Creating IGV HTML reports with tracktables

Thomas Carroll[1*]

[1] Bioinformatics Core, MRC Clinical Sciences Centre;

[*]thomas.carroll (at)imperial.ac.uk

May 12, 2015

Contents

1 The tracktables package

Visualising genomics data in genome browsers is a key step in both quality control and the initial interrogation of any hypothesis under investigation.

The organisation of large collections of genomics data (such as from large scale high-thoughput sequencing experiments) alongside their associated sample or experimental metadata allows for the rapid evaluation of patterns across experimental groups.

Broad's Intergrative Genome Viewer (IGV) provides a set of methods to make use of sample metadata when visualising genomics data. As well as identifying sample metadata within the genome browser,

this sample information can be used in IGV to group, sort and filter samples.

This use of sample metadata, alongside the ability to control IGV through ports, provides the desired framework to rapidly interrogate large cohorts of genomics data once the appropriate file structure is built.

The Tracktables package provides a set of tools to build IGV session files from data-frames of sample files and their associated metadata as well as produce IGV linked HTML reports for high-throughput visualisation of sample data in IGV.

2 Creating IGV sessions and HTML reports using tracktables

The two main functions within the `tracktables` package are the `MakeIGVSession()` function for creating IGV session XMLs and any associated sample metadata files and the `maketracktable()` function to create HTML pages containing the sample metadata and interval tables used to control IGV.

2.1 Creating input files for tracktables

`tracktables` functions require the user to provide both a data-frame of metadata information and a data-frame of the paths of sample files to be visualised in IGV.

These data-frames must both have one column named "SampleName" which contains unique sample IDs. This column will be used to match samples and only samples within both files will be included in the IGV session.

The remaining metadata SampleSheet columns may be user-defined but must all contain column titles. (See example below)

The FileSheet (with file paths) must contain the columns "SampleName", "bam","bigwig" and "interval". These columns may contain NA values when no relavant file is associated to a sample.

Here we create a small example SampleSheet (containing metadata) and FileSheet (containing file locations) from some example histone mark, RNA polymerase 2 and Ebf ChIP-seq.

```
library(tracktables)

fileLocations <- system.file("extdata",package="tracktables")

bigwigs <- dir(fileLocations,pattern="*.bw",full.names=TRUE)
intervals <- dir(fileLocations,pattern="*.bed",full.names=TRUE)
bigWigMat <- cbind(gsub("_Example.bw","",basename(bigwigs)),
                   bigwigs)
intervalsMat <- cbind(gsub("_Peaks.bed","",basename(intervals)),
                      intervals)
```

```
FileSheet <- merge(bigWigMat,intervalsMat,all=TRUE)
FileSheet <- as.matrix(cbind(FileSheet,NA))
colnames(FileSheet) <- c("SampleName","bigwig","interval","bam")

SampleSheet <- cbind(as.vector(FileSheet[,"SampleName"]),
                     c("EBF","H3K4me3","H3K9ac","RNAPol2"),
                     c("ProB","ProB","ProB","ProB"))
colnames(SampleSheet) <- c("SampleName","Antibody","Species")
```

The SampleSheet contains a small section of metadata for the EBF, RNApol2, H3K4me3 and H3K9ac ChIP. The "SampleName" column contains the unique IDs.

```
head(SampleSheet)
```

```
##      SampleName Antibody  Species
## [1,] "EBF"      "EBF"     "ProB"
## [2,] "H3K4me3"  "H3K4me3" "ProB"
## [3,] "H3K9ac"   "H3K9ac"  "ProB"
## [4,] "RNAPol2"  "RNAPol2" "ProB"
```

The FileSheet contains the "SampleName" column with unique IDs matching those founds in the SampleSheet. The remaining columns are "bam","bigwig" and "interval" and list the full paths of relevant files to be displayed in IGV.

```
head(FileSheet)
```

```
##      SampleName bigwig                       interval                bam
## [1,] "EBF"      "pathTo/EBF_Example.bw"     "pathTo/EBF_Peaks.bed" NA
## [2,] "H3K4me3"  "pathTo/H3K4me3_Example.bw" NA                     NA
## [3,] "H3K9ac"   "pathTo/H3K9ac_Example.bw"  NA                     NA
## [4,] "RNAPol2"  "pathTo/RNAPol2_Example.bw" NA                     NA
```

Note that not all samples have intervals associated to them and ,here, none of these samples have BAM files associated to them. NA values within the FileSheet will be ignored by `tracktables` functions.

