Assessing ChIP-seq sample quality with *ChIPQC*

Thomas Carroll

Edited: March 24, 2014; Compiled: July 27, 2014

Contents

1 Introduction

This practical will cover how to assess ChIP-seq data quality using the ChIPQC package with real world datasets.

Due to limitation on time the data has been processed and ChIPQCexperiments objects prepared for this practical but we will still cover how to set up data for ChIPQC and how to run the main procssing commands.

2 Single sample assessment using `ChIPQCsample()`

2.1 A simple example of using `ChIPQCsample()`

ChIPQC package allows for the rapid generation of ChIP-seq quality metrics from BAM format aligned data. The main function `ChIPQCsample()` can be run with simply a BAM file and will return a `ChIPQCsample` object. Here we can run ChIPQCsample on a single Bam file of an EBF1 ChIP-seq experiment for chromosome 11. The data used can be found in "/data/ChIPQC/" so feel free to give the function a go.

```
> library(ChIPQC)
> #bamFile <- "/data/ChIPQC/Chr11_Ebf1DupMarked.bam"
> #exampleExp = ChIPQCsample(bamFile,peaks=NULL,annotation=NULL,chromosomes="chr11")
> load("/Users/tcarroll/Downloads/BiocTalk (1)/ebf1_ChIPQCsample.RData")
> QCMetrics(exampleExp)
```

```
     Reads       Map%     Filt%      Dup%     ReadL     FragL     RelCC       SSD
1369600.00     100.00     10.40      6.56     36.00    177.00      3.57      1.00
      RiP%
        NA
```

This is the simplest way to run ChIPQCsample but to take full advantage of ChIPQC features we can add additional information on blacklisted regions, genome annotation and any peaks we have called for this ChIP.

Annotation is provided for human ("hg19", "hg18") mouse ("mm10","mm9"), rat ("rn4"), C Elegans ("ce6") and D Melanogaster ("dm3") by way of the Bioconductor TranscriptDb annotations packages. Additional or custom anotation can also be provided to the ChIPQCsample function and we go into this in detail later on.

Blacklist regions are included for hg19 (link to data) and here can be provided as a GRanges object or complete file path to blacklists in bed format. Blacklists for a range of species is available from Anshul Kudaje's google site.

Peaks can also be provided to the ChIPQCsample function as a GRanges object or as complete file path to peaks file in bed format.

```
> bamFile <- "/BiocTalk//Ebf1DupMarked.bam"
> peaksFile <- "/Ebf1_WithInput_Input_2_proB_peaks.bed"
> BlackListFile <- "/BiocTalk//mm9-blacklist.bed"
> exampleExp = ChIPQCsample(bamFile,peaks=peaksFile,annotation="mm9",blacklist=BlackListFile,chromosomes="c
> QCmetrics(exampleExp)
```

```
     Reads       Map%     Filt%      Dup%     ReadL     FragL     RelCC       SSD
1369600.00     100.00     10.40      6.56     36.00    177.00      3.57      1.00
      RiP%      RiBL%
      5.75       2.23
```

Now result of `QCmetrics` contains full metrics for ChIP-seq and the additional information on RiP% and RiBL%. Lets just remind ourselves what all these mean.

Reads Total reads in bam file

Map% Percentage of total reads mappng within file.

Filt% Percentage of mapped reads passing MapQ filter.

Dup% Percentage of mapped reads marked as duplicates.

ReadL Mean read length (as integer).

FragL Predicted fragmentlength by cross-coverage method.

RelCC The relative cross-coverage score

SSD Standardised standard deviation

RiP% Reads mapped to peaks

RiBL% Reads mapped to blacklists

Now we have our full ChIPQCsample object we can start to review the metrics generated and produce the useful plots.

2.2 Cross-coverage and the FragmentLength/Relative cross-coverage scores (FragCC/RelCC)

For transcription factors and narrow epigenetic marks, an accumulation of watson and crick reads around the binding site/mark may often be seen. The degree to which your ChIP-seq signal is arranged into the watson and crick read clusters around such sites has been previously exploited as a metric of ChIP efficiency.

In the ChIPQC we assess the reduction in total genome covered which occurs from shifting the watson reads along the genome (5' to 3' of chromsome). This is performed by measuring measure total coverage after each every successive shift of 1bp. As the watson reads overlap the crick reads around peaks the total genome covered will be reduced. The total coverage after each successive shift is then converted to cross-coverage scores after each shift.

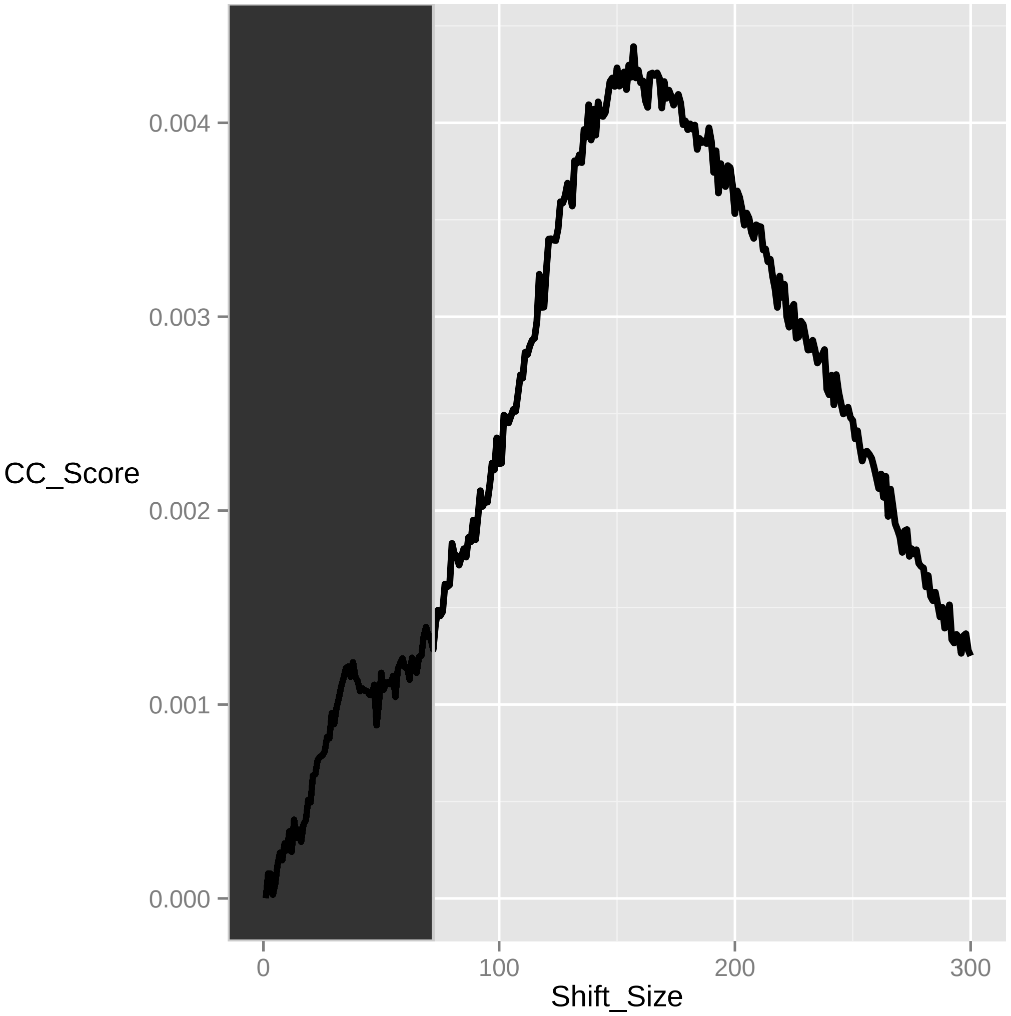
$$CrossCoverageScore_n = (TotalGenomeCoverage_0 - TotalGenomeCoverage_n)/TotalGenomeCoverage_0 \quad (1)$$

Where *n* is the bp shift of watson reads and *0* is after no shift of watson reads.

