

# Package: sRNAGenetic (via r-universe)

August 23, 2024

**Title** Analysis of sRNA Expression Changes During Plant Polyploidization

**Version** 0.1.0

**Description** The most important function of the R package sRNAGenetic is the genetic effects analysis of miRNA after plant polyploidization via two methods, and at the same time, it provides various forms of graph related to data characteristics and expression analysis. In terms of two classification methods, one is the calculation of the additive (a) and dominant (d), the other is the evaluation of ELD (expression level domainance) by comparing the total expression of the miRNA in allotetraploids with the expression level in the parent species.

**License** GPL (>= 3)

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.1.2

**Imports** DESeq2, futile.logger, ggplot2, ggsci, plyr, VennDiagram

**Depends** R (>= 2.10)

**LazyData** true

**Repository** <https://w123yu.r-universe.dev>

**RemoteUrl** <https://github.com/w123yu/srnagenetic>

**RemoteRef** HEAD

**RemoteSha** 69197f2bc4c62b4234e9d0d108155b111d2aa4b7

## Contents

basepreplot . . . . .	2
Countfilter . . . . .	3
F1_miRNA_count . . . . .	4
F1_miRNA_rpm . . . . .	4

F1_sRNA_seq . . . . .	5
genetic . . . . .	6
Get12Bins . . . . .	6
GetDatable . . . . .	7
lenplot . . . . .	8
mirnapredata . . . . .	9
miVennData . . . . .	10
miVennPlot . . . . .	11
P1_miRNA_count . . . . .	12
P1_miRNA_rpm . . . . .	12
P1_sRNA_seq . . . . .	13
P2_miRNA_count . . . . .	14
P2_miRNA_rpm . . . . .	14
P2_sRNA_seq . . . . .	15
polyDESeq . . . . .	15
Rpmfilter . . . . .	16
srnapredata . . . . .	17
<b>Index</b>	<b>18</b>

---

basepreplot	<i>Generate the base frequency plot of miRNA</i>
-------------	--

---

## Description

Generate the base frequency plot of miRNA

## Usage

```
basepreplot(file_dataframe, width = 0.6, size = 12)
```

## Arguments

`file_dataframe` A dataframe. The output result after running `mirnapredata`.

`width` A numeric. The width of the output bar plot, and default is 0.6.

`size` A numeric. The size of axis text, and default is 0.6.

## Value

The miRNA base frequency plot

**Examples**

```
##P1
P1_miRNA_data <- mirnapreddata(mirnaseq_dataframe = P1_miRNA_count)
##P2
P2_miRNA_data <- mirnapreddata(mirnaseq_dataframe = P2_miRNA_count)
##F1
F1_miRNA_data <- mirnapreddata(mirnaseq_dataframe = F1_miRNA_count)
##Drawing
basepreplot(file_dataframe = P1_miRNA_data)
basepreplot(file_dataframe = P2_miRNA_data)
basepreplot(file_dataframe = F1_miRNA_data)
```

Countfilter

*Filtering low expressed miRNAs based on count: Countfilter***Description**

Filtering low expressed miRNAs based on count: Countfilter

**Usage**

```
Countfilter(P1_count, P2_count, F1_count, count_threshold = 5)
```

**Arguments**

P1_count	A dataframe. The count data of miRNA from the P1 species. The first column must be the miRNA sequence. Others are listed as the count of miRNA, and each column denotes one biological replicate of the sample.
P2_count	A dataframe. Similar with P1_count, the count data of miRNA from the P2 species.
F1_count	A dataframe. Similar with P1_count, the count data of miRNA from the F1 species.
count_threshold	A numeric. In all samples, there is at least one sample whose count value is more than or equal to count_threshold to be retained. By default, the count value more than or equal to 5 is retained.

**Value**

A dataframe. The result includes all miRNAs that fulfill the count value requirement (count >= count\_threshold) in at least one sample.

**Examples**

```
##Get the filtered mirna count table (default: Count >= 5 in at least one sample)
Count5result <- Countfilter(P1_count = P1_miRNA_count,
                             P2_count = P2_miRNA_count,
                             F1_count = F1_miRNA_count, count_threshold = 5)
```

---

F1_miRNA_count	<i>Sequences of miRNAs from one species</i>
----------------	---

---

**Description**

The first column of this data set is all miRNA sequences from "F1", and other columns are the corresponding count values from different samples

**Format**

A dataframe containing all miRNAs with count value in "F1" (the S3 generations of Parents (Maternal parent: *Triticum turgidum*; Male parent: *Aegilops tauschii*)).