2.2 Creating an IGV session

`tracktables` can create an IGV session XML and associated sample information file from this SampleSheet and FileSheet.

In addition to the FileSheet and SampleSheet, the `MakeIGVSession()` function requires the location to write to, the filename for the session XML and the genome to be used in IGV (see IGV for details on supported genomes).

```
MakeIGVSession(SampleSheet,FileSheet,igvdirectory=getwd(),"Example","mm9")
```

This creates two files in the current working directory containing the sample information file for IGV ("SampleMetadata.txt") and the session XML itself to be loaded into IGV ("Example.xml").

2.3 Creating a Tracktable HTML report

As well as producing session XMLs and sample information files, the `tracktables` package can produce HTML reports which contain metadata and methods to control IGV.

The report structure is made of a main **Tracktables Experiment Report** which houses the metadata from the SampleSheet data-frame and links to open a sample's files in IGV (the sample's bigwig, bam or interval files). All sample files are associated with their relevant sample metadata and grouped together by their unique sample name.

When a sample has an interval file associated to it, the Tracktables Experiment Report also contains a link to a further sample specific **Tracktables Interval Report**. This interval report contains a table of interval locations, any metadata associated with intervals and further links to focus IGV on an interval's genomic location.

```
HTMLreport <- maketracktable(fileSheet=FileSheet,
                             SampleSheet=SampleSheet,
                             filename="IGVExample.html",
                             basedirectory=getwd(),
                             genome="mm9")
```

In this example the `maketracktables()` function creates an HTML report "IGVExample.html" (The Tracktable Experiment Report) in the current working directory alongside the relevant sample IGV session XMLs, The Tracktable Experiment Reports (named by SampleName) and the sample information file. The function also returns the XML doc to allow the user to add further customisation to the main report.

Further configuration of the report can be achieved through the use of the `colourBy` arguments and `igvParams` class object passed to the `maketracktables()` function. The `colourBy` argument accepts a character argument corresponding to the metadata column by which samples will be coloured in IGV.

The `igvParams` class defines how files will be displayed in IGV. `igvParams` accepts a range of arguments corresponding to supported IGV display methods (see reference manual for full details).

In this example, all files are colour grouped by their antibody metadata and display ranges sets to be between 1 and 5 for bigwigs files with no autoscale set.

```
igvDisplayParams <- igvParam(bigwig.autoScale = "false",
                             bigwig.minimum = 1,
                             bigwig.maximum = 5)


HTMLreport <- maketracktable(FileSheet,SampleSheet,"IGVex2.html",getwd(),"mm9",
                             colourBy="Antibody",
```

```
                                    igvParam=igvDisplayParams)
```

3 Using relative or absolute paths

`tracktables` allows for the use of both relative and absolute paths when creating IGV sessions and tracktables experiment reports with the `MakeIGVSession()`, `MakeIGVSessionXML()` and `maketracktable()` functions. The choice of relative or absolute paths for files and igv sessions is controlled by the use of `full.xml.paths` and `full.file.paths` arguments.

3.1 Creating a Tracktable HTML report using relative paths

In the examples in **Creating a Tracktable HTML report** and **Creating an IGV session** we have created a Tracktables Experiment Report and IGV session (XML and sample information files) using relative paths.

This allows for the report to be independent of a stable FTP or HTTP address as well as being highly portable and available offline (once IGV has been installed locally).

