GSE132177 AS analysis

Xiong Wang Department of Laboratory Medicine, Tongji Hospital, Tongji Medical College, Huazhong University of Science and Technology, Wuhan 430030, China. Email: wangxiong@tjh.tjmu.edu.cn

GEO database link: GSE132177 , ENA database link： PRJNA546262

1. Create conda environment

```
#conda install mamba
conda create -n rna
conda activate rna
mamba install -y axel fastqc multiqc trim-galore hisat2 subread samtools salmon star
```

2. Download fq files using axel

```
#download filereport_read_run_PRJNA546262_tsv from ENA
database:https://www.ebi.ac.uk/ena/browser/view/PRJNA546262
#upload this file to server.

#conda activate rna
cd ~/project/AS/AD_GSE132177/data/rawdata

head -1 filereport_read_run_PRJNA* |tr '\t' '\n' |cat -n

#create ftp link file
cat filereport_read_run_PRJNA* |awk -F '\t' 'NR>1 {print $4}' |tr ';' '\n' |grep '_'
>fq.url

#create axel_download.sh file
vim axel_download.sh
###
cat fq.url |while read id
do
  axel -n 30 ${id}
done
###

nohup sh axel_download.sh >axel_download.log

#Check md5 nunmber
cat filereport_read_run_PRJNA* | awk -F'\t' 'NR>1{print$3}' | tr ';' '\n' >md51
cat filereport_read_run_PRJNA* | awk -F'\t' 'NR>1{print$4}' |tr ';' '\n' |awk -F'/'
'{print$NF}' >md52

paste -d'  ' md51  md52 |grep '_' >md5.txt

nohup md5sum -c md5.txt >check &
cat check
#SRR9201192_1.fastq.gz: OK
```

```
#SRR9201192_2.fastq.gz: OK
#SRR9201195_1.fastq.gz: OK
#SRR9201195_2.fastq.gz: OK
#SRR9201198_1.fastq.gz: OK
#SRR9201198_2.fastq.gz: OK
#SRR9201201_1.fastq.gz: OK
#SRR9201201_2.fastq.gz: OK
#SRR9201204_1.fastq.gz: OK
#SRR9201204_2.fastq.gz: OK
#SRR9201207_1.fastq.gz: OK
#SRR9201207_2.fastq.gz: OK
#all files were ok
```

3. Data QC

The fastqc was used to perform QC check

```
#conda activate rna
cd ~/project/AS/AD_GSE132177/data
mkdir qc

#define qcdir and fqdir
qcdir=~/project/AS/AD_GSE132177/data/qc
fqdir=~/project/AS/AD_GSE132177/data/rawdata

fastqc -t 20 -o $qcdir $fqdir/*.fastq.gz

cd ~/project/AS/AD_GSE132177/data/qc
multiqc *.zip
# download multiqc_report.html to check the QC result.
```

4. Data cleaning

```
#conda activate rna
cd ~/project/AS/AD_GSE132177/data
mkdir -p cleandata/trim_galore
cd cleandata/trim_galore

#create sample id list file
ls ../../rawdata/*.gz | awk -F'/' '{print$4}' | awk -F '_' '{print$1}'| uniq
>sample.ID.txt

#create trim_galore.sh
vim trim_galore.sh
###
rawdata=~/project/AS/AD_GSE132177/data/rawdata
cleandata=~/project/AS/AD_GSE132177/data/cleandata/trim_galore

cat ~/project/AS/AD_GSE132177/data/cleandata/trim_galore/sample.ID.txt | while read id
do
trim_galore --phred33 -q 20 --length 36 --stringency 3 --fastqc --paired --max_n 3 -o
${cleandata} ${rawdata}/${id}_1.fastq.gz ${rawdata}/${id}_2.fastq.gz
```

```
done
###

nohup sh trim_galore.sh >trim_galore.log &
```

5. Alignment

5.1 Reference genome

```
#Ensembl: http://asia.ensembl.org/index.html
#ftp://ftp.ensembl.org/pub/release-105/fasta/Mus_musculus/dna/

cd ~/database/genome/Ensembl/Mus_musculus/GRCm39_release105

#DNA
wget -c http://ftp.ensembl.org/pub/release-
105/fasta/mus_musculus/dna/Mus_musculus.GRCm39.dna.primary_assembly.fa.gz

#cDNA
wget -c http://ftp.ensembl.org/pub/release-
105/fasta/mus_musculus/cdna/Mus_musculus.GRCm39.cdna.all.fa.gz

#gtf
wget -c ftp://ftp.ensembl.org/pub/release-
105/gtf/Mus_musculus/Mus_musculus.GRCm39.105.gtf.gz
```

5.2 STAR alignment

5.2.1 STAR index

```
cd ~/database/genome/Ensembl/Mus_musculus/GRCm39_release105/
mkdir -p index/STAR
```

```
#conda activate rna
STAR \
--runMode genomeGenerate \
--runThreadN 40 \
--genomeDir ~/database/genome/Ensembl/Mus_musculus/GRCm39_release105/index/STAR \
--genomeFastaFiles
~/database/genome/Ensembl/Mus_musculus/GRCm39_release105/Mus_musculus.GRCm39.dna.primary_
assembly.fa \
--sjdbGTFfile
~/database/genome/Ensembl/Mus_musculus/GRCm39_release105/Mus_musculus.GRCm39.105.gtf
```

If rMATS started with fq files, only index files of STAR were required.

5.2.2 STAR alignment

```
#conda activate rna
cd ~/project/AS/AD_GSE132177/Mapping/STAR

#creat sample id list file
```

```
ls ../../data/cleandata/trim_galore/*.gz | awk -F'/' '{print$6}' | awk -F '_'
'{print$1}'| uniq >sample.ID.txt

# create STAR.sh
vim STAR.sh
###
index=~/database/genome/Ensembl/Mus_musculus/GRCm39_release105/index/STAR
cleandata=~/project/AS/AD_GSE132177/data/cleandata/trim_galore

cat ~/project/AS/AD_GSE132177/Mapping/STAR/sample.ID.txt | while read id
do
STAR --runThreadN 4 --genomeDir ${index} \
 --readFilesIn ${cleandata}/${id}_1_val_1.fq.gz ${cleandata}/${id}_2_val_2.fq.gz \
 --readFilesCommand zcat \
 --outSAMtype BAM SortedByCoordinate \
 --outFileNamePrefix ./${id} \
 --outBAMsortingThreadN 4
done
## runThreadN should be small, otherwise error occurred
###

nohup sh STAR.sh >STAR.log &
```