**Source**

Generated from the S3 generations of *Triticum turgidum* (AABB) and *Aegilops tauschii* (DD).

**References**

Li, A., et al., mRNA and Small RNA Transcriptomes Reveal Insights into Dynamic Homoeolog Regulation of Allopolyploid Heterosis in Nascent Hexaploid Wheat. *Plant Cell*, 2014. 26(5): p. 1878-1900.

**Examples**

```
data(F1_miRNA_count)
```

---

F1_miRNA_rpm	<i>Sequences of miRNAs from one species</i>
--------------	---

---

**Description**

The first column of this data set is all miRNA sequences from "F1", and other columns are the corresponding RPM values of different samples

**Format**

A dataframe containing all miRNAs with RPM value in "F1" (the S3 generations of Parents (Maternal parent: *Triticum turgidum*; Male parent: *Aegilops tauschii*)).

**Source**

Generated from the S3 generations of *Triticum turgidum* (AABB) and *Aegilops tauschii* (DD).

**References**

Li, A., et al., mRNA and Small RNA Transcriptomes Reveal Insights into Dynamic Homoeolog Regulation of Allopolyploid Heterosis in Nascent Hexaploid Wheat. *Plant Cell*, 2014. 26(5): p. 1878-1900.

**Examples**

```
data(F1_miRNA_rpm)
```

---

F1\_sRNA\_seq

*Sequences of sRNAs from one species*

---

**Description**

The first column of this data set is all sequences from one speceie for the data statistics

**Format**

A dataframe containing numerous sequences of all sRNAs in "F1" (the S3 generations of Parents (Maternal parent: *Triticum turgidum*; Male parent: *Aegilops tauschii*)). However, only 400 sRNAs are selected as test data due to the large data of sRNA.

**Details**

This data.frame is very useful for the functional demonstration of "srapredata"

**Source**

Generated from the S3 generations of *Triticum turgidum* (AABB) and *Aegilops tauschii* (DD).

**References**

Li, A., et al., mRNA and Small RNA Transcriptomes Reveal Insights into Dynamic Homoeolog Regulation of Allopolyploid Heterosis in Nascent Hexaploid Wheat. *Plant Cell*, 2014. 26(5): p. 1878-1900.

**Examples**

```
data(F1_sRNA_seq)
```

---

 genetic

*Classification for 12 expression patterns*


---

### Description

The input data is generated from the analysis result of DESeq2.

### Usage

```
genetic(pv11, pv12, pv21, fc11, fc12, fc21, Pvalue)
```

### Arguments

pv11	A numeric. The P value of F1_vs_P1 (Treatment:F1; Control:P1).
pv12	A numeric. The P value of F1_vs_P2 (Treatment:F1; Control:P2)
pv21	A numeric. The P value of P2_vs_P1 (Treatment:P2; Control:P1)
fc11	A numeric. The Log2(FoldChange) value of F1_vs_P1 (Treatment:F1; Control:P1)
fc12	A numeric. The Log2(FoldChange) value of F1_vs_P2 (Treatment:F1; Control:P2)
fc21	A numeric. The Log2(FoldChange) value of P2_vs_P1 (Treatment:P2; Control:P1)
Pvalue	A numeric. Filtration criteria of P value for Classification.

### Value

A dataframe.

---

 Get12Bins

*Genetic effects analysis: Twelve bins of expression analysis (method2)*


---

### Description

Genetic effects analysis: Twelve bins of expression analysis (method2)

### Usage

```
Get12Bins(P1_count, P2_count, F1_count, count_threshold = 5, Pvalue = 0.05)
```

**Arguments**

P1_count	A dataframe. The count data of miRNA from the P1 species. The first column must be the miRNA sequence. Others are listed as the count of miRNA, and each column denotes one biological replicate of the sample.
P2_count	A dataframe. Similar with P1_count, the count data of miRNA from the P2 species.
F1_count	A dataframe. Similar with P1_count, the count data of miRNA from the F1 species.
count_threshold	A numeric. In all samples, there is at least one sample whose count value is more than or equal to count_threshold to be retained. By default, the count value more than or equal to 5 is retained.
Pvalue	A numeric. The threshold of significance test among different groups. Default is 0.05.