The cross-coverage scores after successive shifts can then be visualised and reviewed to identify the expected reduction in coverage around the fragment length as well as any evidence of artefacts by a reduction in coverage at the read length.

The plotCC function will calculate cross-coverage scores and plot those after successive shifts. Also shaded in grey is the area to be excluded when identifying the fragment length peak.

```
> plotCC(exampleExp)
```



We can see that this Ebf1 ChIP has a peak in cross-coverage scores at around 160bp, corresponding to the fragment size, high above that observed at the 0 shift. This indicates that we have successfully enriched for signal around binding sites. Further to this we can also see an artefact peak at the read length and that this peak is much lower than that observed at fragment length, again indicating an enrichment for ChIP-signal over background.

Further to the visual inspection of the cross-coverage scores, we can extract the RelCC and FragCC scores for this sample using `RelativeCrossCoverage` and `FragmentLengthCrossCoverage` functions respectively.

These metrics can be considered to relate to efficiency of ChIP (FragCC) and efficiency of ChIP compared to artefact signal (RelCC) and are calulated as below.

$$FragCC = CrossCoverageScore_{max} \tag{2}$$

$$RelCC = CrossCoverageScore_{max}/CrossCoverageScore_{readlength} \tag{3}$$

Where *max* is shift with maximum cross-coverage score (area 0 to 1.5*readlength) and *readlength* is the cross-coverage score at the readlength.

The area around the read length is excluded from selection of shift with maximum cross-coverage scores (shaded in grey in cross-coverage plots). This is due to the presense of the artefact peak which, in less enriched samples, may have greater cross-coverage score than observed at the fragment length. To avoid obsuring the selection of the fragment length peak for fragment length prediction and ascertation of FragCC score and RelCC score this region is removed prior to calculation of max CrossCoverageScore.

```
> FragmentLengthCrossCoverage(exampleExp)
```

```
[1] 0.004271447
```

```
> RelativeCrossCoverage(exampleExp)
```

```
[1] 3.571335
```

```
>
```

In this example we find, as expected from cross coverage scores graph, that the FragCC is high and RelCC score is above 1 indicating a successfull ChIP.

2.3 Distribution of Signal: Within peaks, blacklists and known annotation

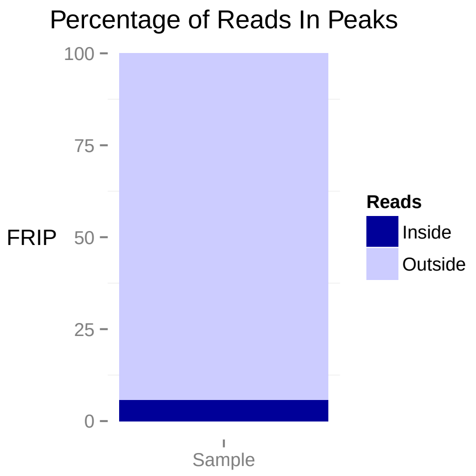
Another useful set of characteristics of your ChIP-seq data is where in the genome the signal is distributed. This can be done by looking for the proportion of signal in peaks, blacklists or even in known annotation.

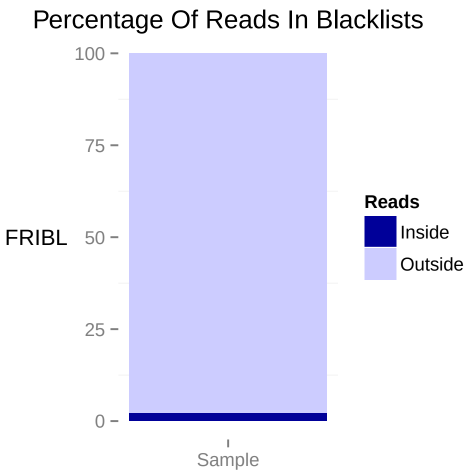
To get the fraction of reads landing in peaks and blacklists we can use the `rip` and `ribl` function as well as the `plotFrip` and `plotFribl` functions for visualisation.

```
> frip(exampleExp)
```

```
[1] 0.05747079
```

```
> p1 <- plotFrip(exampleExp)
> p2 <- plotFribl(exampleExp)
```





The frip and fribl plot, show that we have a proportion of signal in peaks is greater than 5% indicating an acceptable quality. Reassuringly, we have a higher signal in peaks than excluded regions (2%) and so an enrichment over artefact signal.

When provided with annotation in the form of genomic regions, ChIPQC measures the enrichment of signal within them. The regi function provides a simple enrichment statistic (described below) which illustrates distribution of signal within genomic interval annotation over that expected given their size.

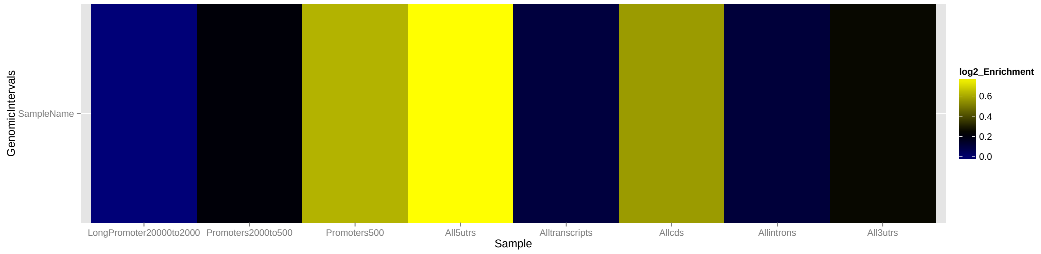
$$regi = ProportionOfReadsInInterval/ProportionOfGenomeInInterval \tag{4}$$

To review regi statistics we can use the regi function or plot enrichment using plotRegi.

```
> regi(exampleExp)
```

```
LongPromoter20000to2000      Promoters2000to500            Promoters500
            -0.02709166              0.21537437              0.62420705
               All5utrs          Alltranscripts                 Allcds
             0.78965251              0.09784935              0.57156632
             Allintrons                All3utrs
             0.10562037              0.25087385
```

```
> plotRegi(exampleExp)
```



The regi scores and heatmap show us an enrichment for regions around the TSS including 5'UTRs,CDS (potentially first exon) and 500upstream regions. This suggests that Ebf1 is a promoter associated transcription factor as expected by known patterns of binding.

2.4 Distribution of Signal: Distribution of coverage depth across the genome

The final metrics to review are those of the distribution of global pile-up across the genome. We can access this in two ways within ChIPQC, first by visualising the histogram of coverage depths and secondly by applying the SSD metrics before and after removal of blacklisted regions.

First we draw the coverage histogram using plotCoverageHistogram. Note the cut-off at 100bp for visualisation purposes. If you want to replot the whole histogram you can extract the data using coveragehistogram function.