5.3 Gene matrix

The featurecounts was used to generate expression matrix on gene level.

```
#conda activate rna
cd ~/project/AS/AD_GSE132177/expression

##link the bam file to the current directory
ln -s ~/project/AS/AD_GSE132177/Mapping/STAR/*.out.bam ./
#rename the bam files as WT1-3,AD1-3.bam

mkdir featurecounts
cd ~/project/AS/AD_GSE132177/expression/featurecounts

##define gtf and input variables
gtf=~/database/genome/Ensembl/Mus_musculus/GRCm39_release105/Mus_musculus.GRCm39.105.gtf
inputdir=~/project/AS/AD_GSE132177/expression
featureCounts=/home/data/wangxiong/miniconda3/envs/rna/bin/featureCounts

$featureCounts -T 8 -p -t exon -g gene_id -a $gtf -o all.id.txt $inputdir/*.bam
#finish in 5 minutes

#QC
multiqc all.id.txt.summary
#download multiqc_report.html to check the QC result

#gene expression matrix
cat all.id.txt | cut -f 1,7- > counts.txt

#colnames modification
```

```
less -S all.id.txt |grep -v '#' |cut -f 1,7- |sed
's#/home/data/wangxiong/project/AS/AD_GSE132177/expression/##g' |sed 's#.bam##g'
>raw_counts.txt

head -3 raw_counts.txt  |column -t
#Geneid              AD1  AD2  AD3  WT1  WT2  WT3
#ENSMUSG00000102628  9    12   12   18   9    17
#ENSMUSG00000100595  0    0    0    0    0    0
```

5.4 Transcript matrix

The salmon was used to generate transcript expression matrix from clean fq files.

5.3.1 salmon index

```
#conda activate rna
cd ~/database/genome/Ensembl/Mus_musculus/GRCm39_release105/index
salmon index -t
~/database/genome/Ensembl/Mus_musculus/GRCm39_release105/Mus_musculus.GRCm39.cdna.all.fa.
gz -i Mus_musculus.GRCm39.cdna.all.salmon
```

5.3.2 transcript matrix

```
cd ~/project/AS/AD_GSE132177/expression

#creat sample id list
ls ~/project/AS/AD_GSE132177/data/rawdata/*.gz | awk -F'/' '{print$10}' | awk -F '_'
'{print$1}'| uniq >sample.ID.txt

# create salmon.sh
vim salmon.sh
###
index=~/database/genome/Ensembl/Mus_musculus/GRCm39_release105/index/Mus_musculus.GRCm39.
cdna.all.salmon/
input=~/project/AS/AD_GSE132177/data/cleandata/trim_galore
outdir=~/project/AS/AD_GSE132177/expression/salmon

cat ~/project/AS/AD_GSE132177/expression/sample.ID.txt |while read id
do
salmon quant -i ${index} -l A -1 ${input}/${id}_1_val_1.fq.gz -2
${input}/${id}_2_val_2.fq.gz -p 5 -o ${outdir}/${id}.quant
done
###

nohup bash salmon.sh 1>salmon.log 2>&1 &
```

5.3.3 merge matrix

```
cd ~/project/AS/AD_GSE132177/expression/salmon

ls SRR*/quant.sf | awk -F '.' '{print$1}' >sample.list
```

```
cat sample.list
#SRR9201192
#SRR9201195
#SRR9201198
#SRR9201201
#SRR9201204
#SRR9201207

cat ~/project/AS/AD_GSE132177/expression/salmon/sample.list | while read id
do
  awk '{print$1,$5}' ${id}.quant/quant.sf |sed 's/NumReads/'${id}'/' | sed 's/ /\t/' >
${id}.count
done

paste -d '\t' *.count | awk -F '\t' '{print$1,$2,$4,$6,$8,$10,$12}' | tr ' ' '\t'
>transcript.raw_counts.txt
```

6. AS analysis using rMATS

6.2.1 Prepare data

```
conda create -n rmats
conda activate rmats
conda install rmats, rmats2sashimiplot

cd ~/project/AS/AD_GSE132177/splicing/rmats2

##link aligned and sorted bam files after STAR alignment here, and renamed them as
WT1_1.bam and AD1_3.bam
ln -s ~/project/AS/AD_GSE132177/Mapping/STAR/*.out.bam ./

#link the gtf file here
ln -s
~/database/genome/Ensembl/Mus_musculus/GRCm39_release105/Mus_musculus.GRCm39.105.gtf ./

#STAR index was not required for rMATS using bam files as input
```

```
#create b1.txt to save case group bam files
vim b1.txt
###
./AD_1.bam,./AD_2.bam,./AD_3.bam
###
```

```
#create b1.txt to save control group bam files
vim b2.txt
###
./WT_1.bam,./WT_2.bam,./WT_3.bam
###
```

6.2.2 Run rMATS 4.1.2

```
rmats.py --b1 ./b1.txt --b2 ./b2.txt --gtf ./Mus_musculus.GRCm39.105.gtf -t paired --
readLength 150 --nthread 8 --od ./output --tmp ./tmp_output
```

```
(rmats) wangxiong 12:04:18 ~/project/AS/AD_GSE132177/splicing/rmats2
$ rmats.py --b1 ./b1.txt --b2 ./b2.txt --gtf ./Mus_musculus.GRCm39.105.gtf -t paired --readLength 150 --nthread 8 --od .
/output --tmp ./tmp_output
gtf: 15.31102728843689
There are 55414 distinct gene ID in the gtf file
There are 142435 distinct transcript ID in the gtf file
There are 34375 one-transcript genes in the gtf file
There are 842144 exons in the gtf file
There are 26924 one-exon transcripts in the gtf file
There are 21849 one-transcript genes with only one exon in the transcript
Average number of transcripts per gene is 2.570379
Average number of exons per transcript is 5.912479
Average number of exons per transcript excluding one-exon tx is 7.057510
Average number of gene per geneGroup is 7.451096
statistic: 0.014858245849609375

read outcome totals across all BAMs
USED: 377009538
NOT_PAIRED: 120363
NOT_NH_1: 70926913
NOT_EXPECTED_CIGAR: 12400755
NOT_EXPECTED_READ_LENGTH: 43689738
NOT_EXPECTED_STRAND: 0
EXON_NOT_MATCHED_TO_ANNOTATION: 18423870
JUNCTION_NOT_MATCHED_TO_ANNOTATION: 995505
CLIPPED: 33740376
total: 557307058
outcomes by BAM written to: ./tmp_output/2022-04-16-12_04_45_591245_read_outcomes_by_bam.txt

novel: 466.53388929367065
The splicing graph and candidate read have been saved into ./tmp_output/2022-04-16-12_04_45_591245_*.rmats
save: 9.92946171760559
loadsg: 0.2182481288909912

==========
Done processing each gene from dictionary to compile AS events
Found 52860 exon skipping events
Found 6498 exon MX events
Found 9978 alt SS events
There are 6302 alt 3 SS events and 3676 alt 5 SS events.
Found 4246 RI events
==========

ase: 1.4364874362945557
count: 13.768274545669556
Processing count files.
Done processing count files.
```

