Analysis plots for AAV plasmid library and coverage

Tomas Bjorklund

Mon Nov 2 13:46:12 2020

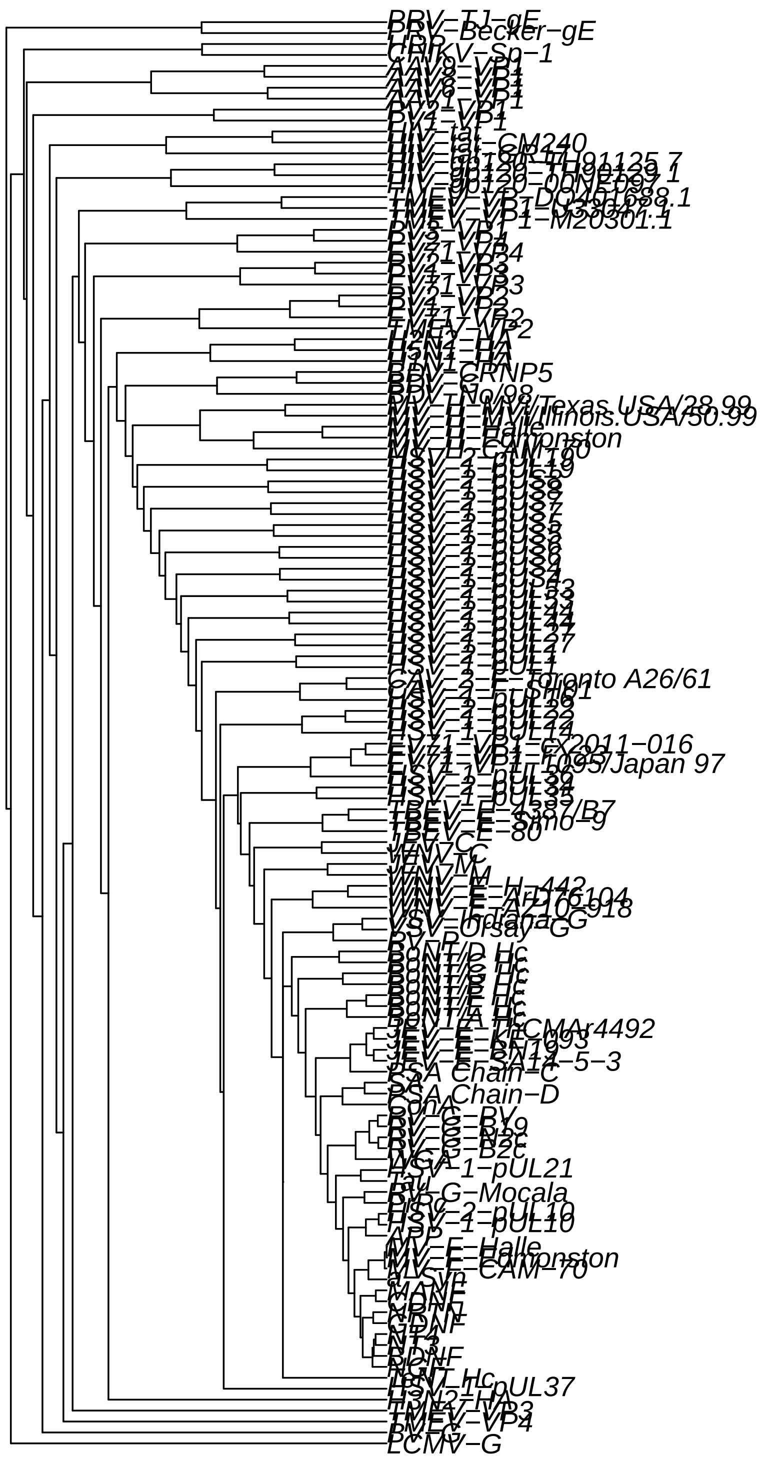
This script provides a number of measurements on the AAV plasmid library and the readouts in vitro & in vivo.

```
suppressPackageStartupMessages(library(knitr))
```