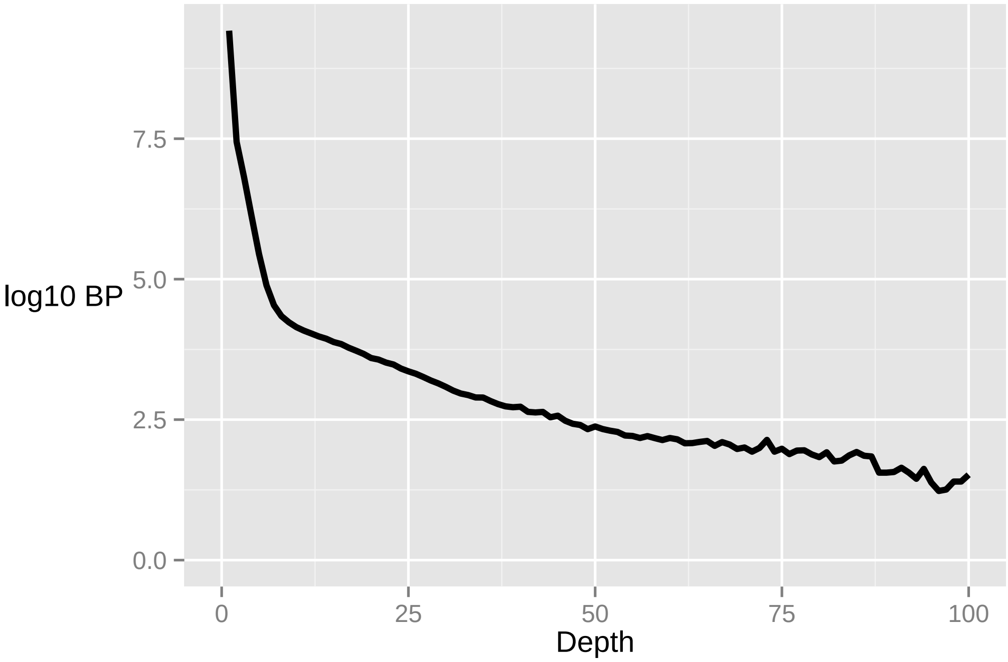
```
> coveragehistogram(exampleExp)[1:10]
```

```
         0          1          2          3          4          5          6          7
2644410366   27778827    6486312    1328557     280542      77788      34320      21983
         8          9
     17086      14003
```

```
> sum(coveragehistogram(exampleExp)[-c(1:19)])
```

```
[1] 43192
```

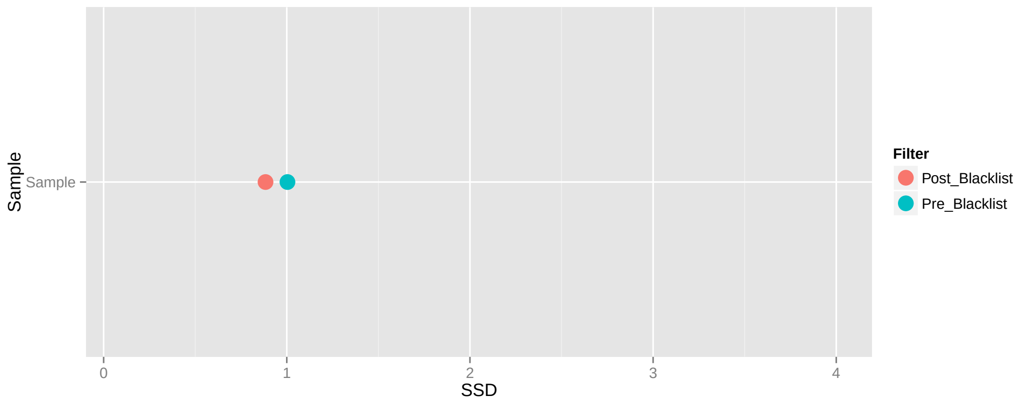
```
> plotCoverageHist(exampleExp)
```



The coverage histogram shows that there is a significant stretch of high signal pile-up (more than 40,000 bps at greater than 20 depth). This may indicate significant signal associated to binding events but could also be from signal seen within artefact regions.

To assess the contribution of artefact signal to global distrubtion of signal pileup we measure SSD before and after exclusion of signal from known blacklisted regions.

```
> plotSSD(exampleExp)
```



Here we find that the SSD signal is not greatly affected by blacklisted regions and so, taken together with the coverage histogram, indicates the Ebf1 ChIP to have a clear ChIP-signal above background.

2.5 Conclusion

From the combination of metrics used here we can establish that this Ebf1 ChIP showed an enrichment for structured ChIP-signal, had a signal pile-up above that seen within artefact regions and that this ChIP is positively associated with TSS regions.

3 Assessing a ChIP-seq experiment using `ChIPQC()`

When assessing ChIP-seq quality it is most useful to consider your sample quality alongside other ChIP and input samples. Taken together, a full experiment of ChIP and input samples allows for the identifaction of expected enrichment of sample metrics above input but also the variation of sample quality between biological replicates and the identification of bias within input/control samples.

The ChIPQC function function wraps the functionality of ChIPQCsample to allow for the assessment of within experiment ChIP sample/input quality alongside user supplied experimental metadata.

3.1 An example ChIP-seq experiment using `ChIPQC()`

The ChIPQC function accepts a samplesheet of metadata and file locations alongside the same set of arguments as ChIPQCsample.

An example of the layout for a sample sheet can be seen in the SampleSheet object below and in the ChIPQC and Diffbind vignettes. ChIPQC can also accept the DBA object from Diffbind package as a starting point for quality control. The result of a call to ChIPQC is the ChIPQCexperiment object which contains the list of ChIPQCsample objects.

```
> load("/Users/tcarroll/Downloads/BiocTalk (1)/SampleSheet.RData")
> SampleSheet[1:3,]
> #resExperiment = ChIPQC(SampleSheet,peaks=peaksFile,annotation="mm9",blacklist=BlackListFile)
> load("/Users/tcarroll/Downloads/BiocTalk (1)/BCell_Examples.RData")
> resExperiment
```

```
   SampleID Tissue  Factor Condition Treatment Peak caller    ControlID Replicate
1     DNAse   Ch12   DNAse                                macs   Input_Ch12         1
2      Ebf1   ProB    Ebf1                                macs Input_2_proB         1
3 H3K4me3_1   ProB H3K4me3                                macs Input_2_proB         1
```

```
                               bamRead                              bamControl
1     //AlignedData/DNAseDupMarked.bam   //AlignedData/Input_Ch12DupMarked.bam
2      //AlignedData/Ebf1DupMarked.bam //AlignedData/Input_2_proBDupMarked.bam
3 //AlignedData/H3K4me3_1DupMarked.bam //AlignedData/Input_2_proBDupMarked.bam
                                 Peaks
1     //AlignedData/DNAseDupMarked.bed
2      //AlignedData/Ebf1DupMarked.bed
3 //AlignedData/H3K4me3_1DupMarked.bed
```

```
Samples: 19 : DNAse Ebf1 ... RNAPol2 RNAPol2ser2
                  Tissue       Factor Replicate Peaks
DNAse               Ch12        DNAse         1 72437
Ebf1                ProB         Ebf1         1  9841
H3K4me3_IkNeg       ProB      H3K4me3         1 18462
H3K4me3_IkPos       ProB      H3K4me3         2 20052
H3K9ac_IkNeg        ProB       H3K9ac         1 27486
H3K9ac_IkPos        ProB       H3K9ac         2 25902
Ikaros_1_DPT         DPT       Ikaros         1 22253
Ikaros_1_preproB preProB       Ikaros         1 16240
Ikaros_1_proB       ProB       Ikaros         1 15377
Ikaros_2_preproB preProB       Ikaros         2 16038
Input_2_proB        ProB        Input         1     0
Input_Ch12          Ch12        Input         1     0
Irf                 ProB          Irf         1 39628
Mi2b_IKneg          ProB         Mi2b         1     0
Mi2b_IKpos          ProB         Mi2b         2     0
Myc                 Ch12          Myc         1 44982
Pu1                 ProB          Pu1         1 48472
RNAPol2             Ch12      RNAPol2         1 59266
RNAPol2ser2         Ch12 RNAPol2_Ser2         1 30787
                     Reads Map% Filt%  Dup% ReadL FragL  RelCC  SSD  RiP% RiBL%
DNAse            109762060  100  15.5 27.60    36    73 0.9280 3.22 45.40  7.94
Ebf1              29098218  100  24.3  6.60    36   177 2.6400 2.67  3.07 11.00
H3K4me3_IkNeg     76096001  100  15.1 27.00    36   228 3.9300 2.98 45.90  7.94
H3K4me3_IkPos    128801543  100  14.0 50.00    36   243 3.8200 3.61 50.70  7.40
H3K9ac_IkNeg      64585441  100  14.8 13.00    36   215 3.9300 2.28 36.30  6.73
H3K9ac_IkPos      69136473  100  13.9 14.70    36   213 4.1700 2.01 37.90  5.84
Ikaros_1_DPT      48253184  100  23.0 15.30    36   173 1.8000 3.19  5.31 10.50
Ikaros_1_preproB  35903381  100  24.7 15.30    36   175 0.2780 3.43  3.50 12.50
Ikaros_1_proB     24827255  100  23.2 13.10    36   165 0.9210 2.28  3.49 10.50
Ikaros_2_preproB  35493401  100  24.7 15.20    36   181 0.3110 3.42  3.43 12.60
Input_2_proB      23454376  100  22.1  9.73    36   120 1.6500 1.93    NA  9.02
Input_Ch12        19106045  100  25.5 21.50    36   157 0.0556 2.06    NA  9.81
Irf               27877382  100  21.4 20.80    36   153 2.3200 2.24 10.50  9.91
Mi2b_IKneg        38541596  100  23.5 29.30    36   151 0.4990 3.09    NA 11.40
Mi2b_IKpos        34823163  100  23.8  6.08    36   152 0.6380 3.02    NA 11.60
Myc               39300897  100  19.4 11.60    36   150 1.5700 2.29 16.80  8.74
Pu1               34095496  100  20.8 20.30    36   204 3.3300 2.52 22.20  8.89
RNAPol2           33484861  100  12.3 17.90    36   152 1.9200 1.11 45.70  4.98
RNAPol2ser2       48222450  100  15.0  6.63    36   125 0.8050 1.61 30.30  5.86
```