6.2.3 Check resuts

```
cd ~/project/AS/AD_GSE132177/splicing/rmats2/output
ll -h
```

```
(rmats) wangxiong 12:15:30 ~/project/AS/AD_GSE132177/splicing/rmats2/output
$ ls -h
A3SS.MATS.JCEC.txt              fromGTF.novelJunction.RI.txt      JCEC.raw.input.A5SS.txt  MXE.MATS.JC.txt
A3SS.MATS.JC.txt                fromGTF.novelJunction.SE.txt      JCEC.raw.input.MXE.txt   RI.MATS.JCEC.txt
A5SS.MATS.JCEC.txt              fromGTF.novelSpliceSite.A3SS.txt  JCEC.raw.input.RI.txt    RI.MATS.JC.txt
A5SS.MATS.JC.txt                fromGTF.novelSpliceSite.A5SS.txt  JCEC.raw.input.SE.txt    SE.MATS.JCEC.txt
fromGTF.A3SS.txt                fromGTF.novelSpliceSite.MXE.txt   JC.raw.input.A3SS.txt    SE.MATS.JC.txt
fromGTF.A5SS.txt                fromGTF.novelSpliceSite.RI.txt    JC.raw.input.A5SS.txt    summary.txt
fromGTF.MXE.txt                 fromGTF.novelSpliceSite.SE.txt    JC.raw.input.MXE.txt     tmp
fromGTF.novelJunction.A3SS.txt  fromGTF.RI.txt                    JC.raw.input.RI.txt
fromGTF.novelJunction.A5SS.txt  fromGTF.SE.txt                    JC.raw.input.SE.txt
fromGTF.novelJunction.MXE.txt   JCEC.raw.input.A3SS.txt           MXE.MATS.JCEC.txt
```

```
cat summary.txt
```

```
$ cat summary.txt
EventType       TotalEventsJC   TotalEventsJCEC SignificantEventsJC     SigEventsJCSample1HigherInclusion       SigEventsJCSample2HigherI
nclusion        SignificantEventsJCEC   SigEventsJCECSample1HigherInclusion     SigEventsJCECSample2HigherInclusion
SE      44018   44293   240     127     113     258     140     118
A5SS    2605    2605    37      16      21      41      18      23
A3SS    4474    4479    39      19      20      40      17      23
MXE     6231    6247    10      2       8       9       2       7
RI      3377    3390    47      20      27      75      31      44
```

7. AS result plots

7.1 Plots with rmats2sashimiplot

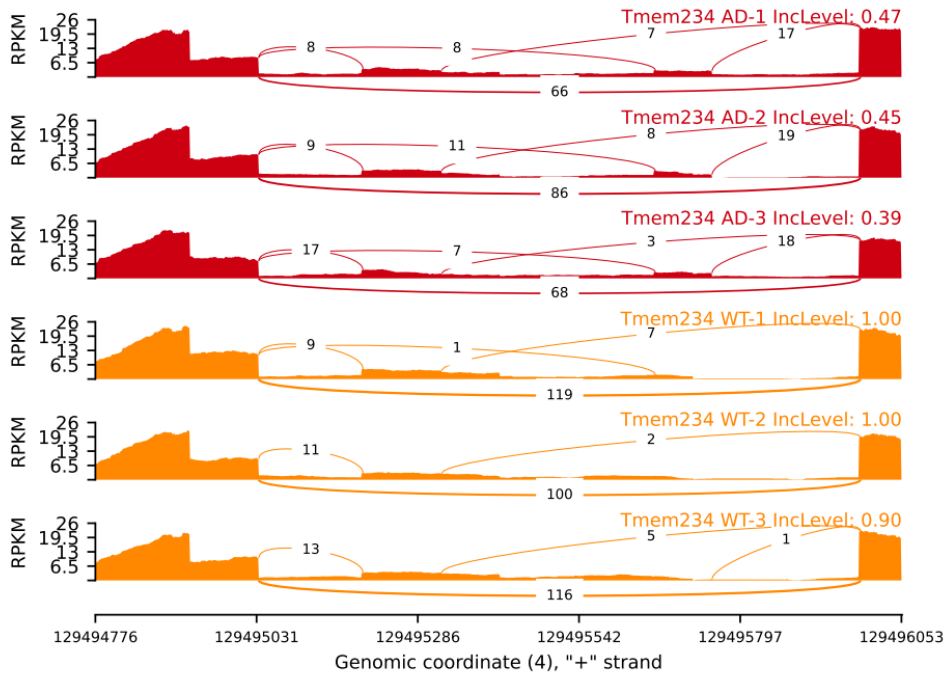
7.1.1 Plots of all samples

```
cd ~/project/AS/AD_GSE132177/splicing/rmats2sashimiplot
mkdir output

#example
#filter MXE events with FDR<0.05 and deltaPSI>0.1 to sig_MXE.txt
cat MXE.MATS.JC.txt | awk 'NR==1'>sig.MXE.txt
cat MXE.MATS.JC.txt | awk -F '\t' '{if($22<0.05 && $25>0.1)print$0}'>> sig.MXE.txt
cat MXE.MATS.JC.txt | awk -F '\t' '{if($22<0.05 && $25<(-0.1))print$0}'>> sig.MXE.txt

rmats2sashimiplot \
--b1 ../rmats2/AD_1.bam,../rmats2/AD_2.bam,../rmats2/AD_3.bam \
--b2 ../rmats2/WT_1.bam,../rmats2/WT_2.bam,../rmats2/WT_3.bam \
-t MXE -e ../rmats2/output/sig.MXE.txt \
--l1 AD --l2 WT \
-o output
```

```
cd ~/project/AS/AD_GSE132177/splicing/rmats2sashimiplot/output/Sashimi_plot
#download the pdf files.
```

Tmem234 for example.

:129494776:129495036:+@4:129495200:129495326:+@4:129495663:129495754:+@4:129495988:129496054:-