These report and session files however require the sample and XML files to be maintained in the same relative path to the report itself. To take advantage of the portability of the report, it is recommended that the files to be used are copied to a directory within the report directory and the entire report directory transferred as a unit.

In this example, first a new directory ("IGVDirectory") is created in the current working directory and all the files are copied to it.

```
library(tracktables)

oldFileLocations <- system.file("extdata",package="tracktables")

dir.create(file.path(getwd(),"IGVDirectory"),
           showWarnings = FALSE,recursive = TRUE)
file.copy(oldFileLocations,
          file.path(getwd(),"IGVDirectory"),
          recursive = TRUE)
fileLocations <- file.path(getwd(),"IGVDirectory","extdata")
```

Next the samplesheet of metadata and filesheet of locations is created.

```
bigwigs <- dir(fileLocations,pattern="*.bw",full.names=TRUE)
intervals <- dir(fileLocations,pattern="*.bed",full.names=TRUE)
bigWigMat <- cbind(gsub("_Example.bw","",basename(bigwigs)),
                   bigwigs)
intervalsMat <- cbind(gsub("_Peaks.bed","",basename(intervals)),
```

```
                      intervals)

FileSheet <- merge(bigWigMat,intervalsMat,all=TRUE)
FileSheet <- as.matrix(cbind(FileSheet,NA))
colnames(FileSheet) <- c("SampleName","bigwig","interval","bam")

SampleSheet <- cbind(as.vector(FileSheet[,"SampleName"]),
                     c("EBF","H3K4me3","H3K9ac","RNAPol2"),
                     c("ProB","ProB","ProB","ProB"))
colnames(SampleSheet) <- c("SampleName","Antibody","Species")
```

The tracktables report is created from a call to `maketracktable`. By default all paths are created relative the directory specified by `basedirectory`.

```
HTMLreport <- maketracktable(fileSheet=FileSheet,
                             SampleSheet=SampleSheet,
                             filename="IGVEx3.html",
                             basedirectory=file.path(getwd(),"IGVDirectory"),
                             genome="mm9")
```

By default the report and all XML files will be created in the directory specified by `basedirectory` argument.

The directory "IGVdirectory" now contains the tracktables experiment and intervals reports, all the IGV XML and sample information files as well as the sample files within the "IGVdirectory/extdata" directory.

This directory can used as a highly portable, self-contained report independent of external dependencies e.g. availability of online storage, internet connection.

3.2 Creating a Tracktable HTML report using absolute paths

When an FTP, HTTP or stable address to host sample and tracktables files is available, tracktables can be used to produce reports and xml files with absolute paths. This allows for the report itself to be hosted online or used locally while referencing files and links on the user defined ftp/http address.

The `full.xml.paths` and `full.file.paths` functions control how paths are defined with tracktables report and for online storage of tracktables reports will be set to TRUE. The `basedirectory` argument is set to the URL files will be hosted at. All files will be written to the directory set by the `writedirectory` and should be copied to the URL directory set by `basedirectory` argument for the report to be functional.

In this example, a github address, set by `basedirectory`, is used to host the reduced files and the `writedirectory` is set to the current working directory. The full.xml.paths, full.file.paths are set to TRUE so all files will be linked to to `basedirectory` by absolute paths.

```
urlForFiles  <- "https://github.com/ThomasCarroll/tracktables-Data/raw/master/"

# This will link to data and XMLs placed in github earlier.
# In practice a dedicated FTP would be required for larger files.
HTMLreport <- maketracktable(fileSheet=FileSheet,SampleSheet=SampleSheet,
                             filename="IGVEx4.html",genome="mm9",
                             basedirectory=urlForFiles,
                             full.xml.paths=T,
                             full.file.paths=T,
                             writedirectory=file.path(getwd(),"IGVDirectory")
                            )
```

All files will be created in the `writedirectory` and with the exception of the tracktables experiment report ("IGVExample.html"), should be copied to the `basedirectory` for the report to be functional. In this example, the URL already contains the files required.