The ChIPQCexperiment displays a table of metrics the same as seen for ChIPQCsample and all accessors and plotting functions used for ChIPQCsample objects can be used with the ChIPQCexperiment object.

In addition to standard plotting routine, ChIPQCexperiment plots can be grouped by the metadata provided using the argument facetBy and for plotCoverageHistogram and plotCC methods the colours and line types controled by colourBy and lineBy respectively.

To group/colour/line type by metadata, a character vector of the metadata column title/s to use may be provided.

3.2 Examining Cross-coverage and FragCC/RelCC scores across an experiment

As with the ChIPQCsample object, the cross coverage scores for a group of samples can be plotted using plotCC(). By default samples will be groups by their Tissue and Factor combinations and coloured by their replicate number. Here we group by Factor, colour by Tissue and set the line type by the replicate number.

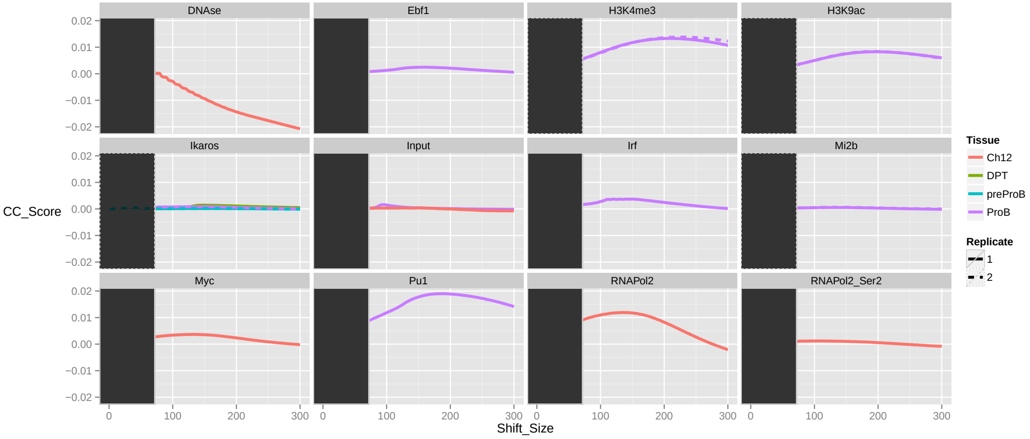
```
> FragmentLengthCrossCoverage(resExperiment)[1:3]
```

```
       DNAse          Ebf1 H3K4me3_IkNeg
0.0001493727  0.0024606567  0.0133150115
```

```
> RelativeCrossCoverage(resExperiment)[1:3]
```

```
       DNAse          Ebf1 H3K4me3_IkNeg
   0.9277036     2.6368994     3.9339780
```

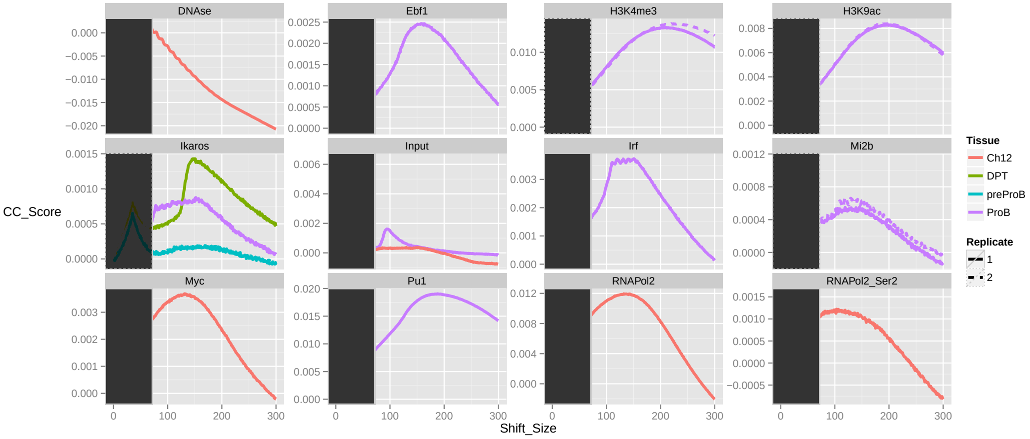
```
> plotCC(resExperiment,colourBy="Tissue",facetBy="Factor",lineBy="Replicate")
```



From this, it is immediately apparent that some samples not only have much higher scores, and hence efficiency, than others but that there fragment lengths appear to be very different from each other. The DNAse sample for example has a fragment length almost half of Pu1 sample.

Now we have established the difference in total efficiency, we can look at the overall shape of cross-coverage scores and the relationship between signal peak and artefact peak in the cross-coverage scores. To help better visualise, we will apply a further facet wrap to the ggplot2 object returned by plotCC inorder to compare within Factors.

```
> ccplot <- plotCC(resExperiment,colourBy="Tissue",facetBy="Factor",lineBy="Replicate")
> ccplot +facet_wrap(~Factor,scales="free_y")
```



The free scaled cross coverage score plots now reveal more about the distibution of signal within the samples. The Ebf1, Ikaros, Myc, Ifr and RNAPol2 all show tight peaks within their cross coverage score profiles illustrating their sharp bindng profiles as expected for a transcription factor (and RNApol2 around TSS/Enhancers.) The histone marks and RNAPol2-Ser2 however show longer more diffused peaks reflecting the wider bredth of signal seen for these epigenetic marks.