**Value**

A dataframe. The output results contain the P value, log2FoldChange and grouping information for each miRNA expressed in all species (count  $\geq$  count\_threshold). F1\_vs\_P1(P value: pv11,log2FoldChange: fc11), F1\_vs\_P2(P value: pv12,log2FoldChange: fc12), P2\_vs\_P1(P value: pv21,log2FoldChange: fc21)

**Examples**

```
##Get the table of 12 expression patterns
Binresult <- Get12Bins(P1_count = P1_miRNA_count,
                      P2_count = P2_miRNA_count,
                      F1_count = F1_miRNA_count,
                      count_threshold = 5,Pvalue = 0.05)
```

---

 GetDatable

*Genetic effects analysis of miRNA: |d/a| (method 1)*


---

**Description**

The additive (a) and dominant (d) values were calculated by the expression level of each miRNA. Edwards et al. proposed that the " $|d/a|$ " can be used as the criterion to estimate the expression patterns of miRNAs. Specific classification criteria are as follows,  $|d/a| \leq 0.2$ , additivity;  $|d/a| > 0.2$  and  $|d/a| \leq 0.8$ , partial dominance;  $|d/a| > 0.8$  and  $|d/a| \leq 1.2$ , dominance;  $|d/a| > 1.2$ , overdominance.

**Usage**

```
GetDatable(P1_RPM, P2_RPM, F1_RPM, rpm_threshold = 1)
```

**Arguments**

P1_RPM	A dataframe. The rpm data of miRNA from the P1 species. The first column must be the miRNA sequence. Others are listed as the rpm of miRNA, and each column denotes one biological replicate of the sample.
P2_RPM	A dataframe. Similar with P1_RPM, the rpm data of miRNA from the P2 species.
F1_RPM	A dataframe. Similar with P1_RPM, the rpm data of miRNA from the F1 species.
rpm_threshold	A numeric. the average of rpm value among all the biological replicates. By default, the average of rpm more than or equal to 1 is retained.

**Value**

A dataframe. The output results contain the value of "ld/a|" and grouping results for each miRNA expressed in all species (average\_rpm >= rpm\_threshold).

**Examples**

```
##Get the classification results based on the value of |d/a|
DAresult <- GetDataable(P1_RPM = P1_miRNA_rpm,
                       P2_RPM = P2_miRNA_rpm,
                       F1_RPM = F1_miRNA_rpm, rpm_threshold = 1)
```

---

lenplot

*Generate the sRNA length distribution plot*


---

**Description**

Generate the sRNA length distribution plot

**Usage**

```
lenplot(file_dataframe, width = 0.6, size = 12)
```

**Arguments**

file_dataframe	A dataframe. The output result after running "srnapredata".
width	A numeric. The width of the output bar plot, and default is 0.6.
size	A numeric. The size of text in the output plot, and default is 12.

**Value**

The sRNA length distribution plot

## Examples

```
##F1
F1_sRNA <- srnapredata(srnaseq_dataframe = F1_sRNA_seq, group = "F1")
##P1
P1_sRNA <- srnapredata(srnaseq_dataframe = P1_sRNA_seq, group = "P1")
##P2
P2_sRNA <- srnapredata(srnaseq_dataframe = P2_sRNA_seq, group = "P2")
##integrate all sRNA data from P1, P2, and F1
sRNA_data <- rbind(F1_sRNA,P1_sRNA,P2_sRNA)
##plot
lenplot(file_dataframe = sRNA_data)
```

---

mirnapredata

*Generate the data of miRNA base frequency in each position*

---

## Description

Generally, the "T" base account for the highest percentage of miRNA in the first position. The function of "mirnapredata" can provide the input data for the next drawing of miRNA base distribution in each position.

## Usage

```
mirnapredata(mirnaseq_dataframe)
```

## Arguments

```
mirnaseq_dataframe
```

A dataframe. The first column must be the sRNA sequence.

## Value

A dataframe. About the output results, the first column is the base, the second column is the base frequency, the third column is the position.

## Examples

```
##P1
P1_miRNA_data <- mirnapredata(mirnaseq_dataframe = P1_miRNA_count)
##P2
P2_miRNA_data <- mirnapredata(mirnaseq_dataframe = P2_miRNA_count)
##F1
F1_miRNA_data <- mirnapredata(mirnaseq_dataframe = F1_miRNA_count)
```

---

miVennData

*Species specific expression analysis: miVennData*


---

### Description

miVennData: Extract the species-specific miRNAs and the shared miRNAs among parents and offspring.

