OMIM"
## [16] "ONTOLOGY"     "ONTOLOGYALL"  "PATH"         "PFAM"         "PMID"
## [21] "PROSITE"      "REFSEQ"       "SYMBOL"       "UCSCKG"       "UNIGENE"
## [26] "UNIPROT"
```

```r
genes <- select(org.Hs.eg.db, key="17q21.1", keytype="MAP", columns=c("GENENAME", "ENTREZID"))
```

```
## 'select()' returned 1:many mapping between keys and columns
```

```r
num_genes <- nrow(genes)
paste("Number of genes present on 17q21.1: ", num_genes)
```

```
## [1] "Number of genes present on 17q21.1:  21"
```

```r
genes_with_go_tag <- select(org.Hs.eg.db, key="17q21.1", keytype="MAP", columns=c("GENENAME", "ENTREZID
```

```
## 'select()' returned 1:many mapping between keys and columns
```

```r
library(plyr)
```

```
##
## Attaching package: 'plyr'

## The following object is masked from 'package:XVector':
##
##     compact

## The following object is masked from 'package:IRanges':
```

```
##
##      desc

## The following object is masked from 'package:S4Vectors':
##
##      rename

counts <- count(genes_with_go_tag, "GO")
top_five_go_id <- counts[order(-counts$freq),][1:5,]


go_annotations_for_ormdl3 <- select(org.Hs.eg.db, key="ORMDL3", keytype="SYMBOL", columns=c("EVIDENCE",

## 'select()' returned 1:many mapping between keys and columns

num_tas_evidence_codes <- length(go_annotations_for_ormdl3[go_annotations_for_ormdl3$EVIDENCE == "TAS",]


paste("number of GO annotations for ORMDL3 having TAS (traceable author statement) as their evidence co

## [1] "number of GO annotations for ORMDL3 having TAS (traceable author statement) as their evidence c
```

Assessment: Interactive tables for genomic annotation

```
library(Homo.sapiens)
g = genes(Homo.sapiens)

##    403 genes were dropped because they have exons located on both strands
##    of the same reference sequence or on more than one reference sequence,
##    so cannot be represented by a single genomic range.
##    Use 'single.strand.genes.only=FALSE' to get all the genes in a
##    GRangesList object, or use suppressMessages() to suppress this message.

library(ERBS)
data(HepG2)


kp = g[resize(g,1) %over% HepG2]

nn = names(kp)
m = select(Homo.sapiens, keys=nn, keytype="ENTREZID",
           columns=c("SYMBOL", "GENENAME", "TERM", "GO"))

## 'select()' returned 1:many mapping between keys and columns
#library(DT)
#datatable(m)
```

Using Kyoto Encyclopedia of Genes and Genomes (KEGG)

- Detailed definitions that go beyond term to characterize gene ontology
- More advanced material to think about structure of relationship between terms - details on sqlite representation
- Interface to kyoto encyclopedia of genes and genomes (KEGG) - REST package without serializing, just issuing queries
- Generates information about organism-specific pathways that are defined (must know 3-letter prefix)
- Different types of entities that can be returned - numerical code for genes given prefix, colon, number

- Instead if possess prefix called path + organism-specific path code we can get information about pathway
- Can find list of genes that are annotated
- If we are interested in a pathway and need a gene list, can get this
- Can also get diagram which indicates structure of network
- Colored boxes most likely refer to genes, could be modifications of genes
- Sub-pathways identified and use to understand the nature of relationships between different genes

```
# load KEGGREST package and inspect organism-specific gene pathways
library(KEGGREST)
brca2K = keggGet("hsa:675")     # reference to a specific gene
names(brca2K[[1]])
```

```
##  [1] "ENTRY"      "NAME"      "DEFINITION" "ORTHOLOGY"  "ORGANISM"
##  [6] "PATHWAY"    "DISEASE"   "BRITE"      "POSITION"   "MOTIF"
## [11] "DBLINKS"    "STRUCTURE" "AASEQ"      "NTSEQ"
```

```
brpat = keggGet("path:hsa05212")     # info on a pathway
brpat[[1]]$GENE[seq(1,132,2)] # entrez gene ids for pathway
```

