

# Visualising ChIP-seq data – tracktables

Thomas Carroll<sup>1\*</sup>

<sup>1</sup> Bioinformatics Facility, MRC Clinical Sciences Centre;

\*`thomas.carroll (at) imperial.ac.uk`

September 20, 2014

## **Abstract**

tracktables provides a set of tools to organise visualise your ChIP-seq data.

## Contents

---

<b>1</b>	<b>Standard workflow</b>	<b>2</b>
1.1	Quick start . . . . .	2

## 1 Standard workflow

---

### 1.1 Quick start

Tracktables has two main sets of functions to assist in rapid visualisation of genomics data.

The first set of functions deal with creating IGV sessions XMLs, sample information files and HTML tables allowing for dynamic updating of IGV.

In this quick start we will look at building a simple HTML page of a ChIP-seq IGV session.

```
library(tracktables)
SampleSheet <- cbind(
  c("Pu1", "Myc", "Ik_prePro", "Ik_pro"),
  c("ProB", "ProB", "PreProB", "ProB"),
  c("Pu1", "Myc", "Ik", "Ik")
)
colnames(SampleSheet) <- c("SampleName", "Tissue", "Factor")
fileSheet <- cbind(
  c("Pu1", "Myc", "Ik_prePro", "Ik_pro"),
  c(NA, NA, NA, NA),
  c("/Users/tcarroll/Downloads/randomTracks-2/Pu1DupMarkedNormalised.bw",
    "/Users/tcarroll/Downloads/randomTracks-2/MycDupMarkedNormalised.bw",
    "/Users/tcarroll/Downloads/randomTracks-2/Ikaros_2_preproBDupMarkedNormalised.bw",
    "/Users/tcarroll/Downloads/randomTracks-2/Ikaros_1_proBDupMarkedNormalised.bw"),
  c("/Users/tcarroll/Downloads/randomTracks/Pu1_WithInput_Input_2_proB_peaks.bed",
    "/Users/tcarroll/Downloads/randomTracks/Myc_WithInput_Input_Ch12_peaks.bed",
    "/Users/tcarroll/Downloads/randomTracks/Ikaros_2_preproB_WithInput_Input_2_proB_peaks.bed",
    "/Users/tcarroll/Downloads/randomTracks/Ikaros_1_proB_WithInput_Input_2_proB_peaks.bed")
)
colnames(fileSheet) <- c("SampleName", "bam", "bigwig", "interval")
```

```
library(tracktables)
SampleSheet <- cbind(
  c("Pu1", "Myc", "Ik_prePro", "Ik_pro"),
  c("ProB", "ProB", "PreProB", "ProB"),
  c("Pu1", "Myc", "Ik", "Ik")
)
colnames(SampleSheet) <- c("SampleName", "Tissue", "Factor")
```

```
fileSheet <- cbind(
  c("Pu1", "Myc", "Ik_prePro", "Ik_pro"),
  c(NA, NA, NA, NA),
  c("/Users/tcarroll/Downloads/randomTracks-2/Pu1DupMarkedNormalised.bw",
    "/Users/tcarroll/Downloads/randomTracks-2/MycDupMarkedNormalised.bw",
    "/Users/tcarroll/Downloads/randomTracks-2/Ikaros_2_preproBDupMarkedNormalised.bw",
    "/Users/tcarroll/Downloads/randomTracks-2/Ikaros_1_proBDupMarkedNormalised.bw"),
  c("/Users/tcarroll/Downloads/randomTracks/Pu1_WithInput_Input_2_proB_peaks.bed",
    "/Users/tcarroll/Downloads/randomTracks/Myc_WithInput_Input_Ch12_peaks.bed",
    "/Users/tcarroll/Downloads/randomTracks/Ikaros_2_preproB_WithInput_Input_2_proB_peaks.bed",
    "/Users/tcarroll/Downloads/randomTracks/Ikaros_1_proB_WithInput_Input_2_proB_peaks.bed")
)
colnames(fileSheet) <- c("SampleName", "bam", "bigwig", "interval")
```

The first set of functions are for summarising and visualising coverage of genomic regions across samples and gene/genomic-interval sets. The `regionPlot` function calculates normalised coverage across or over user specified regions to produce *ChIPprofile* object.

```
print("Coming Soon!")
```