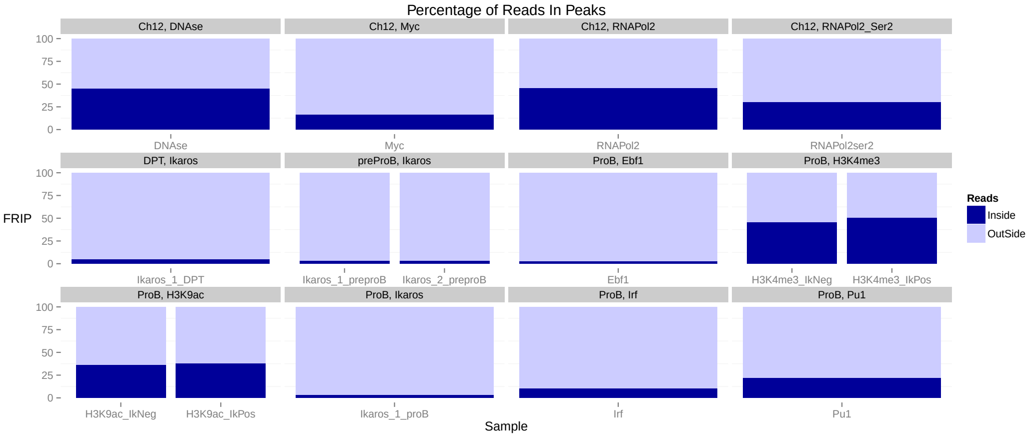
The signal of the Ikaros ChIP between cell lines can also seen to be highly varable with DP thymocytes containing highest RelCC scores, ProB lower and preProB the lowest. This reflects the increased concentrations of Ikaros along haemopoetic differentiation with DP thymocytes having the highest Ikaros levels.

Finally, an enrichment for fragment length signal can be seen in the ProB input. The sharpness of this enrichment suggests a highly duplicated peak like signal within this track which would be cause for further investigation. This may be from the "sono-seq" effect commonly observed in input, where gel-selection of fragment lengths for input causes a small peak in cross coverage scores close to selected fragment length.

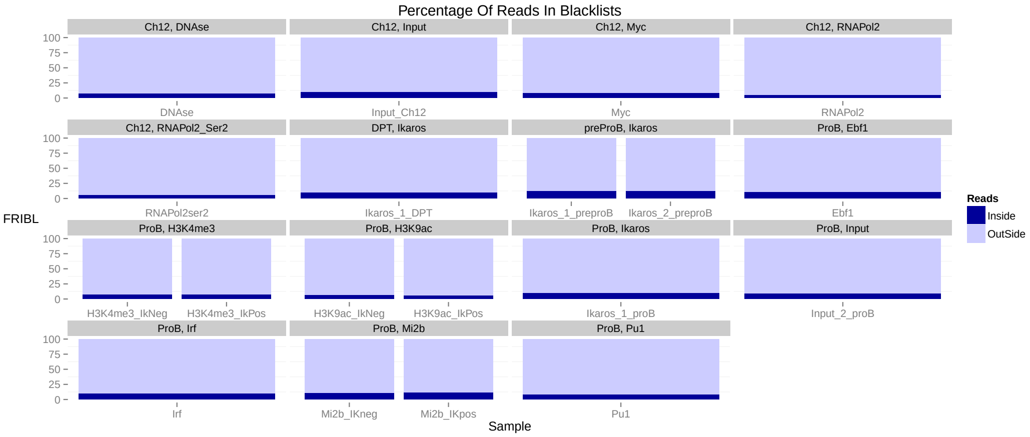
3.3 Distribution of Signal across a ChIPQC experiment

As with the the ChIPQCsample object, the fraction of signal in peaks, blacklists and annotated genomic intervals can provide an understanding of the ChIPs' efficiency and pattern of enrichment.

```
> plotFrip(resExperiment)
```



```
> plotFribl(resExperiment)
```



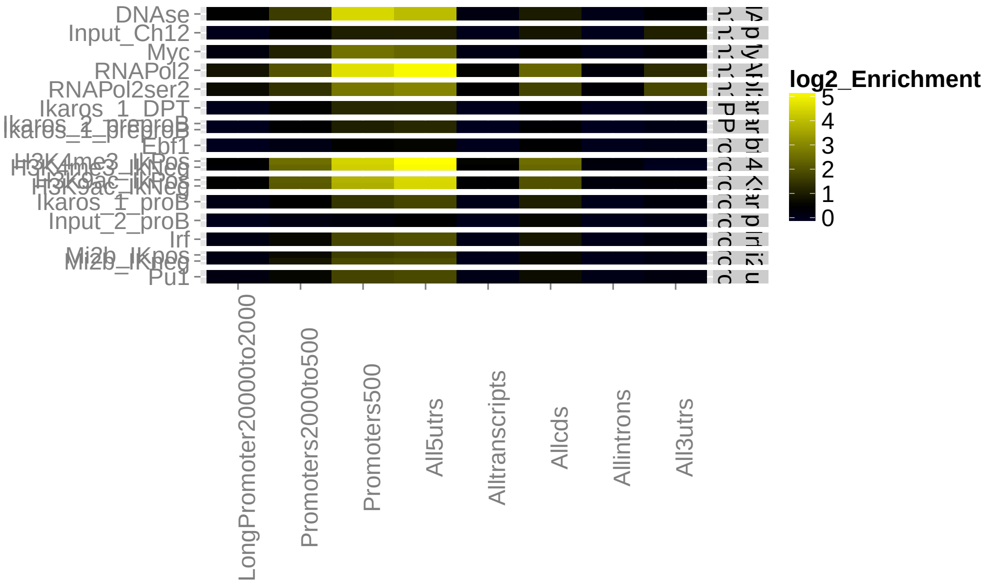
The output from plotFrip immediately identifies the Histone, Polymerase and DNAse ChIP has having the highest enrichment (25% to 50%)for reads in peaks as expected for such epigenetic marks. Also apparent are the significant enrichment seen within Pu1, Myc and Irf whereas the Ikaros ChIPs all show considerably enrichment for signal within peaks.

The Fribl plot here shows that all samples have equivalent levels of signal so no outlier or need to investigate signal within known blacklisted regions.