### Usage

```
miVennData(
  P1_RPM,
  P2_RPM,
  F1_RPM,
  rpm_threshold = 1,
  output_file = "venn_list"
)
```

### Arguments

P1_RPM	A dataframe. The rpm data of miRNA from the P1 species. The first column must be the miRNA sequence. Others are listed as the rpm of miRNA, and each column denotes one biological replicate of the sample.
P2_RPM	A dataframe. Similar with P1_RPM, the rpm data of miRNA from the P2 species.
F1_RPM	A dataframe. Similar with P1_RPM, the rpm data of miRNA from the F1 species.
rpm_threshold	A numeric. the average of rpm value among all the biological replicates. By default, the average of rpm more than or equal to 1 is retained.
output_file	Specify the output file. "venn_list" is the default option, which outputs all the information of the Venn diagram. "all_common" is one of options, which outputs the miRNAs shared by parents and offspring. "P1_specific" is one of options, which outputs P1 specific expression miRNA. "P2_specific" is one of options, which outputs P2 specific expression miRNA. "F1_specific" is one of options, which outputs F1 specific expression miRNA.

### Value

A dataframe. The output results is based on your selection (output\_file).

### Examples

```
##Extract the species-specific miRNAs and the shared miRNAs among parents and offspring.
##output_file = "venn_list"
venn_list <- miVennData(P1_RPM = P1_miRNA_rpm,
                       P2_RPM = P2_miRNA_rpm,
```

```

        F1_RPM = F1_miRNA_rpm,
        rpm_threshold = 1,output_file = "venn_list")
##output_file = "P1_specific"
P1_specific <- miVennData(P1_RPM = P1_miRNA_rpm,
                        P2_RPM = P2_miRNA_rpm,
                        F1_RPM = F1_miRNA_rpm,
                        rpm_threshold = 1,output_file = "P1_specific")
##output_file = "P2_specific"
P2_specific <- miVennData(P1_RPM = P1_miRNA_rpm,
                        P2_RPM = P2_miRNA_rpm,
                        F1_RPM = F1_miRNA_rpm,
                        rpm_threshold = 1,output_file = "P2_specific")
##output_file = "F1_specific"
F1_specific <- miVennData(P1_RPM = P1_miRNA_rpm,
                        P2_RPM = P2_miRNA_rpm,
                        F1_RPM = F1_miRNA_rpm,
                        rpm_threshold = 1,output_file = "F1_specific")
##output_file = "all_common"
all_common <- miVennData(P1_RPM = P1_miRNA_rpm,
                        P2_RPM = P2_miRNA_rpm,
                        F1_RPM = F1_miRNA_rpm,
                        rpm_threshold = 1,output_file = "all_common")

```

---

miVennPlot

*Species specific expression analysis: miVennPlot*


---

## Description

miVennPlot: generate the Venn diagram with the specific expression information of miRNAs.

## Usage

```
miVennPlot(P1_RPM, P2_RPM, F1_RPM, rpm_threshold = 1)
```

## Arguments

P1_RPM	A dataframe. The rpm data of miRNA from the P1 species. The first column must be the miRNA sequence. Others are listed as the rpm of miRNA, and each column denotes one biological replicate of the sample.
P2_RPM	A dataframe. Similar with P1_RPM, the rpm data of miRNA from the P2 species.
F1_RPM	A dataframe. Similar with P1_RPM, the rpm data of miRNA from the F1 species.
rpm_threshold	A numeric. the average of rpm value among all the biological replicates. By default, the average of rpm more than or equal to 1 is retained.

## Value

The Venn diagram with the specific expression information of miRNAs.

**Examples**

```
##Drawing
miVennPlot(P1_RPM = P1_miRNA_rpm,
           P2_RPM = P2_miRNA_rpm,
           F1_RPM = F1_miRNA_rpm, rpm_threshold = 1)
```

---

P1_miRNA_count	<i>Sequences of miRNAs from one species</i>
----------------	---

---

**Description**

The first column of this data set is all miRNA sequences from "P1", and other columns are the corresponding count values from different samples

**Format**

A dataframe containing all miRNAs with count value in "P1" (Male parent: *Aegilops tauschii*).