Generation of phylotree for selected proteins

```
in.fasta <- readFasta("input/DNA-lib_RetrogradeTransport.fasta")
aaSeqs <- Biostrings::translate(sread(in.fasta), genetic.code = GENETIC_CODE,
    if.fuzzy.codon = "solve")
name.table <- data.table(as.character(ShortRead::id(in.fasta)), gsub("([ ])",
    "_", tstrsplit(ShortRead::id(in.fasta), ",", fixed = TRUE)[[6]]))
setnames(name.table, c("V1", "V2"), c("FullName", "ShortName"))
names(aaSeqs) <- name.table$ShortName
saveRDS(name.table, file = "data/geneNames.rds")
aaLib.file <- tempfile(pattern = "aalib_", tmpdir = tempdir(), fileext = ".fasta")
writeXStringSet(aaSeqs, aaLib.file, format = "fasta")


tree.file <- tempfile(pattern = "results_", tmpdir = tempdir(), fileext = ".tree")

sys.out <- system(paste("/usr/bin/usearch -cluster_agg ", aaLib.file, " -treeout  /home/rstudio/data/tree.ph
    " 2>&1", sep = ""), intern = TRUE, ignore.stdout = FALSE)
tree <- read_newick_phylo("/home/rstudio/data/tree.phy", simplify = FALSE, missing_edge_length = NA)
# tmp.list <- tree$edge.length tmp.list <- as.integer(tmp.list)
# tree$edge.length <- tmp.list

tree.calib <- makeChronosCalib(tree, age.min = 0, age.max = max(tree$edge.length))
dendrogram <- chronos(tree, calibration = tree.calib)


Setting initial dates...
Fitting in progress... get a first set of estimates
         Penalised log-lik = -187154.9
Optimising rates... dates... -187154.9

Done.
```

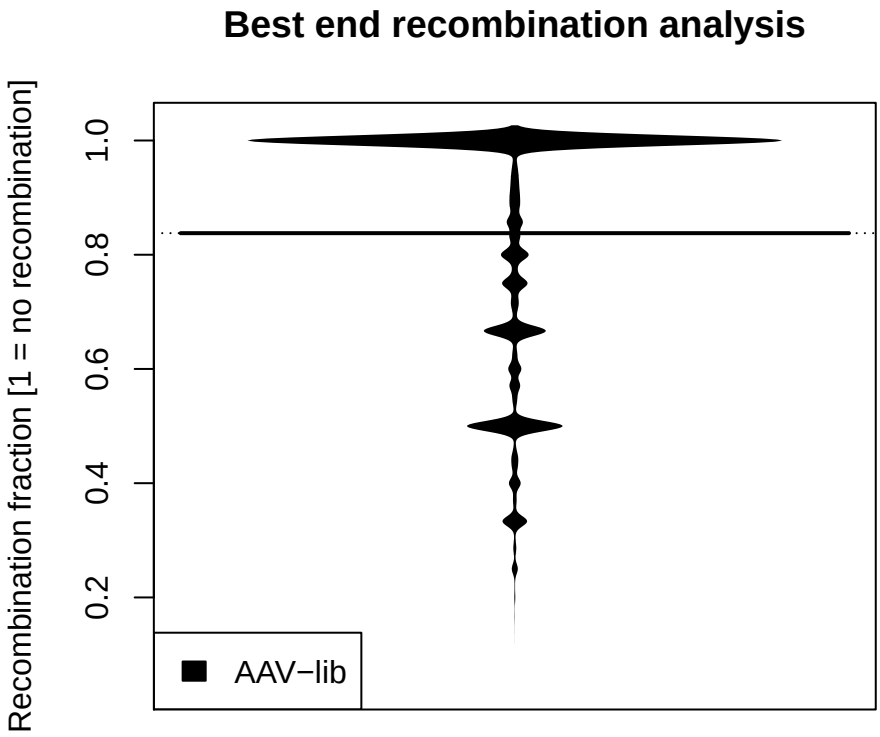
```
plot(dendrogram)
```


CAV−2−F−Toronto A26/61
MV−H−Edmonston
VSV−Orsay−G
BoNT/A Hc
PSA Chain−C
PSA Chain−D
MV−F−Edmonston
LCMV−G