Samples such as RNA polymerase 2, DNAse and Histone marks will have an expected enrichment for genomic locations. Here RNApol2 should be expected to have a stonger enrichment at the TSS than RNApol2ser2 where as RNApol2ser2 should show enrichment within the 3'UTRs regions.

```
> plotRegi(resExperiment)
> regi(resExperiment)["All3utrs",]
```

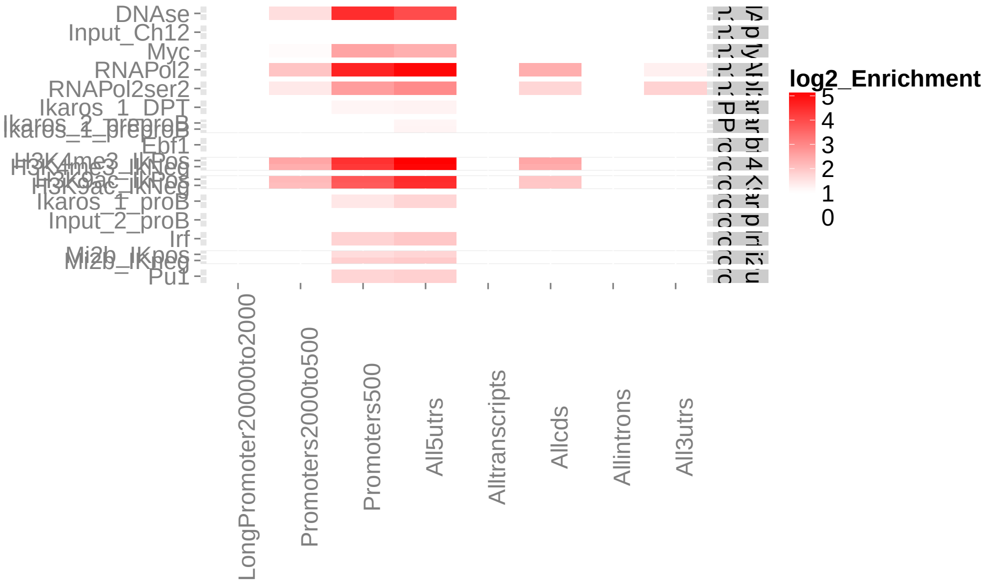
```
          DNAse              Ebf1     H3K4me3_IkNeg     H3K4me3_IkPos      H3K9ac_IkNeg
     0.42145087        0.08548231       -0.08226842        0.01884930        0.39311725
   H3K9ac_IkPos      Ikaros_1_DPT  Ikaros_1_preproB     Ikaros_1_proB  Ikaros_2_preproB
     0.40439726        0.10945327        0.10804849        0.29671723        0.10832958
   Input_2_proB        Input_Ch12               Irf        Mi2b_IKneg        Mi2b_IKpos
     0.15271636        1.08959831        0.28397139        0.16220769        0.16271435
            Myc               Pu1           RNAPol2       RNAPol2ser2
     0.32845215        0.23293713        1.31671974        1.80520683
```



From the Regi plot it can seen that all histone marks, DNA and RNA pol2 show the expected enrichment across gene regions. Combined with the output from regi function, the RNApol2Ser2 has the greatest enrichment at 3'UTRs and so the expected pattern of enrichment.

To better visualiase this enrichment we can adjust the scale to the enrichment seen in Ch12 input for 3'UTRs.

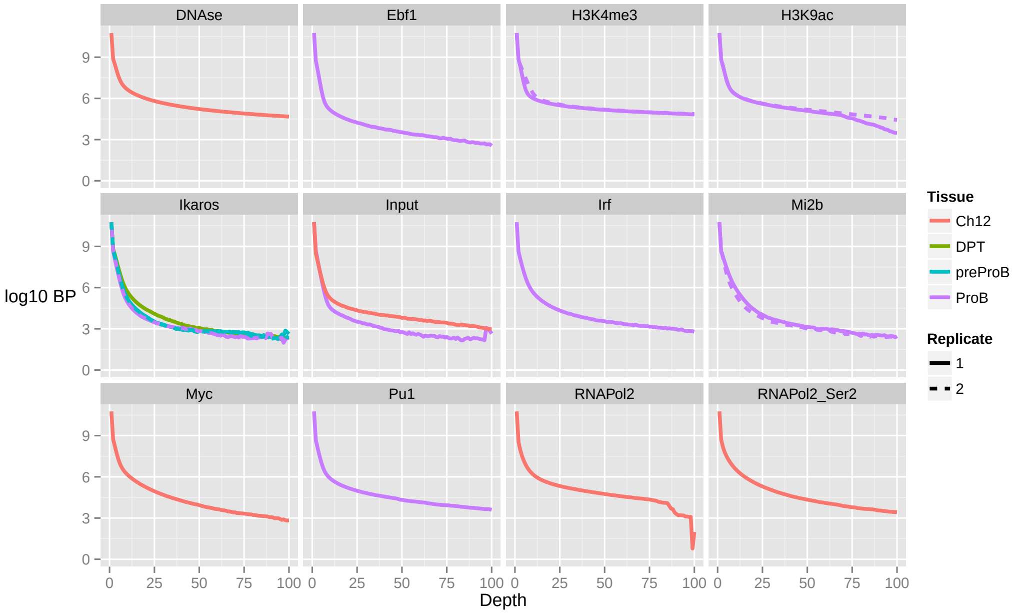
```
> plotRegi(resExperiment)+scale_fill_gradient2(low="white",high="red",mid="white",midpoint=regi(resExperime
```



Both inputs showed a small but comparatively low enrichment for reads in genic regions. Such enrichment around gene regions can typically be seen for input samples due to the increased accessibilty of chromatin to fragmentation around TSSs.

As with the ChiPQCsample object we can plot the coverage histogram and SSD before and after blacklisting by using the plotCoverageHist and plotSSD functions.

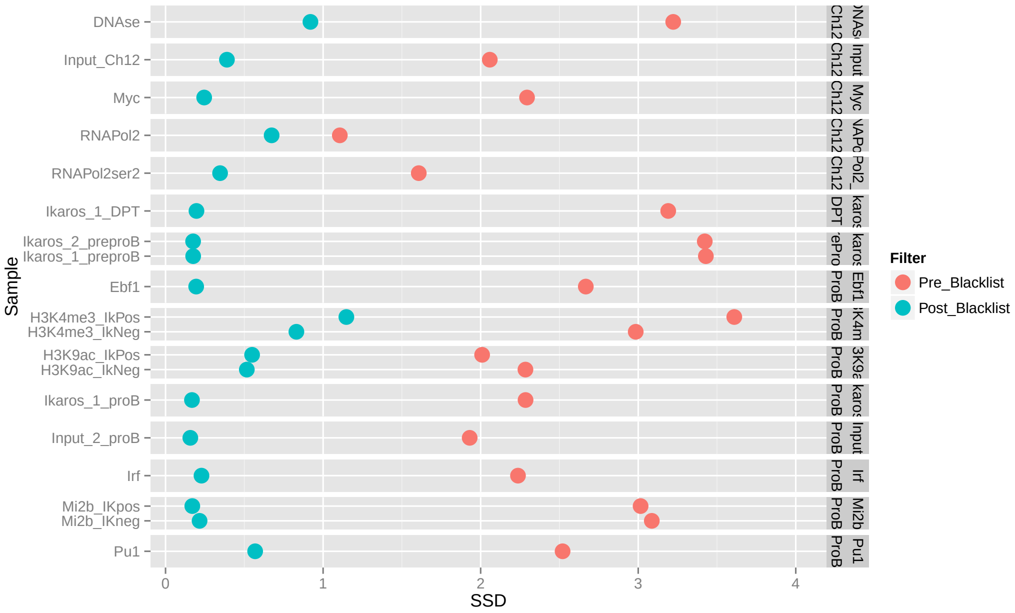
```
> plotCoverageHist(resExperiment,facetBy="Factor",colourBy="Tissue",lineBy="Replicate")
```



The coverage histogram shows the expected greater spans of high signal in Histone, RNA pol2 and DNAse ChIPs as well as for transcription factors with both high Rip and RelCC scores. Although most ProB transcription factors have greater spans of high depth than their input, enrichment for these transcription factors can be seen to be much smaller than observed for histone, polymerase and DNAse ChIPs.

The Ikaros ChIPs universally had low enrichment and as seen with other metrics DP thymocytes had the greatest signal.

The two inputs samples show very different patterns of signal depth. The Ch12 input show considerable span of high signal where as the ProB shows a spike in signal at above 98 reads high, in keeping with observed high duplication rate for this sample and fragment length peak in cross-coverage scores.