```
##  [1] "3845"  "5290"  "5293"  "5291"  "5295"  "5296"  "8503"  "9459"  "5879"
## [10] "5880"  "5881"  "4790"  "5970"  "207"   "208"   "10000" "1147"  "3551"
## [19] "8517"  "572"   "598"   "842"   "369"   "673"   "5894"  "5604"  "5594"
## [28] "5595"  "5599"  "5602"  "5601"  "5900"  "5898"  "5899"  "10928" "998"
## [37] "5337"  "5338"  "7039"  "1950"  "1956"  "2064"  "2475"  "6198"  "6199"
## [46] "3716"  "6774"  "6772"  "7422"  "1029"  "1019"  "1021"  "595"   "5925"
## [55] "1869"  "1870"  "1871"  "7157"  "1026"  "1647"  "4616"  "10912" "581"
## [64] "578"   "1643"  "51426"
```

```
# inspect some entrez ids
select(org.Hs.eg.db, keys="5888", keytype="ENTREZID", columns ="SYMBOL")
```

```
## 'select()' returned 1:1 mapping between keys and columns
```
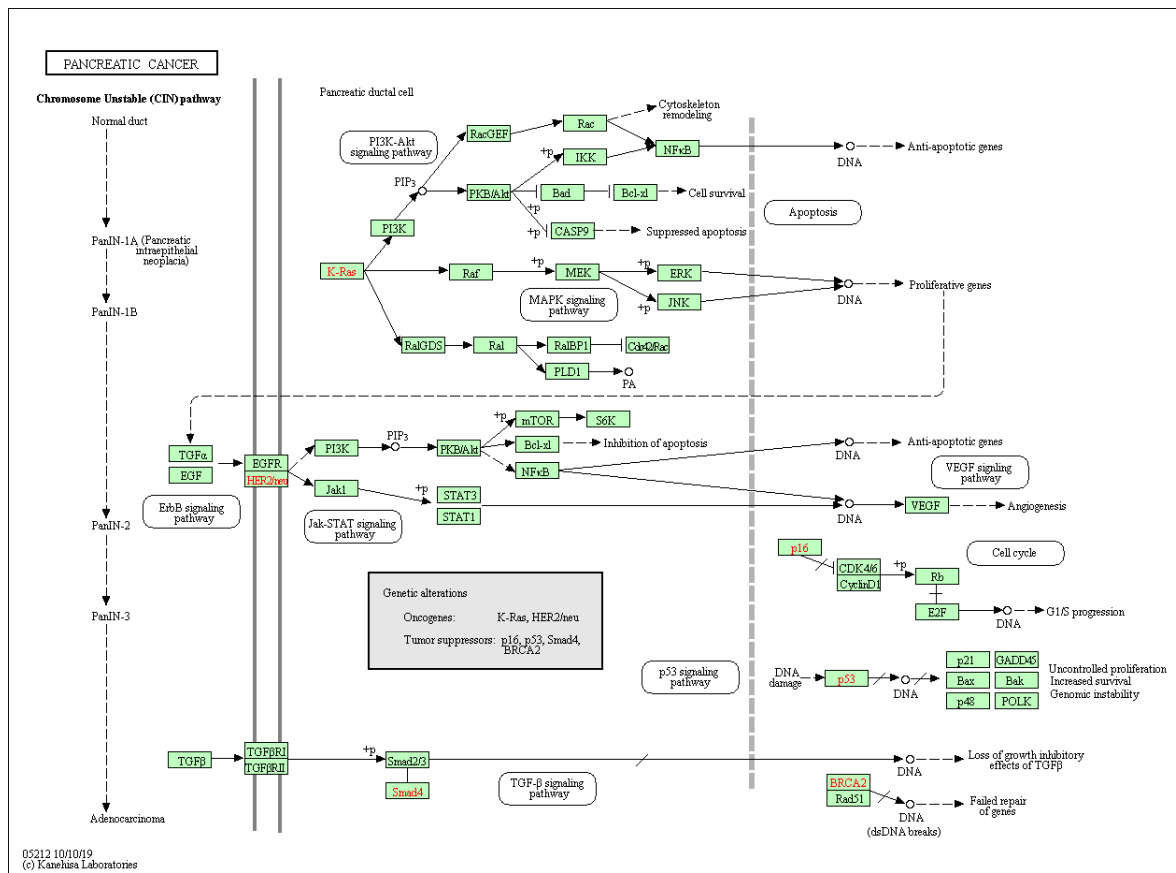
```
##   ENTREZID SYMBOL
## 1     5888  RAD51
```