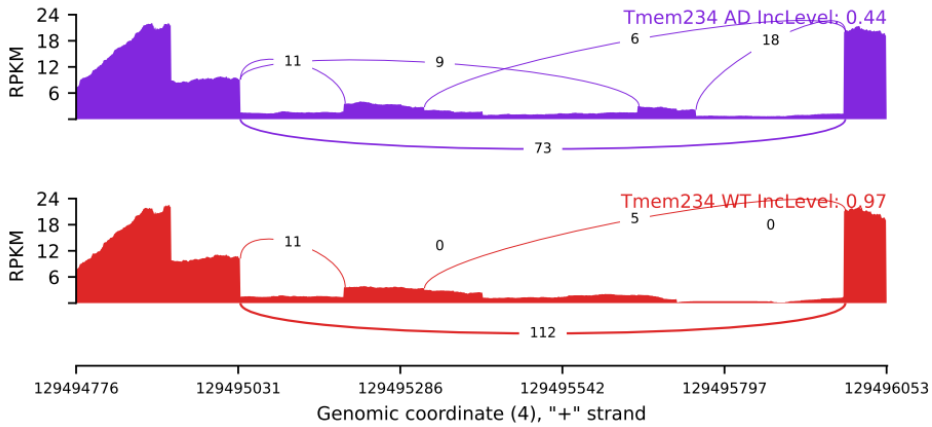
7.1.2 Plots by group

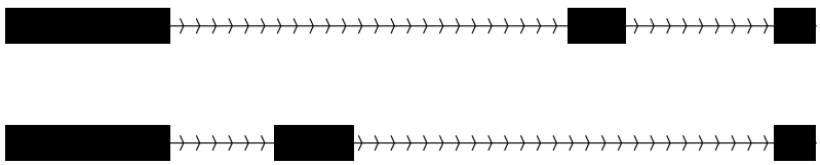
```
#conda activate rmats
cd ~/project/AS/AD_GSE132177/splicing/rmats2sashimiplot
mkdir group_output

#create group.gf
vim group.gf
###
AD: 1-3
WT: 4-6
###

#example
#filter MXE events with FDR<0.05 and deltaPSI>0.1 to sig_MXE.txt
cat MXE.MATS.JC.txt | awk 'NR==1'>sig.MXE.txt
cat MXE.MATS.JC.txt | awk -F '\t' '{if($22<0.05 && $25>0.1)print$0}'>> sig.MXE.txt
cat MXE.MATS.JC.txt | awk -F '\t' '{if($22<0.05 && $25<(-0.1))print$0}'>> sig.MXE.txt

rmats2sashimiplot \
```

```
--b1 ../rmats2/AD_1.bam,../rmats2/AD_2.bam,../rmats2/AD_3.bam \
--b2 ../rmats2/WT_1.bam,../rmats2/WT_2.bam,../rmats2/WT_3.bam \
-t MXE -e ../rmats2/output/sig.MXE.txt \
--l1 AD --l2 WT \
--group-info grouping.gf \
-o group_output
```

```
cd ~/project/AS/AD_GSE132177/splicing/rmats2sashimiplot/group_output/Sashimi_plot
```

:129494776:129495036:+@4:129495200:129495326:+@4:129495663:129495754:+@4:129495988:129496054:





8. AS statistics with maser

This step was done in Rstudio with maser package

8.1 AS statistics

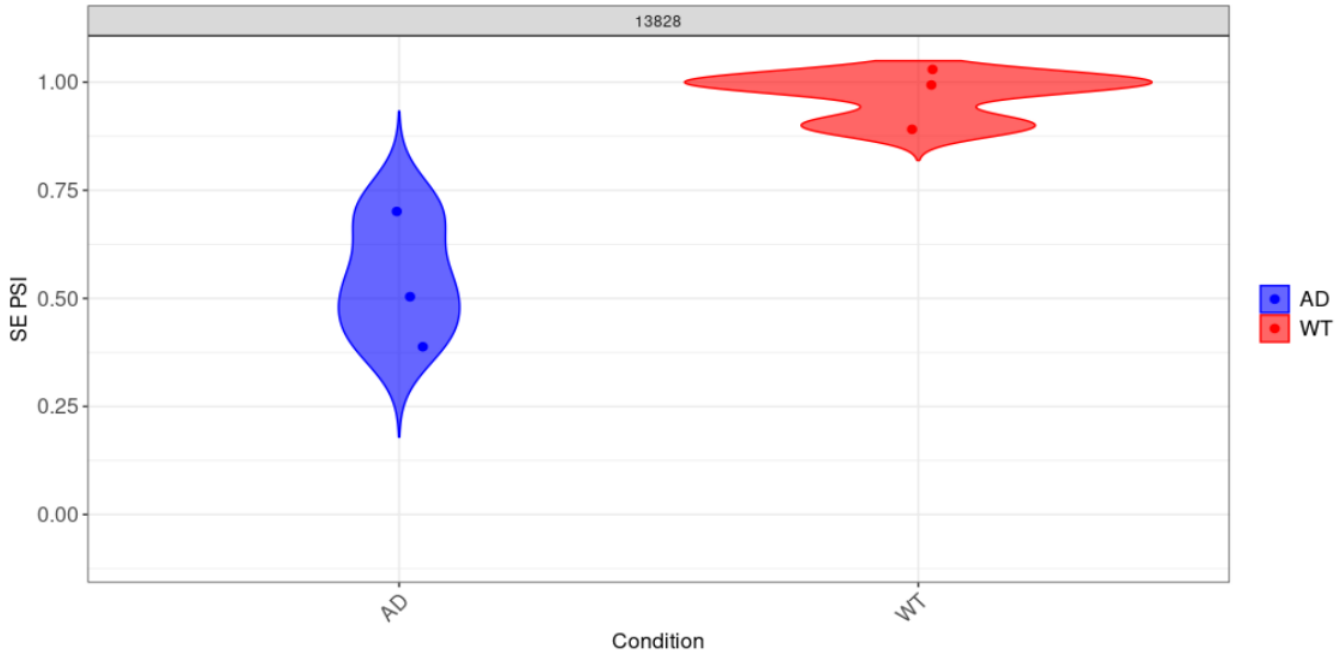
```
rm(list=ls())
if(!require("maser")) BiocManager::install("maser",update = F,ask = F)
if(!require("rtracklayer")) BiocManager::install("rtracklayer",update = F,ask = F)
library(maser)
library(rtracklayer)

##Step1: Importing rMATS events ##
#path to rMATS data
```

```
path <-("~/project/AS/AD_GSE132177/splicing/rmats2/output/")

AD <- maser(path, c("AD", "WT"), ftype = "JC")
AD
#A Maser object with 60705 splicing events.

#Samples description:
#Label=AD     n=3 replicates
#Label=WT     n=3 replicates

#Splicing events:
#A3SS.......... 4474 events
#A5SS.......... 2605 events
#SE.......... 44018 events
#RI.......... 3377 events
#MXE.......... 6231 events

#Access to different data types is done using annotation(), counts(), PSI(), and
summary(), which takes as argument the maser object and event type.
head(summary(AD, type = "SE")[, 1:8])
head(counts(AD, type = "SE"))
head(PSI(AD, type = "SE"))

##Step2: Filtering events ##
AD_filt <- filterByCoverage(AD, avg_reads = 10)
AD_filt
#A Maser object with 46569 splicing events.

#Samples description:
#Label=AD     n=3 replicates
#Label=WT     n=3 replicates

#Splicing events:
#A3SS.......... 2932 events
#A5SS.......... 1586 events
#SE.......... 34816 events
#RI.......... 2046 events
#MXE.......... 5189 events

#The function topEvents() allows to select statistically significant events given a FDR
cutoff and minimum PSI change. Default values are fdr = 0.05 and deltaPSI = 0.1 (ie. 10%
minimum change).
AD_top <- topEvents(AD_filt, fdr = 0.05, deltaPSI = 0.1)
AD_top
#A Maser object with 113 splicing events.

#Samples description:
#Label=AD     n=3 replicates
#Label=WT     n=3 replicates

#Splicing events:
#A3SS.......... 14 events
#A5SS.......... 11 events
```

```
#SE.......... 70 events
#RI.......... 15 events
#MXE.......... 3 events

save(AD, AD_filt, AD_top,file="AD.Rdata")

#Gene specific events can be selected using geneEvents(). For instance, there are 4
splicing changes affecting Tmem234 as seen below.
AD_Tmem234 <- geneEvents(AD_filt, geneS = "Tmem234", fdr = 0.05, deltaPSI = 0.1)
AD_Tmem234
#A Maser object with 3 splicing events.

#Samples description:
#Label=AD     n=3 replicates
#Label=WT     n=3 replicates

#Splicing events:
#A3SS.......... 0 events
#A5SS.......... 0 events
#SE.......... 1 events
#RI.......... 0 events
#MXE.......... 2 events

#Events in a maser object can be queried using an interactive data table provided by
display(). The table allows to look up event information such as gene names, identifiers
and PSI levels.
maser::display(AD_Tmem234, "SE")
```