The effect of blacklisting can be seen to be dramatic on SSD scores.

As with the coverage histogram plots, the histone, polymerase, DNAse marks as well as high scoring TFs have greatest SSDs after blacklisting.

The SSD for Ikaros after blacklisting can be seen to be just above background with again DP thymocytes having the greatest score.

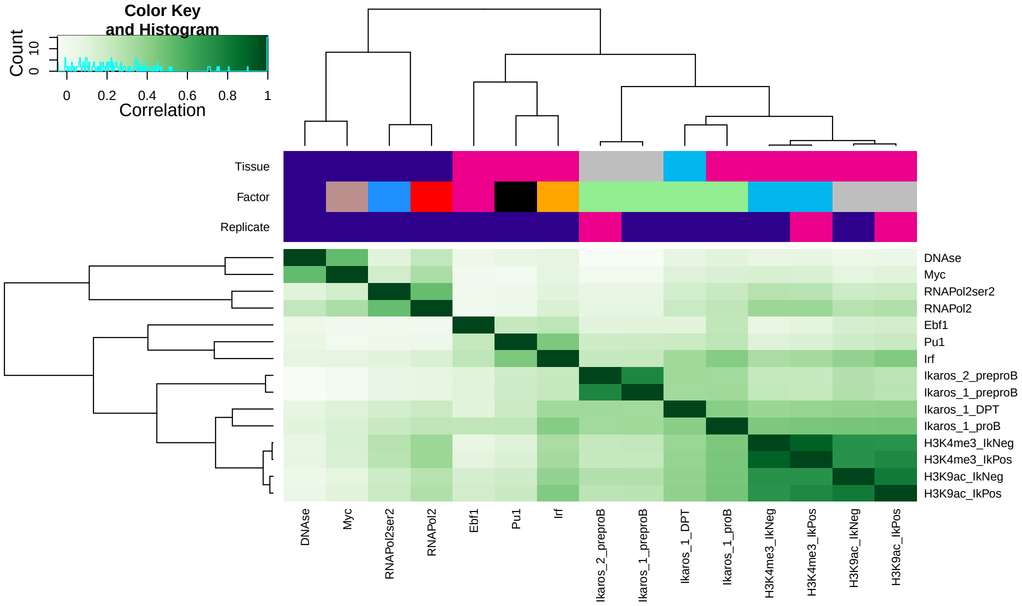
For the inputs, the Ch12 input can be seen to have its SSD score reduced to a background level of around 0.14 indicating the succesfull removal of artefact signal. The ProB input however can be seen to not drop to the background level after blacklisting suggesting remaining regions of artefact signal. The fragment length peaks in cross-coverage scores, observed high signal spike in coverage histogram and failure to reduce SSD after blackisting suggests the presence of highly duplicated peak like artefacts within the ProB input and flags this sample for further Blacklisting.

3.4 Assessing sample similarity with Diffbind

A final set of metrics useful for ChIP-quality relate to the correlation between binding events across samples within a ChIP-experiment.

The Diffbind package allows for the clustering of samples based on the co-occurence of peaks within samples. When analysing an experiment ChIPQC will perform a sample clustering by default as well as construct the Diffbind DBA object. To produce the a sample heatmap, the plotCorHeatmap function can be used.

```
> plotCorHeatmap(resExperiment,facetBy="Factor")
```



The clustering and heatmaps generated from plotCorHeatmap allow us to identify which samples are most closely related and so both the reproducibilty of replicates and the similarity in binding profiles of different epigenetic marks.

Here we see that all replicates cluster tightly together, illustrating the relative reproducibilty of our replicates. Further to this we see that the samples broadly group into their respective tissues with DNA/Pol2 and Myc forming the Ch12 cluster. The ProB transcription factor (Pu1, Irf and Ebf) are seen to group together as expected and are found to cluster away from ProB histone marks. This suggests that these transcription factors may be less associated to these histone marks than the Ikaros ChIPs.

3.5 Conclusion

The analysis of the quality of this experiment's ChIP quality using ChIPQC has identified several informatics characteristics of the data as well as highlighted variability in quality across ChIPs and cell-lines.

The histone , RNA polymerase 2 and DNAse ChIPs all have high RIP% and SSD after blacklisting as expected from broader epigentic marks. Cross-coverage profiles illustrate the broader regions of enrichment for the histone marks, tighter profiles for the RNA pol2 and RNA pol2-ser2 marks, due to their narrow enrichment in TSSs and a sharp narrow profile for The DNAse ChIP. These epigenetic marks showed the characteristic enrichment within TSSs with RNApol2-ser2 most enriched for 3'UTRs.

The transcription factor ChIPS showed a much wider variability in RIP%, SSD and RelCC than seen for histone,pol2 and DNAse ChIPs. Pu1 and Irf were found to be highly efficient ChIPs and Pu1 was seen in its cross coverage scores to have a broader enrichment pattern than seen for for other TFs. The Ikaros were found to have acceptable but low enrichment for signal by all metrics, and the enrichment for Ikaros signal was seen to fit known concentrations of Ikaros within these cell-lines. The ProB Pu1, Ebf and Irf transcription factors were found to cluster away from Ikaros ChIPS indicating a greater co-occurence of binding among them than with Ikaros ChIP.

The inputs used in this study showed very different sources of artefact contamination. The Ch12 input showed significant pile-up of signal but following blacklisting it's SSD score dropped to that of a background level. This highlights that much of the artefact signalwith Ch12 input was from known mm9 blacklists and further artefact removal is unneccesary. The ProB input however shows peak like signal in its cross coverage profile, high level of duplication, a spike in its coverage histogram and remaining artefact signal following blacklisting. Taken togehter this suggests that the ProB input contains highly duplciated peak shaped spikes in signal and further curation and blacklisting should be performed on this input.

4 Advanced Topics

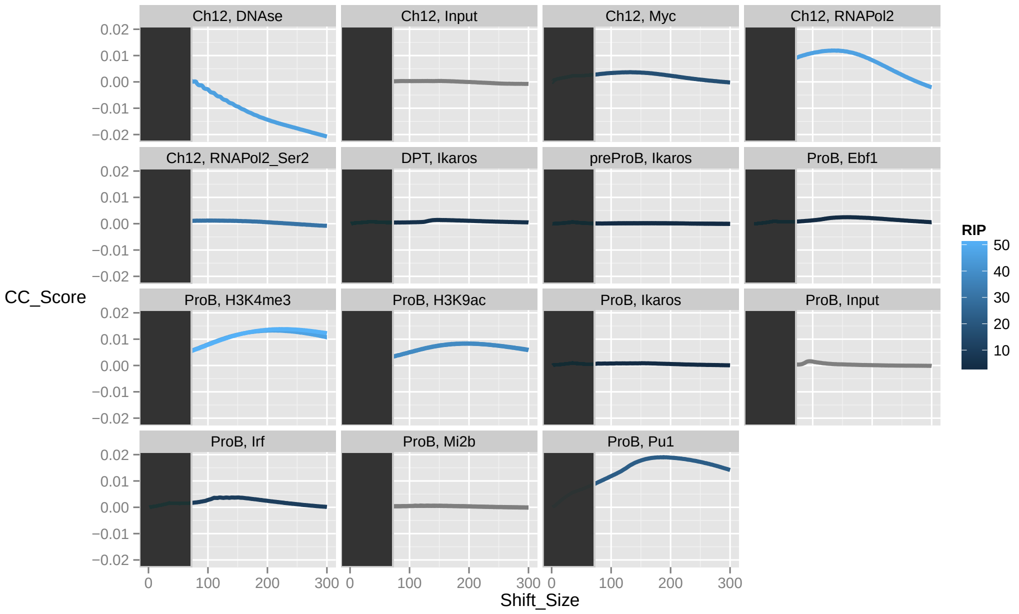
4.1 Providing additional data to ChIPQC plotting and reporting

The sample sheets for ChIPQC may contain the optional metadata columns "Tissue","Factor","Condition" and "Treatment". In order to allow the user to specify custom metadata for their plotting and reporting, the additional addMetadata argument can be supplied with a data frame of samplenames and associated additional metadata. The first column for addMetadata data frame must be SampleID and remaining columns maybe categorical or discrete data.