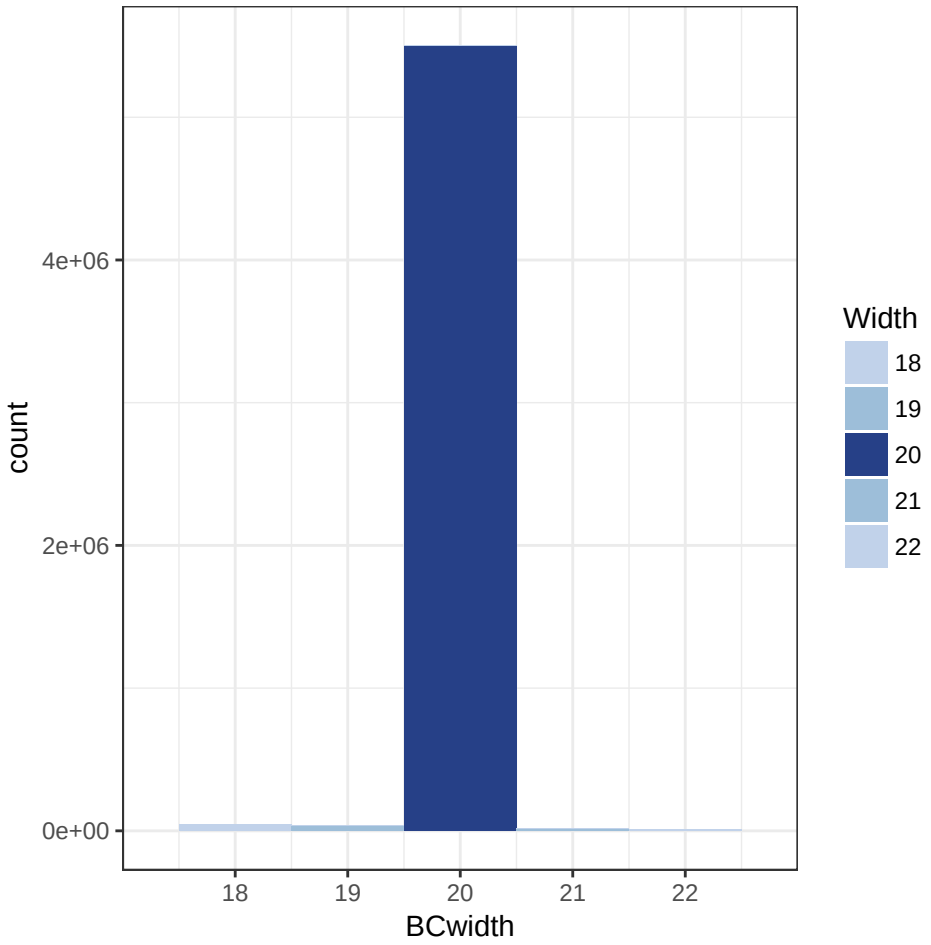
```
opts_chunk$set(fig.width = 5, fig.height = 5)  #Full height 11
```

Generation plots for library purity

```
complete.ranges <- readRDS("output/completeLibraryRanges.rds")
purity.table <- data.table(mcols(complete.ranges)$mCount/mcols(complete.ranges)$tCount)
purity.table$BCwidth <- width(mcols(complete.ranges)$BC)

beanplot(data = purity.table$V1, ll = 0.04, what = c(1, 1, 1, 0), bw = "nrd0",
    log = "", main = "Best end recombination analysis", ylab = "Recombination fraction [1 = no recombinatio
    border = NA, col = list("black", c("grey", "white")))
legend("bottomleft", fill = c("black"), legend = c("AAV-lib"))
```




```
fill.values <- c(A = rgb(193, 210, 234, maxColorValue = 255), B = rgb(157, 190,
    217, maxColorValue = 255), c = rgb(38, 64, 135, maxColorValue = 255), D = rgb(157,
    190, 217, maxColorValue = 255), E = rgb(193, 210, 234, maxColorValue = 255))
names(fill.values) <- c("18", "19", "20", "21", "22")

ggplot(purity.table, aes(x = BCwidth, fill = as.character(BCwidth))) + geom_histogram(binwidth = 1,
    boundary = 0.5) + theme_bw() + scale_fill_manual(name = "Width", values = fill.values) +
    scale_x_continuous(limit = c(17, 23), breaks = c(seq(18, 22, 1)), expand = c(0,
        0))
```