ID	GeneID	geneSymbol	PValue	FDR	IncLevelDifference	PSI_1	PSI_2	Chr	Strand	exon_target	exon_upstream	exon_downstream
All	All	All	All	All	All	All	All	All	All	All	All	All
13828	ENSMUSG00000028797	Tmem234	4.23933680382e-8	0.0000982142775950257	-0.418	0.7,0.533,0.412	1,0.9,1	chr4	+	129495200-129495418	129494776-129494927	129495663-129495723

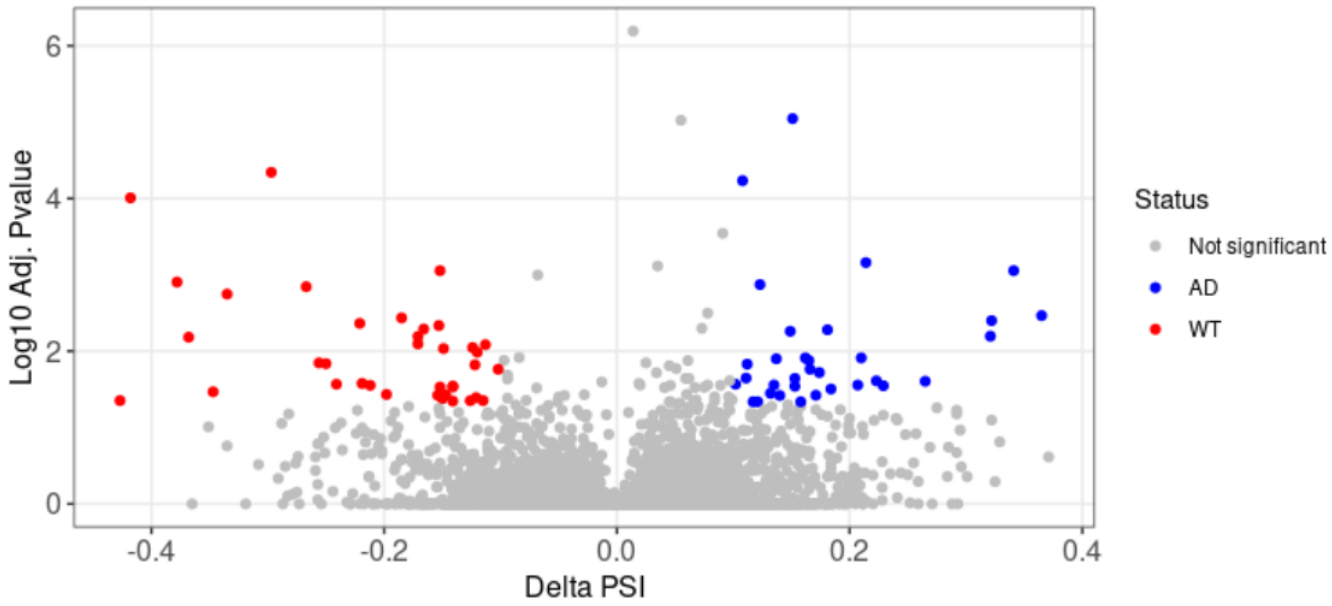
```
#PSI levels for gene events can be plotted using plotGenePSI(), indicating a valid
splicing type.
plotGenePSI(AD_Tmem234, type = "SE", show_replicates = T)
```



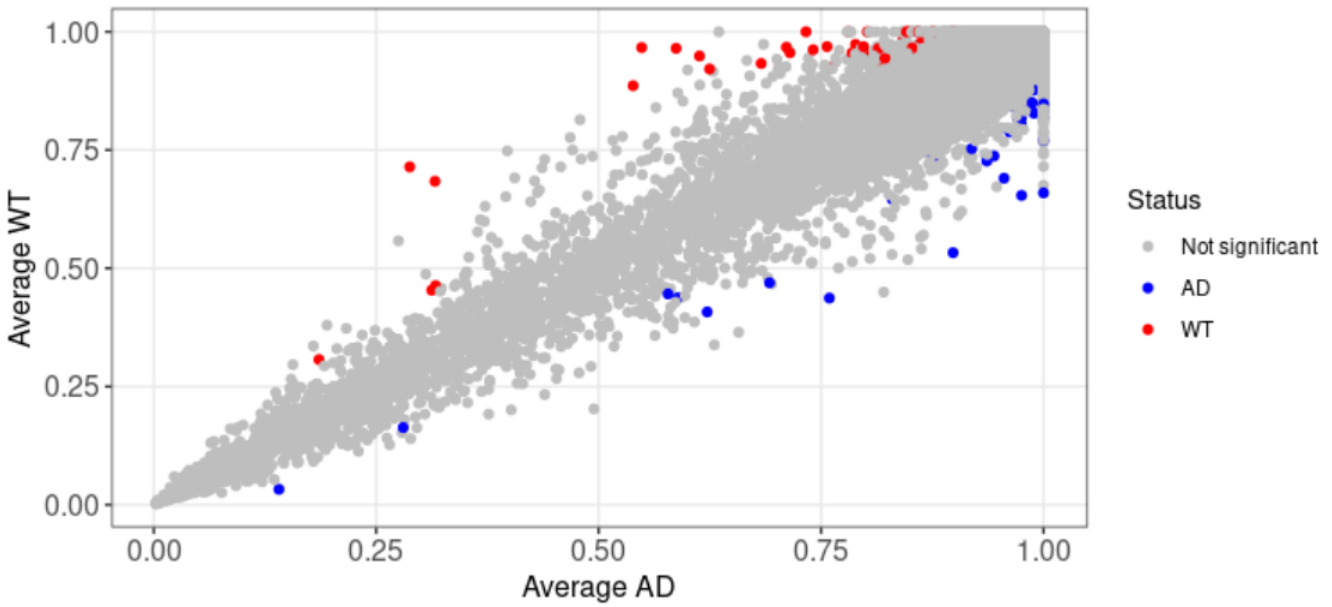
8.2 Global splicing plots

```
## Step3: Global splicing plots ##
#An overview of significant events can be obtained using either dotplot() or volcano()
functions, specifying FDR levels, minimum change in PSI between conditions and splicing
type. Significant events in each condition will be highlighted.
volcano(AD_filt, fdr = 0.05, deltaPSI = 0.1, type = "SE")
dotplot(AD_filt, fdr = 0.05, deltaPSI = 0.1, type = "SE")

#If only significant events should be plotted, then use topEvents() combined with
volcano() or dotplot() for visualization.
dotplot(AD_top, type = "SE")
volcano(AD_top, type = "SE")

#Splicing patterns of individual replicates can also be inspected using principal
component analysis (PCA) and plotted using pca(). Also, boxplots of PSI levels
distributions for all events in the maser object can be visualized using
boxplot_PSI_levels(). The breakdown of splicing types can be plotted using
splicingDistribution() and desired significance thresholds. Please refer to help pages
for examples on how to use these functions.

pca(AD_top)
boxplot_PSI_levels(AD_top,type = "RI")
splicingDistribution(AD_top)
```