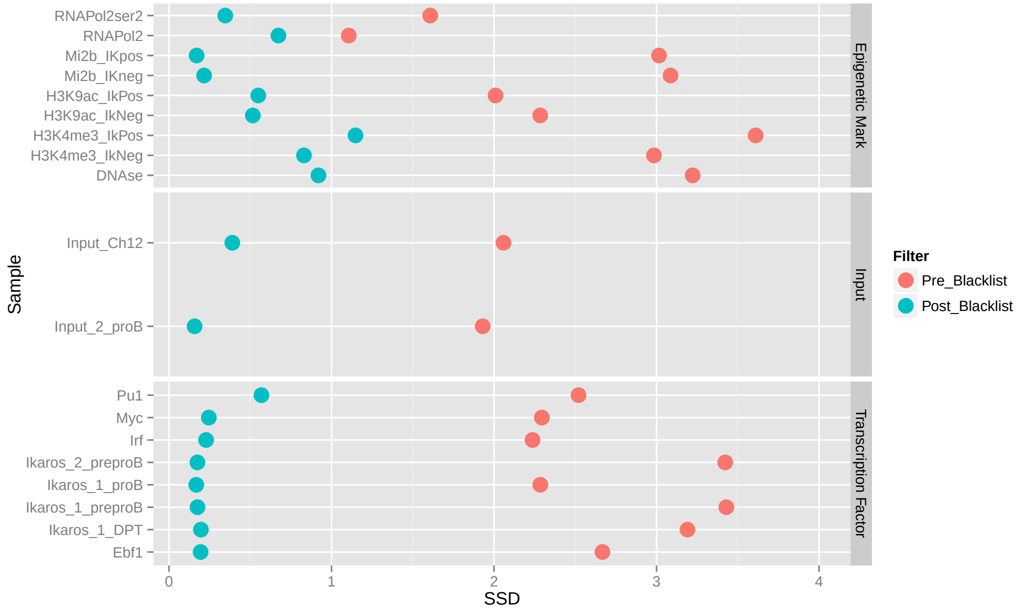
Here, first we illustrate the relationship between RIP and cross coverage scores by including RIP metrics as metadata and setting the column as colourBy argument.

Then we can use addMetadata argument to group SSD by their ChIP type to highlight longer SSD scores typically seen to Epigenetic marks.

```
> metrics <- QCmetrics(resExperiment)
> metricsMetadata <- data.frame(SampleID=rownames(metrics),RIP =metrics[,"RiP%",drop=T])
> plotCC(resExperiment,addMetaData = metricsMetadata,colourBy="RIP")
```



```
> metricsMetadata <- data.frame(SampleID=rownames(metrics),ChIPType=
+                         c(rep("Epigenetic Mark",1),
+                         rep("Transcription Factor",1),
+                         rep("Epigenetic Mark",4),
+                         rep("Transcription Factor",4),
+                         rep("Input",2),
+                         rep("Transcription Factor",1),
+                         rep("Epigenetic Mark",2),
+                         rep("Transcription Factor",2),
+                         rep("Epigenetic Mark",2)))
> plotSSD(resExperiment,addMetaData = metricsMetadata,facetBy="ChIPType")
```



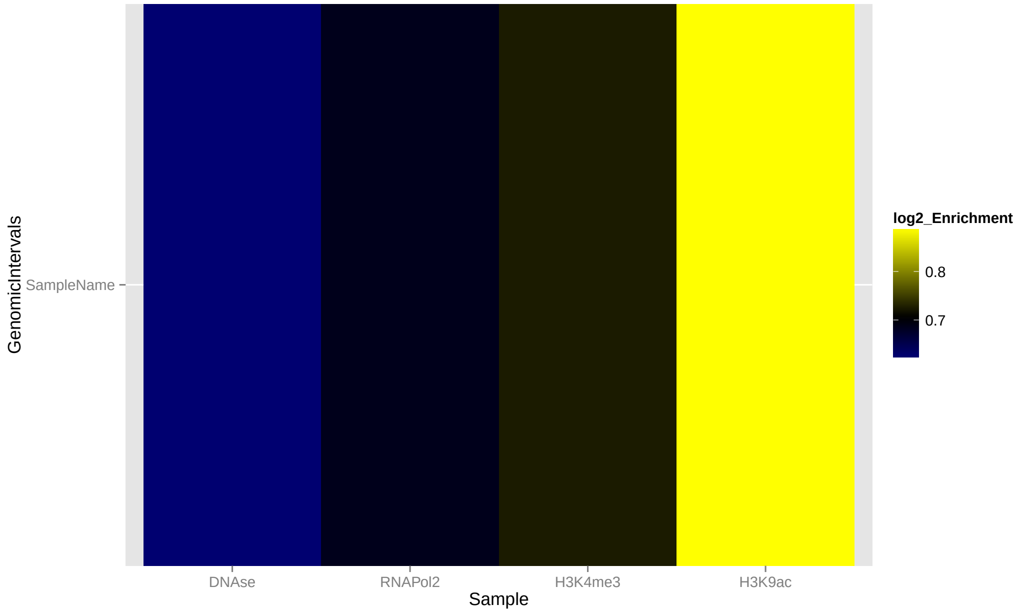
4.2 Specifying custom annotation

ChIPQC produces a simple metric for enrichment of signal in known genomic annotation.By default annotation is provided from the TxDB packages specified by "species" argument. ChIPQC also allows for the specification of genomic annotation in the form of GRanges objects and so enrichment in user defined regions can be assessed.

Annotation must be provided as a named list with the first element being "version" and the remaining list elements being GRanges objects.

Here we will run ChIPQCsample function with DNAse and histone mark peaks as the annotation.

```
> DNAsePeaks <- peaks(QCsample(resExperiment,"DNAse"))
> H3K4me3Peaks <- peaks(QCsample(resExperiment,"H3K4me3_IkPos"))
> H3K9acPeaks <- peaks(QCsample(resExperiment,"H3K9ac_IkPos"))
> RNAPol2Peaks <- peaks(QCsample(resExperiment,"RNAPol2"))
> customAnnotation <- list(version="custom",
+                         DNAse=DNAsePeaks,
+                         RNAPol2=RNAPol2Peaks,
+                         H3K4me3=H3K4me3Peaks,
+                         H3K9ac=H3K9acPeaks)
> #bamFile <- "/data/ChIPQC/Chr11_Ebf1DupMarked.bam
> bamFile <- "/Users/tcarroll/Downloads/Chr11_Ebf1DupMarked.bam"
> #exampleExpCA = ChIPQCsample(bamFile,peaks=NULL,annotation=customAnnotation,chromosomes="chr11")
> load("/Users/tcarroll/Downloads/exampleCustomAnnotation.RData")
> plotRegi(exampleExpCA)
```



The plotRegi now shows the enrichment over expected for the custom annotation. The order of display of custom annotation is dictated by the order in annotation list and so may rearranged as user desires.