```
select(org.Hs.eg.db, keys="675", keytype="ENTREZID", columns ="SYMBOL")
```

```
## 'select()' returned 1:1 mapping between keys and columns
```

```
##   ENTREZID SYMBOL
## 1      675  BRCA2
```

```
# diagram showing structure of network
library(png)
library(grid)
brpng = keggGet("hsa05212", "image")
grid.raster(brpng)
```

Assessment: KEGG

```r
k_id <- keggGet("hsa:3845")
first_gene_name <- strsplit(k_id[[1]]$NAME, ",")[[1]][1]
paste("Name of gene associated with KEGG ID: ", first_gene_name)
```

```
## [1] "Name of gene associated with KEGG ID:  KRAS"
```

```r
library(png)
oo = keggGet("hsa00790", "image")
writePNG(oo, "images/hsa00790.png")
```

EBML's ontology lookup service

- Dealing with collection of ontologies
- Effort by European Bioinformatics Institute to unify access to many ontologies dealing with different topics
- Ontology useful concept - different concept types, anatomies, biological processes, molecular functions
- Common vocabulary not immediately established
- Ontology establishes list of terms and identifying relationships among them
- OBO foundry - existence and relatedness
- Different approaches to dealing with language vocabulary - OBO way into that
- **rols** R interface - staging process of finding out about ontologies and getting content out of them

- **olsSearch** function in rols to look up ontology relationships + sources that deal with the term Glioblastoma
- Find out more by visiting OLS site
- Formal tags for glioblastoma as phenotype from two ontologies
- Rich information in resources about terms and rols package helps deal with concepts

```
library(rols)
```

```
## Warning: package 'rols' was built under R version 4.0.4
```

```
##
## This is 'rols' version 2.18.3
```

```
oo = Ontologies()
oo
```

```
## Object of class 'Ontologies' with 263 entries
##     AEO, AFO ... XL, ZECO
```

```
oo[[1]]
```

```
## Ontology: Anatomical Entity Ontology (aeo)
##    AEO is an ontology of anatomical structures that expands CARO, the
##    Common Anatomy Reference Ontology
##     Loaded: 2020-08-14 Updated: 2021-04-14 Version: 2014-12-05
##     250 terms  26 properties  0 individuals
```

```
glis = OlsSearch("glioblastoma")
glis
```

```
## Object of class 'OlsSearch':
##   query: glioblastoma
##   requested: 20 (out of 617)
##   response(s): 0
```

```
res = olsSearch(glis)
resdf = as(res, "data.frame") # get content
resdf[1:4,1:4]
```

```
##                                                       id
## 1   ncit:class:http://purl.obolibrary.org/obo/NCIT_C3058
## 2 omit:class:http://purl.obolibrary.org/obo/OMIT_0007102
## 3   mpath:class:http://purl.obolibrary.org/obo/MPATH_564
## 4       ordo:class:http://www.orpha.net/ORDO/Orphanet_360
##                                        iri   short_form        obo_id
## 1   http://purl.obolibrary.org/obo/NCIT_C3058    NCIT_C3058    NCIT:C3058
## 2 http://purl.obolibrary.org/obo/OMIT_0007102 OMIT_0007102 OMIT:0007102
## 3   http://purl.obolibrary.org/obo/MPATH_564    MPATH_564    MPATH:564
## 4       http://www.orpha.net/ORDO/Orphanet_360 Orphanet_360 Orphanet:360
```

```
resdf[1,5]  # full description for one instance
```

```
## [1] "Glioblastoma"
```

Assessment: Ontology lookup

```
library(rols)
diab = OlsSearch("diabetes")
```

```
olsRows(allRows(diab))
```

```
## [1] 8107
```

```
fulld = olsSearch(allRows(diab))
adf = as(fulld, "data.frame")
sort(table(adf$ontology_name), decreasing=TRUE)[1:10]
```