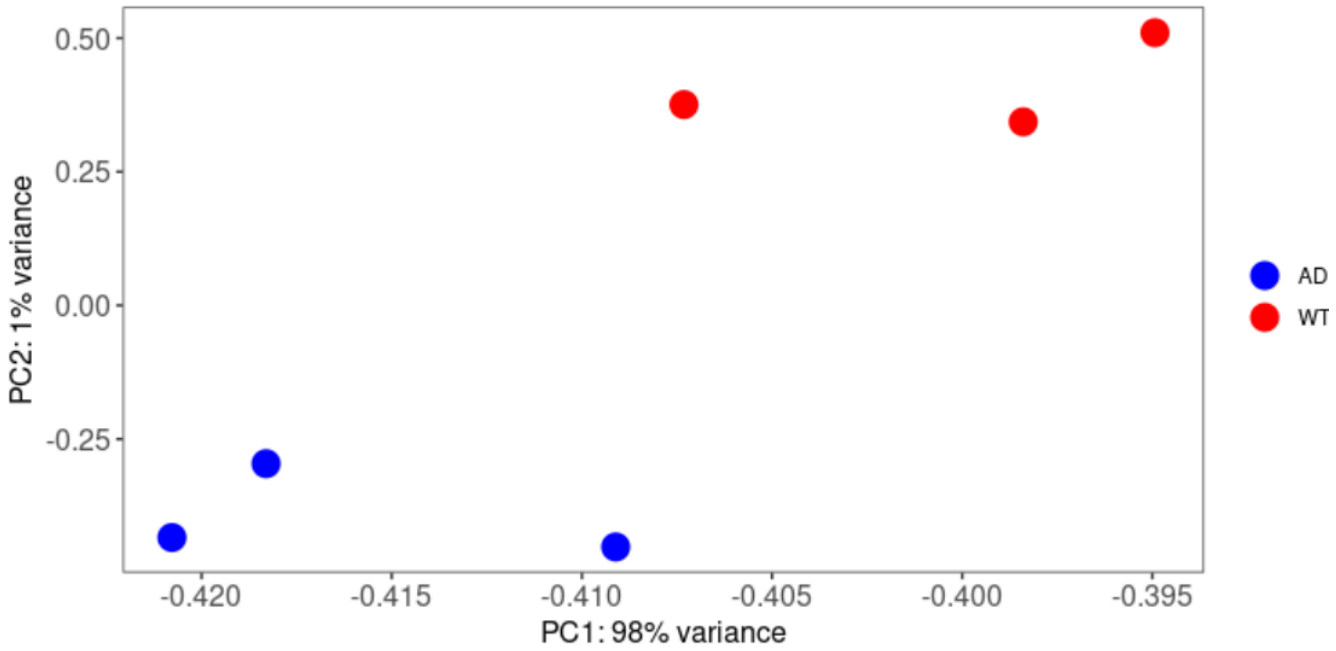
volcano(AD_filt, fdr = 0.05, deltaPSI = 0.1, type = "SE")



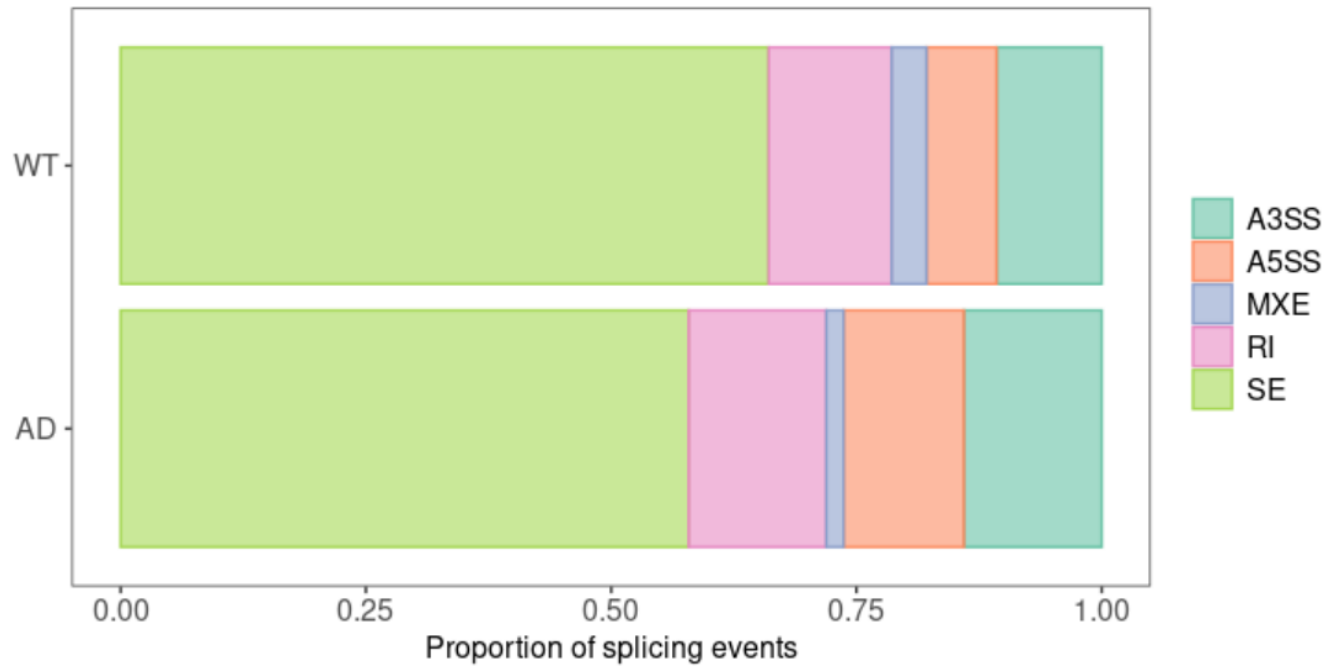
dotplot(AD_filt, fdr = 0.05, deltaPSI = 0.1, type = "SE")



pca(AD_top)



splicingDistribution(AD_top)