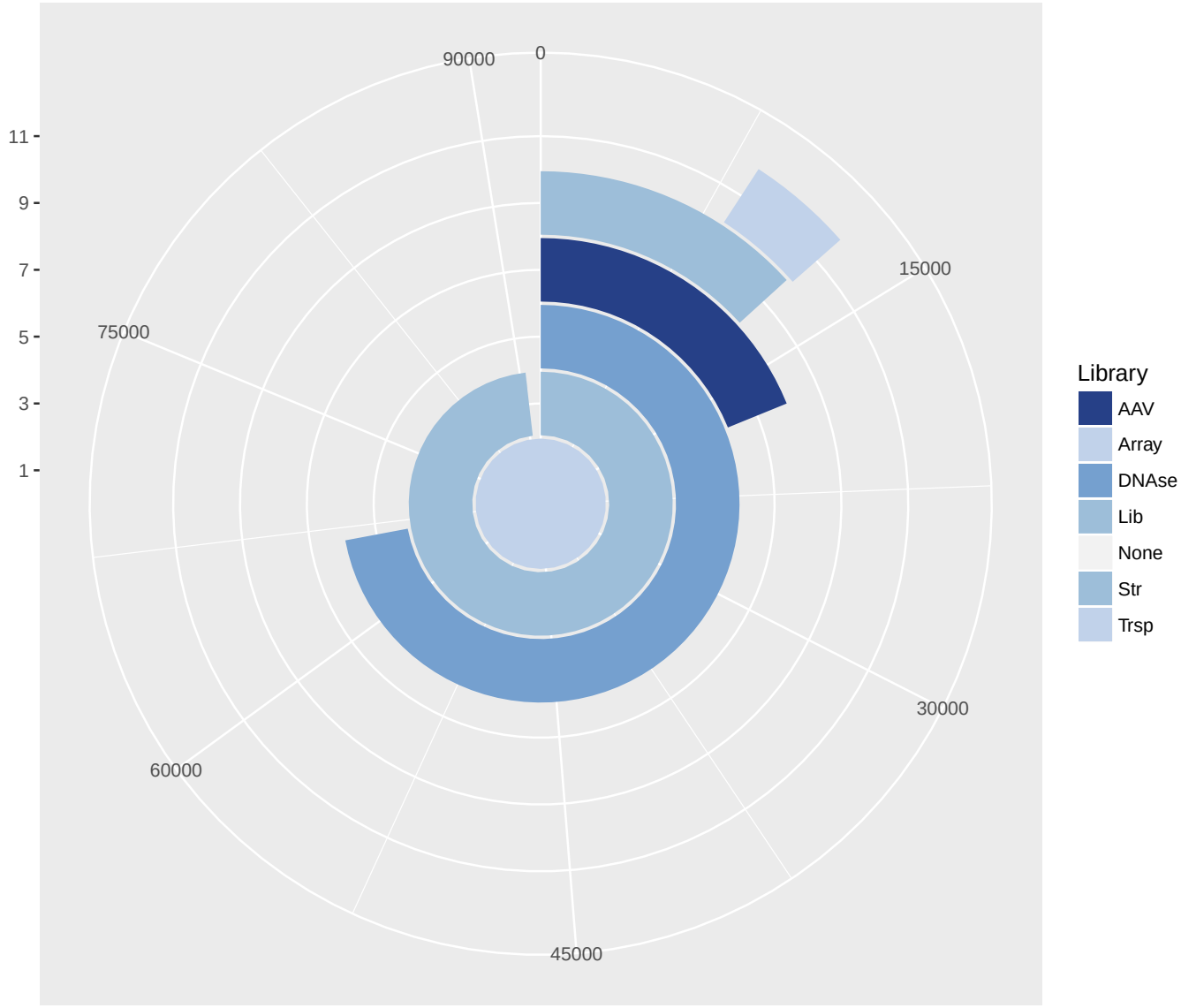



```
opts_chunk$set(fig.width = 8, fig.height = 8)
```