```
##
##    clo  ncit  gexo  rexo  reto mondo   efo  ordo   cco  doid
## 5683   601   208   207   204   201   188   144   141    80
```

```
length(grep("oral glucose", adf$description))
```

```
## [1] 5
```

Verified Assessment: Genomic Annotation

```
library(BSgenome.Hsapiens.UCSC.hg19)
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
library(org.Hs.eg.db)
library(ensembldb)
library(EnsDb.Hsapiens.v75)
library(GO.db)
library(AnnotationHub)
library(rtracklayer)
library(KEGGREST)


library(BSgenome.Hsapiens.UCSC.hg19)
chr_of_interest <- c("chr6", "chr8", "chr19", "chr20", "chrM", "chrX", "chrY")
sort(seqlengths(Hsapiens)[chr_of_interest])
```

```
##      chrM     chr19      chrY     chr20      chr8      chrX      chr6
##     16571  59128983  59373566  63025520 146364022 155270560 171115067
```

```
ratio <- nchar(Hsapiens$chrX) / nchar(Hsapiens$chrY)
paste("Times larger X chromosome is compared to Y: ", ratio)
```

```
## [1] "Times larger X chromosome is compared to Y:  2.61514627570121"
```

```
library(TxDb.Hsapiens.UCSC.hg19.knownGene)
txdb <- TxDb.Hsapiens.UCSC.hg19.knownGene    # abbreviate

library(org.Hs.eg.db)
e_id <- select(org.Hs.eg.db, keys="TP53", keytype="SYMBOL", columns=c("ENTREZID"))[1, "ENTREZID"]
```

```
## 'select()' returned 1:1 mapping between keys and columns
```

```
e_id
```

```
## [1] "7157"
```

```
txdb_genes <- genes(txdb)
```

```
##   403 genes were dropped because they have exons located on both strands
##   of the same reference sequence or on more than one reference sequence,
```

```
##   so cannot be represented by a single genomic range.
##   Use 'single.strand.genes.only=FALSE' to get all the genes in a
##   GRangesList object, or use suppressMessages() to suppress this message.
```

```r
txdb_genes_mapped <- genes(txdb, filter=list(gene_id=e_id))
gid_mapping <- txdb_genes[txdb_genes$gene_id == e_id]

num_bp <- width(gid_mapping)
paste("Number of base pairs TP53: ", num_bp)
```

```
## [1] "Number of base pairs TP53:  25772"
```

```r
tp53_transcripts <- transcripts(txdb, filter=list(gene_id=e_id))
paste("Number of transcripts: ", length(tp53_transcripts))
```

```
## [1] "Number of transcripts:  17"
```

```r
ex = exons(txdb, filter=list(gene_id=e_id))
num_exons <- length(ex)
paste("Number of exons in TP53 gene: ", num_exons)
```

```
## [1] "Number of exons in TP53 gene:  21"
```

```r
rex = reduce(ex)
ex_width = sum(width(ex)) # bases in exons
gene_width = sum(width(gid_mapping)) # bases in genes
prop_in_exons <- ex_width/gene_width
paste("Number of bases in exons: ", ex_width)
```

```
## [1] "Number of bases in exons:  4620"
```

```r
paste("Proportion of bp in exons: ", prop_in_exons)
```

```
## [1] "Proportion of bp in exons:  0.179264317864349"
```

```r
library(ensembldb)
library(EnsDb.Hsapiens.v75)
edb <- EnsDb.Hsapiens.v75    # abbreviate

txs <- transcripts(edb, filter = GeneNameFilter("TP53"))
num_transcripts <- length(txs)

txs[1]$tx_cds_seq_start
```

```
## [1] 7565257
```

```r
width(txs[txs$tx_biotype == "processed_transcript"])
```

```
## [1] 866
```

```r
tp_go <- select(org.Hs.eg.db, keys="TP53", keytype="SYMBOL", columns=c("GO"))
```