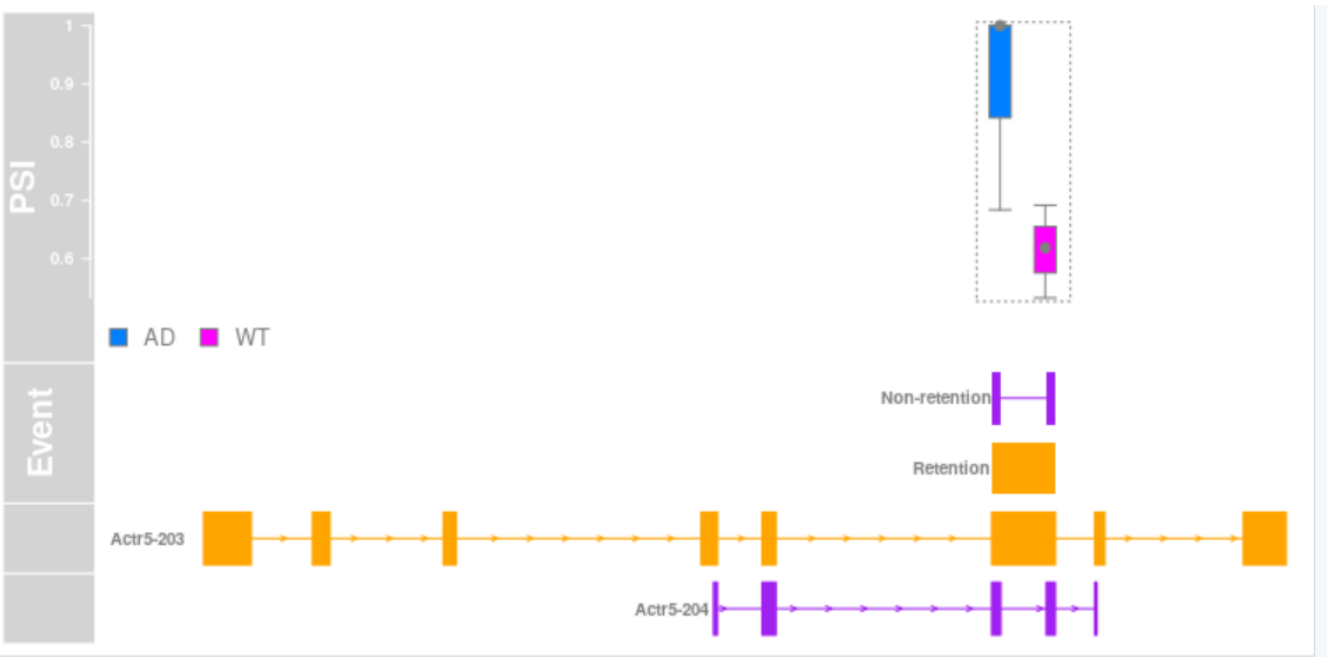
8.3 AS event plots

```
## Step4: Genomic visualization of splicing events ##
#plotTranscripts() requires an Ensembl or Gencode GTF using the hg38 build of the human
genome. Ensembl GTFs can be retrieved using AnnotationHub or imported using import.gff()
from the rtracklayer package. Several GTF releases are available, and maser is compatible
with any version using the hg38 build. We are going to use a reduced GTF extracted from
Ensembl Release 85 for running examples.

## Ensembl GTF annotation
gtf_path <-
("~/database/genome/Ensembl/Mus_musculus/GRCm39_release105/Mus_musculus.GRCm39.105.gtf")
```

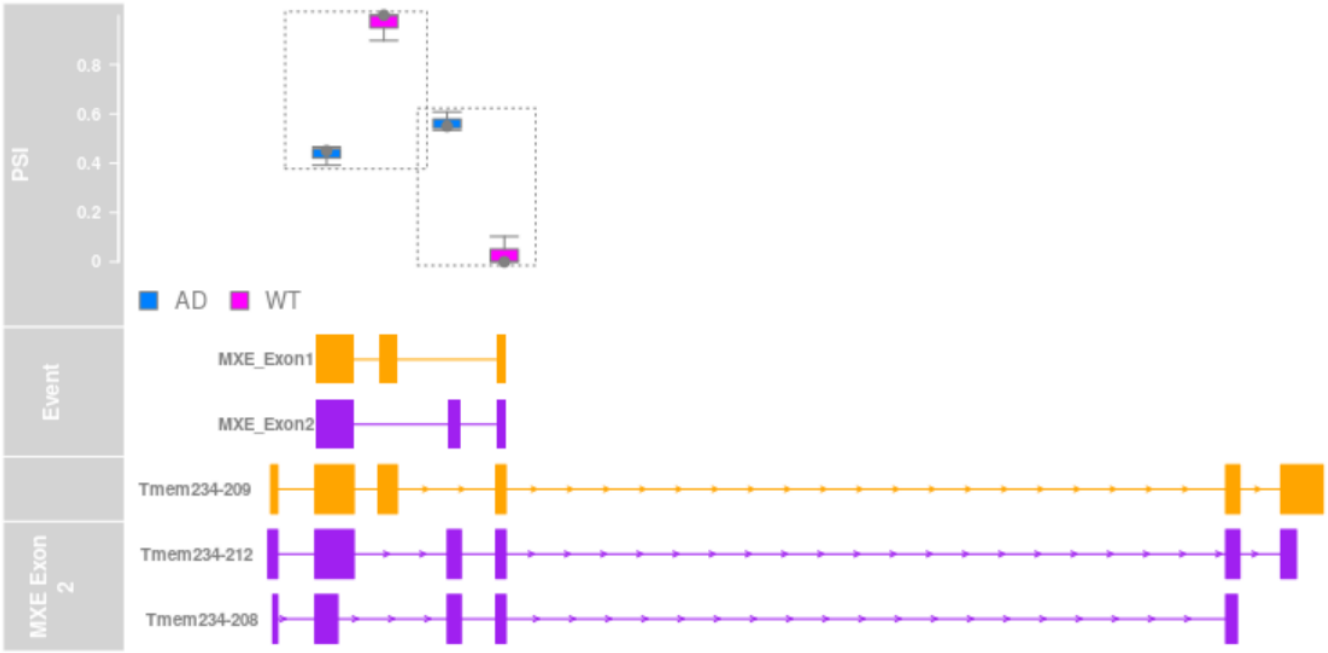
```
ens_gtf <- rtracklayer::import.gff(gtf_path)

# Step4.1: Exon skipping
## Retrieve Atr splicing events
Atr_events <- geneEvents(AD_filt, geneS = "Atr", fdr = 0.05,
                         deltaPSI = 0.1)

## Dislay affected transcripts and PSI levels
maser::display(Atr_events, "SE")
plotTranscripts(Atr_events, type = "SE", event_id = 36696,
                gtf = ens_gtf, zoom = FALSE, show_PSI = TRUE)

# Step4.2: Intron retention
Actr5_events <- geneEvents(AD_filt, geneS = "Actr5", fdr = 0.05, deltaPSI = 0.1 )
maser::display(Actr5_events, "RI")
plotTranscripts(Actr5_events, type = "RI", event_id = 2051,
                gtf = ens_gtf, zoom = FALSE)

# Step4.3: Mutually exclusive exons
#Tracks will display transcripts harboring the first or second mutually exclusive exons,
as well as both flanking exons.

#The PSI track in the mutually exclusive exons event will show two sets of boxplots. The
first set refers to Exon 1 PSI levels while the second set refers to Exon 2 PSI levels in
the two conditions. Therefore, the example below denotes increased Exon 2 PSI in AD.
Tmem234_events <- geneEvents(AD_filt, geneS = "Tmem234", fdr = 0.05, deltaPSI = 0.1 )
maser::display(Tmem234_events, "MXE")
plotTranscripts(Tmem234_events, type = "MXE", event_id = 1647,
                gtf = ens_gtf, zoom = FALSE)

# Stepp4.4: Alternative 5' and 3' exons
#In these type of events, the PSI track indicates inclusion levels for the longest exon.
It might be useful for visualization to set zoom = TRUE when the alternative splicing
generates exons of similar sizes.

Tmem138_gene <- geneEvents(AD_filt, geneS = "Tmem138", fdr = 0.05, deltaPSI = 0.1 )
maser::display(Tmem138_gene, "A3SS")
plotTranscripts(Tmem138_gene, type = "A3SS", event_id = 5943,
                gtf = ens_gtf, zoom = TRUE)

Sema6c_gene <- geneEvents(AD_filt, geneS = "Sema6c", fdr = 0.05, deltaPSI = 0.1 )
maser::display(Sema6c_gene, "A5SS")
plotTranscripts(Sema6c_gene, type = "A5SS", event_id = 3516,
                gtf = ens_gtf, zoom = TRUE)
```