Plot Venn diagrams of fragments

```
load("data/LUTdna.rda")
complete.library <- readRDS("data/allSamplesDataTable.RDS")
setkey(complete.library, Group)
complete.library <- complete.library[-grep("4wks", Group)]
seq.arry <- LUT.dna$LUTnr
seq.lib <- unique(complete.library[J("DNA_pscAAVlib")]$LUTnr)
seq.AAV <- unique(complete.library[J("mRNA_All")]$LUTnr)
seq.DNAse <- unique(complete.library[grep("DNA_AAVlib_DNAse", Group)]$LUTnr)
seq.str <- unique(complete.library[grep("Str", Group)]$LUTnr)
seq.Trsp <- unique(complete.library[grep("SN|Ctx|Th", Group)]$LUTnr)

venn.area1 <- length(seq.arry)
venn.area2 <- length(seq.lib)
venn.area3 <- length(seq.DNAse)
venn.area4 <- length(seq.AAV)
venn.area5 <- length(seq.str)
venn.area6 <- length(seq.Trsp)


isect.Str_Trsp <- length(intersect(seq.str, seq.Trsp))
```

```
venn.n12 <- length(intersect(seq.arry, seq.lib))
venn.n23 <- length(intersect(seq.lib, seq.DNAse))

venn.n13 <- length(intersect(seq.arry, seq.DNAse))
venn.n123 <- length(intersect(intersect(seq.arry, seq.lib), seq.DNAse))


output.table <- data.frame(NameArray = character(), NameLib = character(), NameDNAse = character(),
    NameAAV = character(), NameStr = character(), NameTrsp = character(), ArrayStart = numeric(),
    ArrayEnd = numeric(), LibStart = numeric(), LibEnd = numeric(), DNAseStart = numeric(),
    DNAseEnd = numeric(), AAVStart = numeric(), AAVend = numeric(), StrStart = numeric(),
    StrEnd = numeric(), TrspStart = numeric(), TrspEnd = numeric(), stringsAsFactors = FALSE)
output.table[1:3, 1] <- c("Array", "None", "None")
output.table[1:3, 2] <- c("Lib", "None", "None")
output.table[1:3, 3] <- c("DNAse", "None", "None")
output.table[1:3, 4] <- c("AAV", "None", "None")
output.table[1:3, 5] <- c("Str", "None", "None")
output.table[1:3, 6] <- c("None", "Trsp", "None")
output.table[1:3, 7:18] <- 0
output.table$ArrayEnd[1] <- output.table$LibEnd[2] <- output.table$DNAseEnd[2] <- output.table$AAVend[2] <-
output.table$LibStart[2] <- output.table$LibEnd[1] <- length(intersect(seq.arry,
    seq.lib))
output.table$DNAseStart[2] <- output.table$DNAseEnd[1] <- length(intersect(seq.lib,
    seq.DNAse))
output.table$AAVStart[2] <- output.table$AAVend[1] <- length(intersect(seq.lib,
    seq.AAV))
output.table$StrStart[2] <- output.table$StrEnd[1] <- length(intersect(seq.AAV,
    seq.str))
output.table$TrspStart[2] <- output.table$TrspEnd[1] <- length(seq.str) - length(intersect(seq.str,
    seq.Trsp))
output.table$TrspStart[3] <- output.table$TrspEnd[2] <- output.table$TrspStart[2] +
    length(seq.Trsp)


fill.values <- c(Array = rgb(193, 210, 234, maxColorValue = 255), Lib = rgb(157,
    190, 217, maxColorValue = 255), DNAse = rgb(117, 160, 207, maxColorValue = 255),
    AAV = rgb(38, 64, 135, maxColorValue = 255), Str = rgb(157, 190, 217, maxColorValue = 255),
    Trsp = rgb(193, 210, 234, maxColorValue = 255), None = rgb(255, 255, 255,
        maxColorValue = 255, alpha = 0))


ggplot(output.table) + scale_x_continuous(limit = c(0, 12), breaks = c(seq(1,
    11, 2)), expand = c(0, 0)) + scale_y_continuous(breaks = c(seq(0, 90000,
    15000))) + scale_fill_manual(name = "Library", values = fill.values) + theme(aspect.ratio = 1) +
    geom_rect(data = output.table, aes(fill = NameTrsp, ymax = TrspEnd, ymin = TrspStart,
        xmax = 11.95, xmin = 10.05)) + geom_rect(data = output.table, aes(fill = NameStr,
    ymax = StrEnd, ymin = StrStart, xmax = 9.95, xmin = 8.05)) + geom_rect(data = output.table,
    aes(fill = NameAAV, ymax = AAVend, ymin = AAVStart, xmax = 7.95, xmin = 6.05)) +
    geom_rect(data = output.table, aes(fill = NameDNAse, ymax = DNAseEnd, ymin = DNAseStart,
        xmax = 5.95, xmin = 4.05)) + geom_rect(data = output.table, aes(fill = NameLib,
    ymax = LibEnd, ymin = LibStart, xmax = 3.95, xmin = 2.05)) + geom_rect(data = output.table,
    aes(fill = NameArray, ymax = ArrayEnd, ymin = ArrayStart, xmax = 1.95, xmin = 0)) +
    coord_polar(theta = "y")
```