```
## 'select()' returned 1:many mapping between keys and columns
```

```r
nrow(tp_go)
```

```
## [1] 209
```

```r
length(unique(tp_go$GO))
```

```
## [1] 177
```

```
length(unique(tp_go[tp_go$EVIDENCE == "IMP", "GO"]))
```

```
## [1] 40
```

```
sort(table(tp_go$GO), decreasing = TRUE)[1]
```

```
## GO:0000790
##          3
```

```
library(GO.db)
allterms = keys(GO.db, keytype="TERM")
allterms[1:5]
```

```
## [1] "mitochondrion inheritance"
## [2] "mitochondrial genome maintenance"
## [3] "reproduction"
## [4] "high-affinity zinc transmembrane transporter activity"
## [5] "low-affinity zinc ion transmembrane transporter activity"
```

```
term <- select(GO.db,  keys = "GO:0000981", keytype="GOID", columns="TERM")
```

```
## 'select()' returned 1:1 mapping between keys and columns
```

```
library(AnnotationHub)
ah <- AnnotationHub()
```

```
## snapshotDate(): 2020-10-27
```

```
ah.chain <- subset(ah, rdataclass == "ChainFile" & species == "Homo sapiens")
query(ah.chain, c("hg19", "hg38"))
```

```
## AnnotationHub with 4 records
## # snapshotDate(): 2020-10-27
## # $dataprovider: UCSC, NCBI
## # $species: Homo sapiens
## # $rdataclass: ChainFile
## # additional mcols(): taxonomyid, genome, description,
## #   coordinate_1_based, maintainer, rdatadateadded, preparerclass, tags,
## #   rdatapath, sourceurl, sourcetype
## # retrieve records with, e.g., 'object[["AH14108"]]'
##
##             title
##   AH14108 | hg38ToHg19.over.chain.gz
##   AH14150 | hg19ToHg38.over.chain.gz
##   AH78915 | Chain file for Homo sapiens rRNA hg19 to hg38
##   AH78916 | Chain file for Homo sapiens rRNA hg38 to hg19
```

```
ch <- ah[["AH14150"]]
```

```
## loading from cache
# perform the liftOver
g1_19L <- liftOver(txdb_genes_mapped, ch)
g1_19L
```

```
## GRangesList object of length 1:
## $`7157`
## GRanges object with 1 range and 1 metadata column:
##      seqnames          ranges strand |     gene_id
```

```
##          <Rle>        <IRanges>  <Rle> | <character>
##    [1]    chr17 7661779-7687550    - |        7157
##    -------
##    seqinfo: 1 sequence from an unspecified genome; no seqlengths
```

```
# load KEGGREST package and inspect organism-specific gene pathways
library(KEGGREST)
#brca2K = keggGet("hsa:675")    # reference to a specific gene
#names(brca2K[[1]])
brpat = keggGet("path:hsa04115")    # info on a pathway
brpat_genes <- brpat[[1]]$GENE
length(brpat_genes)/2
```

```
## [1] 73
```

```
cyclin_indices <- grep("cyclin", brpat_genes)
length(cyclin_indices)
```

```
## [1] 15
```

```
brpat_genes[cyclin_indices]
```

```
##  [1] "CDKN2A; cyclin dependent kinase inhibitor 2A [KO:K06621]"
##  [2] "CDKN1A; cyclin dependent kinase inhibitor 1A [KO:K06625]"
##  [3] "CCND1; cyclin D1 [KO:K04503]"
##  [4] "CCND2; cyclin D2 [KO:K10151]"
##  [5] "CCND3; cyclin D3 [KO:K10152]"
##  [6] "CDK4; cyclin dependent kinase 4 [KO:K02089] [EC:2.7.11.22]"
##  [7] "CDK6; cyclin dependent kinase 6 [KO:K02091] [EC:2.7.11.22]"
##  [8] "CCNE1; cyclin E1 [KO:K06626]"
##  [9] "CCNE2; cyclin E2 [KO:K06626]"
## [10] "CDK2; cyclin dependent kinase 2 [KO:K02206] [EC:2.7.11.22]"
## [11] "CCNB1; cyclin B1 [KO:K05868]"
## [12] "CCNB2; cyclin B2 [KO:K21770]"
## [13] "CDK1; cyclin dependent kinase 1 [KO:K02087] [EC:2.7.11.22 2.7.11.23]"
## [14] "CCNG1; cyclin G1 [KO:K10145]"
## [15] "CCNG2; cyclin G2 [KO:K10146]"
```

```
#brpat[[1]]$GENE[seq(1,132,2)] # entrez gene ids for pathway
```