**Source**

Generated from the *Aegilops tauschii* (DD).

**References**

Li, A., et al., mRNA and Small RNA Transcriptomes Reveal Insights into Dynamic Homoeolog Regulation of Allopolyploid Heterosis in Nascent Hexaploid Wheat. *Plant Cell*, 2014. 26(5): p. 1878-1900.

**Examples**

```
data(P1_miRNA_count)
```

---

P1_miRNA_rpm	<i>Sequences of miRNAs from one species</i>
--------------	---

---

**Description**

The first column of this data set is all miRNA sequences from "P1", and other columns are the corresponding RPM values of different samples

**Format**

A dataframe containing all miRNAs with RPM value in "P1" (Male parent: *Aegilops tauschii*).

**Source**

Generated from the *Aegilops tauschii* (DD).

**References**

Li, A., et al., mRNA and Small RNA Transcriptomes Reveal Insights into Dynamic Homoeolog Regulation of Allopolyploid Heterosis in Nascent Hexaploid Wheat. *Plant Cell*, 2014. 26(5): p. 1878-1900.

**Examples**

```
data(P1_miRNA_rpm)
```

---

P1\_sRNA\_seq

*Sequences of sRNAs from one species*

---

**Description**

The first column of this data set is all sequences from one speceie for the data statistics

**Format**

A dataframe containing numerous sequences of all sRNAs in "P1" (Male parent: *Aegilops tauschii*). However, only 400 sRNAs are selected as test data due to the large data of sRNA.

**Details**

This data.frame is very useful for the functional demonstration of "srnapredata"

**Source**

Generated from the *Aegilops tauschii* (DD).

**References**

Li, A., et al., mRNA and Small RNA Transcriptomes Reveal Insights into Dynamic Homoeolog Regulation of Allopolyploid Heterosis in Nascent Hexaploid Wheat. *Plant Cell*, 2014. 26(5): p. 1878-1900.

**Examples**

```
data(P1_sRNA_seq)
```

---

P2_miRNA_count	<i>Sequences of miRNAs from one species</i>
----------------	---

---

**Description**

The first column of this data set is all miRNA sequences from "P2", and other columns are the corresponding count values from different samples

**Format**

A dataframe containing all miRNAs with count value in "P2" (Maternal parent: Triticum turgidum).

**Source**

Generated from the Triticum turgidum (AABB).

**References**

Li, A., et al., mRNA and Small RNA Transcriptomes Reveal Insights into Dynamic Homoeolog Regulation of Allopolyploid Heterosis in Nascent Hexaploid Wheat. *Plant Cell*, 2014. 26(5): p. 1878-1900.

**Examples**

```
data(P2_miRNA_count)
```

---

P2_miRNA_rpm	<i>Sequences of miRNAs from one species</i>
--------------	---

---

**Description**

The first column of this data set is all miRNA sequences from "P2", and other columns are the corresponding RPM values of different samples

**Format**

A dataframe containing all miRNAs with RPM value in "P2" (Maternal parent: Triticum turgidum).

**Source**

Generated from the Triticum turgidum (AABB).

**References**

Li, A., et al., mRNA and Small RNA Transcriptomes Reveal Insights into Dynamic Homoeolog Regulation of Allopolyploid Heterosis in Nascent Hexaploid Wheat. *Plant Cell*, 2014. 26(5): p. 1878-1900.

**Examples**

```
data(P2_miRNA_rpm)
```

---

P2\_sRNA\_seq

*Sequences of sRNAs from one species*

---

**Description**

The first column of this data set is all sequences from one species for the data statistics

**Format**

A dataframe containing numerous sequences of all sRNAs in "P2" (Maternal parent: Triticum turgidum). However, only 400 sRNAs are selected as test data due to the large data of sRNA.

**Details**

This data.frame is very useful for the functional demonstration of "srnapredata"

**Source**

Generated from the Triticum turgidum (AABB).

**References**

Li, A., et al., mRNA and Small RNA Transcriptomes Reveal Insights into Dynamic Homoeolog Regulation of Allopolyploid Heterosis in Nascent Hexaploid Wheat. *Plant Cell*, 2014. 26(5): p. 1878-1900.