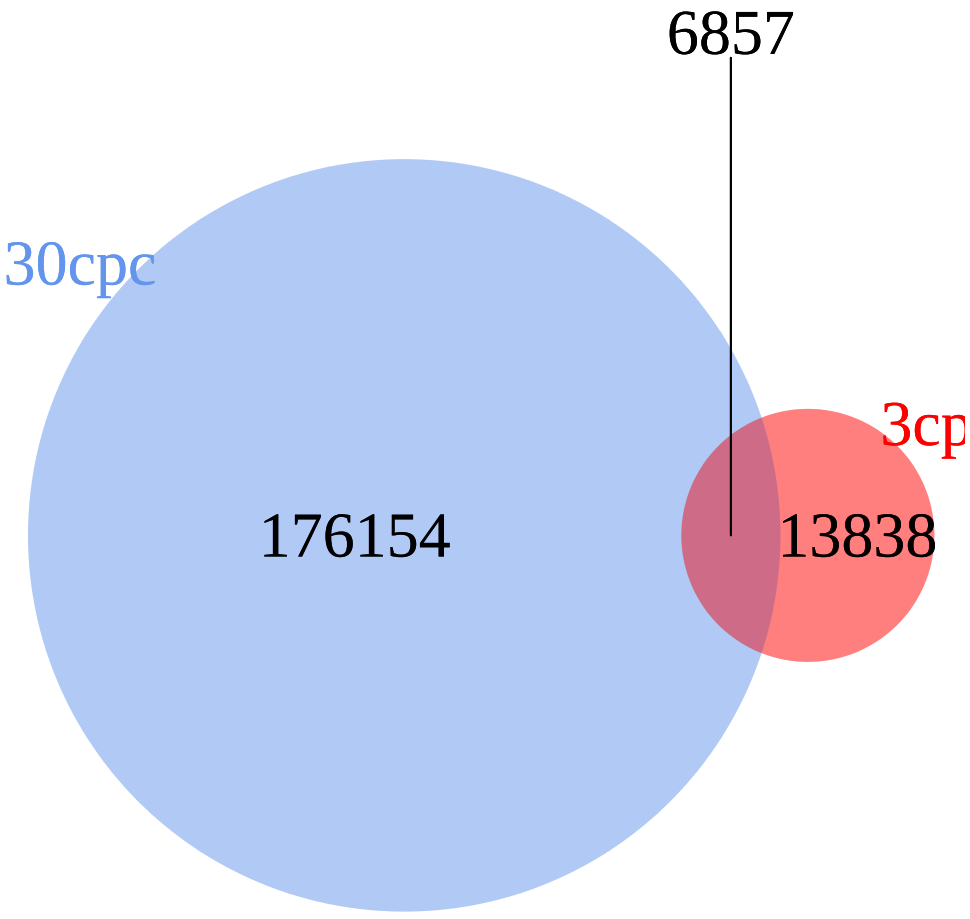

```
opts_chunk$set(fig.width = 5, fig.height = 5)
```

```
knitr::kable(output.table, format = "latex", booktabs = T) %>% kable_styling(latex_options = c("striped",
    "scale_down", "repeat_header")) %>% landscape()
```

NameArray	NameLib	NameDNAse	NameAAV	NameStr	NameTrsp	ArrayStart	ArrayEnd	LibStart	LibEnd	DNAseStart	DNAseEnd	AAVStart	AAVend	StrStart	StrEnd	TrspStart	TrspEnd
Array	Lib	DNAse	AAV	Str	None	0	92343	0	90635	0	66531	0	17401	0	12240	0	8483
None	None	None	None	None	Trsp	0	0	90635	92343	66531	92343	17401	92343	12240	92343	8483	12474
None	None	None	None	None	None	0	0	0	0	0	0	0	0	0	0	12474	92343