3.3 Creating a Tracktable HTML report using absolute and relative paths

Often, the sample files of interest are located on a stable or public URL but the user does not have write access to this URL. In order to build tracktables reports, the user can create only the associated sample metadata and XML files locally and relative to the main report but keep absolute links to sample files such as bigWigs, bams and bed files hosted externally.

The use of such hybrid reports allow for a tracktables report of publically hosted data with more lightweight and highly customisable sample metadata and XML files stored locally.

In this report, the `full.xml.paths` is set to false and all XML files and the sample information file will be written to the directory specified by `writedirectory`.

```
# Example URL
urlForFiles  <- "https://github.com/ThomasCarroll/tracktables-Data/raw/master/"

HTMLreport <- maketracktable(fileSheet=FileSheet,SampleSheet=SampleSheet,
                             filename="IGVEx5.html",genome="mm9",
                             basedirectory=urlForFiles,
                             full.xml.paths=F,
                             full.file.paths=T,
                             writedirectory=file.path(getwd(),"IGVDirectory")
                            )
```

The IGVDirectory now contains a Tracktables Experiment Report and all required XML, html and sample information files with absolute path links to files hosted externally (or locally).

3.4 Creating a Tracktable HTML report from files across multiple hosts (multiple FTP/HTTP addresses)

The `use.path.asis` argument causes all links to sample files to be based on their original path in the fileSheet and so overrides the path set by `basedirectory` argument. This is useful when fileSheet listed files hosted on multiple systems/URLs as can be the case when pooling many publically hosted files.

In this example a locally directory hosting all files will be used. This report then will be able to transferred across directories which can access the sample file locations through its full paths.

To use the full paths to sample files as specified in the filesheet, the `use.path.asis` argument is set to true.

```
urlForFiles  <- "https://github.com/ThomasCarroll/tracktables-Data/raw/master/"

HTMLreport <- maketracktable(fileSheet=FileSheet,SampleSheet=SampleSheet,
                             filename="IGVEx6.html",genome="mm9",
                             basedirectory=urlForFiles,
                             full.xml.paths=F,
                             full.file.paths=T,
                             writedirectory=file.path(getwd(),"IGVDirectory"),
                             use.path.asis=T
                            )
```

In this report, all sample file links are to the files' original locations. All IGV related files have been written to "IGVDirectory". The directory and its report are now self contained with dependencies on the files original locations.

4 Session Information

Here is the output of `sessionInfo` on the system on which this document was compiled:

```
toLatex(sessionInfo())
```

- R version 3.2.0 (2015-04-16), `x86_64-unknown-linux-gnu`
- Locale: `LC_CTYPE=en_US.UTF-8`, `LC_NUMERIC=C`, `LC_TIME=en_US.UTF-8`, `LC_COLLATE=C`, `LC_MONETARY=en_US.UTF-8`, `LC_MESSAGES=en_US.UTF-8`, `LC_PAPER=en_US.UTF-8`, `LC_NAME=C`, `LC_ADDRESS=C`, `LC_TELEPHONE=C`, `LC_MEASUREMENT=en_US.UTF-8`, `LC_IDENTIFICATION=C`
- Base packages: base, datasets, grDevices, graphics, methods, stats, utils
- Other packages: knitr 1.10.5, tracktables 1.2.3
- Loaded via a namespace (and not attached): BiocGenerics 0.14.0, BiocStyle 1.6.0, Biostrings 2.36.1, GenomeInfoDb 1.4.0, GenomicRanges 1.20.3, IRanges 2.2.1, RColorBrewer 1.1-2, Rsamtools 1.20.1, S4Vectors 0.6.0, XML 3.98-1.1, XVector 0.8.0,

bitops 1.0-6, evaluate 0.7, formatR 1.2, highr 0.5, magrittr 1.5, parallel 3.2.0, reportr 1.1.2, stats4 3.2.0, stringi 0.4-1, stringr 1.0.0, tools 3.2.0, tractor.base 2.5.0, zlibbioc 1.14.0