Atr 'SE' event

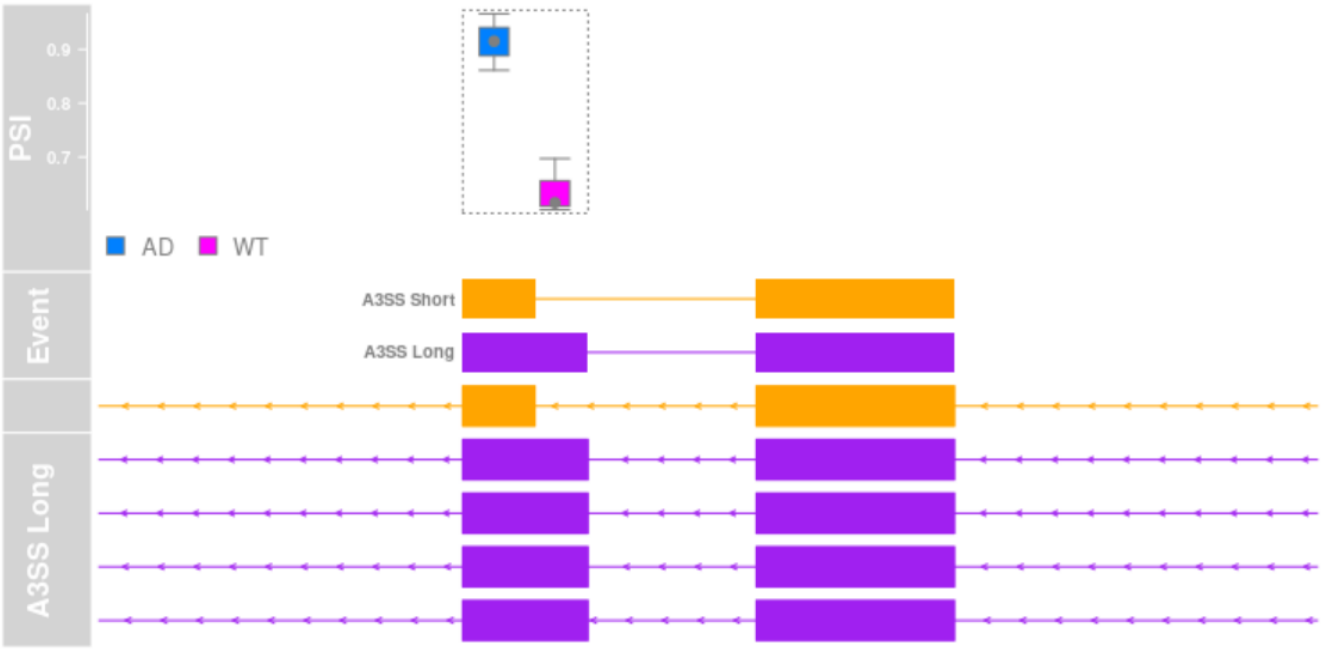
Actr5 'RI' event



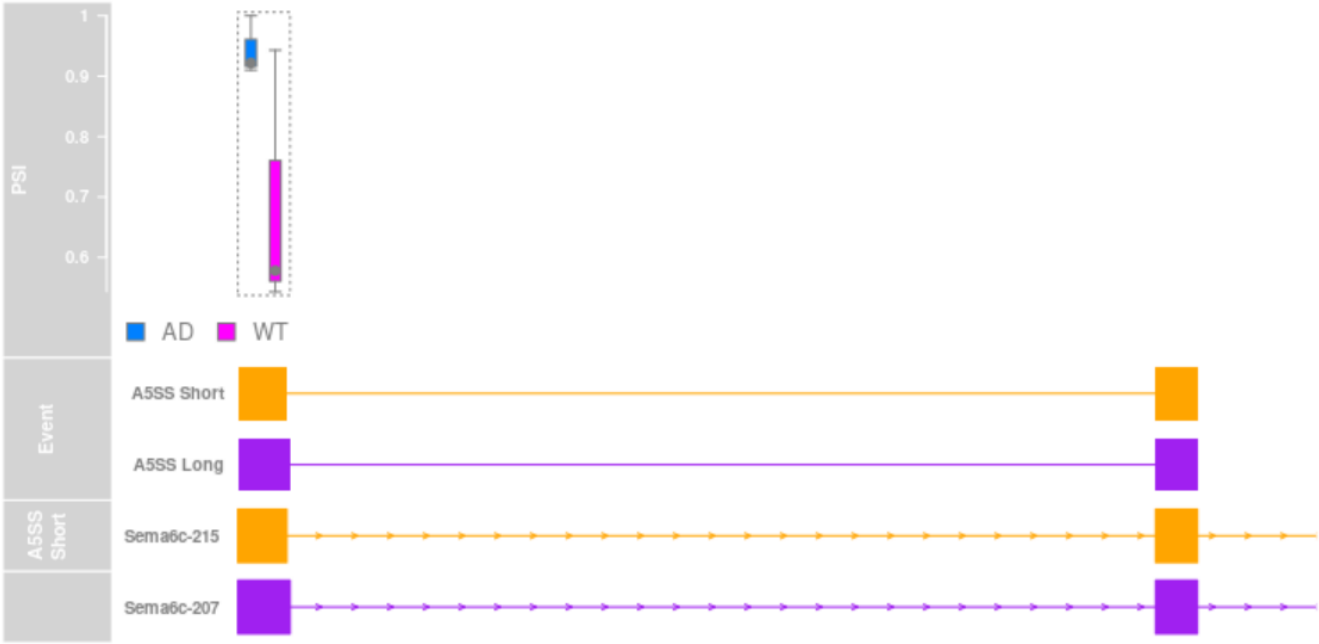
Tmem234 'MXE' event



Tmem138 'A3SS' event



Sema6c 'A5SS' event



That's the end.

> **Xiong Wang** Department of Laboratory Medicine, Tongji Hospital, Tongji Medical College, Huazhong University of Science and Technology, Wuhan 430030, China. Email: wangxiong@tjh.tjmu.edu.cn

April 21, 2022