**Examples**

```
data(P2_sRNA_seq)
```

---

polyDESeq

*Differential expression analysis*

---

**Description**

Differential expression analysis

**Usage**

```
polyDESeq(P1_count, P2_count, F1_count, count_threshold = 5, Pvalue = 0.05)
```

**Arguments**

P1_count	A dataframe. The count data of miRNA from the P1 species. The first column must be the miRNA sequence. Others are listed as the count of miRNA, and each column denotes one biological replicate of the sample.
P2_count	A dataframe. Similar with P1_count, the count data of miRNA from the P2 species.
F1_count	A dataframe. Similar with P1_count, the count data of miRNA from the F1 species.
count_threshold	A numeric. In all samples, there is at least one sample whose count value is more than or equal to count_threshold to be retained. By default, the count value more than or equal to 5 is retained.
Pvalue	A numeric. The threshold of significance test among different groups. Default is 0.05.

**Value**

A dataframe. Differential expression analysis results of miRNA expressed in each two species (count >= count\_threshold).

**Examples**

```
##Drawing
polyDESeq(P1_count = P1_miRNA_count,
          P2_count = P2_miRNA_count,
          F1_count = F1_miRNA_count,
          count_threshold = 5, Pvalue = 0.05)
```

---

Rpmfilter

*Filtering low expressed miRNAs based on RPM: Rpmfilter*


---

**Description**

Filtering low expressed miRNAs based on RPM: Rpmfilter

**Usage**

```
Rpmfilter(P1_RPM, P2_RPM, F1_RPM, rpm_threshold = 1)
```

**Arguments**

P1_RPM	A dataframe. The rpm data of miRNA from the P1 species. The first column must be the miRNA sequence. Others are listed as the rpm of miRNA, and each column denotes one biological replicate of the sample.
P2_RPM	A dataframe. Similar with P1_RPM, the rpm data of miRNA from the P2 species.

F1\_RPM            A dataframe. Similar with P1\_RPM, the rpm data of miRNA from the F1 species.

rpm\_threshold    A numeric. the average of rpm value among all the biological replicates. By default, the average of rpm more than or equal to 1 is retained.

**Value**

A dataframe. The result includes all miRNAs that fulfill the average rpm value requirement (Average rpm  $\geq$  rpm\_threshold) among all species.

**Examples**

```
##Get the filtered mirna rpm table (default: the average rpm  $\geq$  1 in three species)
Rpm1result <- Rpmfilter(P1_RPM = P1_miRNA_rpm,
                       P2_RPM = P2_miRNA_rpm,
                       F1_RPM = F1_miRNA_rpm, rpm_threshold = 1)
```

---

srnapredata                      *Generate the data of sRNA length distribution*

---

**Description**

Generally, the length interval of sRNA is 21-24. The function of "srnapredata" can provide the input data for the next drawing of sRNA length distribution among different species.

**Usage**

```
srnapredata(srnaseq_dataframe, group)
```

**Arguments**

srnaseq\_dataframe            A dataframe. The first column must be the sRNA sequence.

group                        A character. You can select a representative group name for next drawing.

**Value**

A dataframe. The output results consist of three columns, the first column is the length of sRNA, the second column is the frequency, and the third column is the group name.

**Examples**

```
##Only 400 sRNAs are selected as test data due to the large data of sRNA.
##Recommended to use the "data.table" package for reading data quickly.
##F1
F1_sRNA <- srnapredata(srnaseq_dataframe = F1_sRNA_seq, group = "F1")
##P1
P1_sRNA <- srnapredata(srnaseq_dataframe = P1_sRNA_seq, group = "P1")
##P2
P2_sRNA <- srnapredata(srnaseq_dataframe = P2_sRNA_seq, group = "P2")
```

# Index

[basepreplot](#), 2

[Countfilter](#), 3

[F1\\_miRNA\\_count](#), 4

[F1\\_miRNA\\_rpm](#), 4

[F1\\_sRNA\\_seq](#), 5

[genetic](#), 6

[Get12Bins](#), 6

[GetDatable](#), 7

[lenplot](#), 8

[mirnapredata](#), 9

[miVennData](#), 10

[miVennPlot](#), 11

[P1\\_miRNA\\_count](#), 12

[P1\\_miRNA\\_rpm](#), 12

[P1\\_sRNA\\_seq](#), 13

[P2\\_miRNA\\_count](#), 14

[P2\\_miRNA\\_rpm](#), 14

[P2\\_sRNA\\_seq](#), 15

[polyDESeq](#), 15

[Rpmfilter](#), 16

[srnapredata](#), 17