Barcode Venn diagrams for 30cpc and 3cpc DNAse resistant libraries

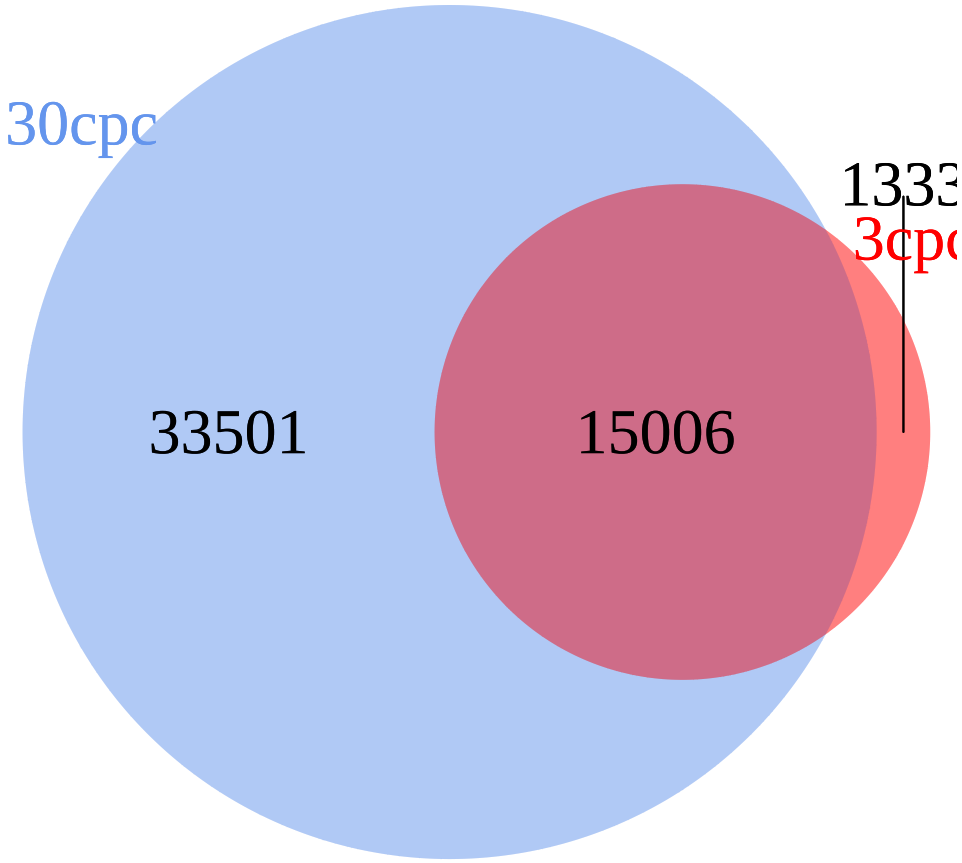
```
total.30cpc <- unique(complete.library[grep("DNA_AAVlib_DNAse_30cpc", Group)]$BC)
total.3cpc <- unique(complete.library[grep("DNA_AAVlib_DNAse_3cpc", Group)]$BC)

total.30cpc <- names(table(strsplit(paste(total.30cpc, collapse = ","), ",")))
total.3cpc <- names(table(strsplit(paste(total.3cpc, collapse = ","), ",")))

venn.area1 <- length(total.30cpc)
venn.area2 <- length(total.3cpc)

venn.n12 <- length(intersect(total.30cpc, total.3cpc))

venn.colors <- c("cornflower blue", "red")
grid.newpage()
venn.plot <- draw.pairwise.venn(area1 = venn.area1, area2 = venn.area2, cross.area = venn.n12,
    scaled = TRUE, fill = venn.colors, alpha = 0.3, lty = "blank", cex = 2,
    cat.cex = 2, cat.col = venn.colors, category = c("30cpc", "3cpc"))
grid.draw(venn.plot)
```



Fragment Venn diagrams for 30cpc and 3cpc DNAse resistant libraries

```
total.30cpc <- unique(complete.library[grep("DNA_AAVlib_DNAse_30cpc", Group)]$Sequence)
total.3cpc <- unique(complete.library[grep("DNA_AAVlib_DNAse_3cpc", Group)]$Sequence)
```

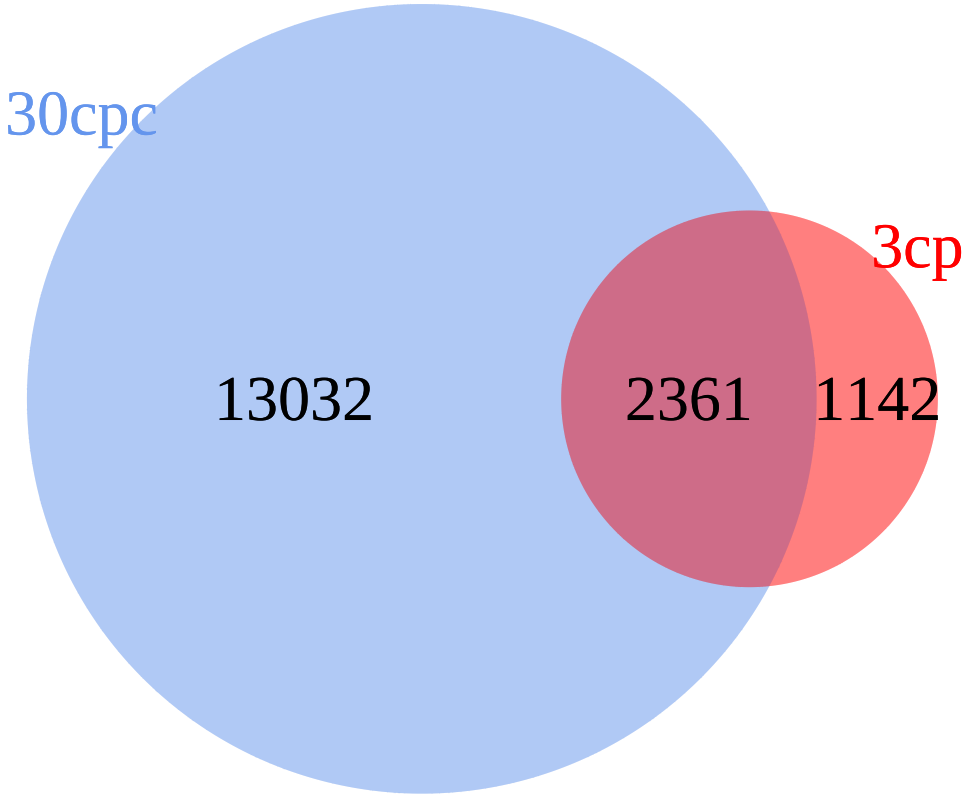
```
venn.area1 <- length(total.30cpc)
venn.area2 <- length(total.3cpc)

venn.n12 <- length(intersect(total.30cpc, total.3cpc))


grid.newpage()
venn.plot <- draw.pairwise.venn(area1 = venn.area1, area2 = venn.area2, cross.area = venn.n12,
    scaled = TRUE, fill = venn.colors, alpha = 0.3, lty = "blank", cex = 2,
    cat.cex = 2, cat.col = venn.colors, category = c("30cpc", "3cpc"))
grid.draw(venn.plot)
```



Barcode Venn diagrams for 30cpc and 3cpc infective libraries

```
RNA.library <- complete.library[-grep("DNAse", Group)]
total.30cpc <- unique(RNA.library[grep("30cpc", Group)]$BC)
total.3cpc <- unique(RNA.library[grep("3cpc", Group)]$BC)

total.30cpc <- names(table(strsplit(paste(total.30cpc, collapse = ","), ",")))
total.3cpc <- names(table(strsplit(paste(total.3cpc, collapse = ","), ",")))

venn.area1 <- length(total.30cpc)
```

```
venn.area2 <- length(total.3cpc)

venn.n12 <- length(intersect(total.30cpc, total.3cpc))

venn.colors <- c("cornflower blue", "red")
grid.newpage()
venn.plot <- draw.pairwise.venn(area1 = venn.area1, area2 = venn.area2, cross.area = venn.n12,
    scaled = TRUE, fill = venn.colors, alpha = 0.3, lty = "blank", cex = 2,
    cat.cex = 2, cat.col = venn.colors, category = c("30cpc", "3cpc"))
grid.draw(venn.plot)
```



Fragment Venn diagrams for 30cpc and 3cpc infective libraries

```
total.30cpc <- unique(RNA.library[grep("30cpc", Group)]$Sequence)
total.3cpc <- unique(RNA.library[grep("3cpc", Group)]$Sequence)


venn.area1 <- length(total.30cpc)
venn.area2 <- length(total.3cpc)

venn.n12 <- length(intersect(total.30cpc, total.3cpc))


grid.newpage()
```

```
venn.plot <- draw.pairwise.venn(area1 = venn.area1, area2 = venn.area2, cross.area = venn.n12,
    scaled = TRUE, fill = venn.colors, alpha = 0.3, lty = "blank", cex = 2,
    cat.cex = 2, cat.col = venn.colors, category = c("30cpc", "3cpc"))
grid.draw(venn.plot)
```