

User Manual for

CMPLOT

Circular Manhattan plot

Version 3.1.3

Designed by Lilin Yin

Huazhong agriculture university

Last updated on May 23, 2016

Getting Started

CMplot is a package that is run in the R software environment, which can be freely downloaded from <http://www.r-project.org> or <http://www.rstudio.com>.

```
install.packages("CMplot") #install the package
library("CMplot")
?CMplot #see the detail information of parameters in CMplot
# source the newest codes of CMplot(NOT necessary)
# source("https://raw.githubusercontent.com/YinLiLin/R-CMplot/master/CMplot.r")
```

Inputting Data

CMplot only needs one input file and output three types of plots. In the required file, each row represents a SNP and consists four columns at least, which are SNP ID, Chromosome ID, physical position, and p value(or SNP effect) of the first phenotype.

```
yourdata <- read.table("yourfile", head=T) #the names of columns are not needed when plotting
CMplot(yourdata)
#users can have a practice with supplied datasets in CMplot
data(pig60K) #import the pig60K dataset of CMplot into R memory
head(pig60K) #view the pig60K dataset
data(cattle50K) #import the cattle50K dataset of CMplot into R memory
head(cattle50K) #view the cattle50K dataset
```

Note that the “Chromosome” must contain numbers, character is allowed.

Example(pig60K) #Genome-wide association study results(**P-value**)

SNP	Chromosome	Position	trait1	trait2	trait3
ALGA0000009	1	52297	0.773819	0.511943	0.511943
ALGA0000014	1	79763	0.773819	0.511943	0.511943
ALGA0000021	1	209568	0.758302	0.984053	0.984053
ALGA0000022	1	292758	0.72003	0.488871	0.488871
ALGA0000046	1	747831	0.973684	0.220968	0.220968
ALGA0000047	1	761957	0.917457	0.057537	0.057537
ALGA0000087	1	1108844	0.670386	0.337274	0.337274
ALGA0000112	1	1276597	0.536422	0.404509	0.404509
ALGA0000120	1	1352912	0.84535	0.482148	0.482148
ALGA0000131	1	1461555	0.128359	0.048774	0.048774

Example(cattle50K) #Genomic selection results(**SNP effect**)

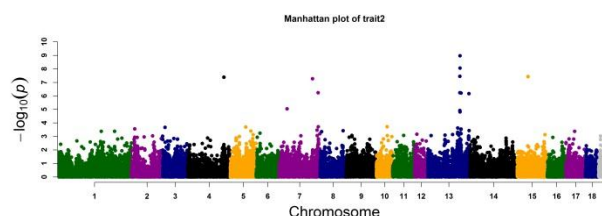
SNP	chr	pos	Somatic cell score	Milk yield	Fat percentage
SNP1	1	59082	0.000244361	0.000484	0.001379

SNP2	1	118164	0.000532272	3.98E-05	0.000599
SNP3	1	177246	0.001633058	0.000312	0.000279
SNP4	1	236328	0.001412865	0.000909	0.00104
SNP5	1	295410	9.07E-05	0.002203	0.000351
SNP6	1	354493	0.000110681	0.000343	0.000106
SNP7	1	413575	0.00036867	0.000649	0.001096
SNP8	1	472657	0.000406556	0.001404	0.000265
SNP9	1	531739	0.001154525	0.000906	0.000975
SNP10	1	590821	0.000270112	0.00053	0.001141

Part I Rectangular Manhattan Plot

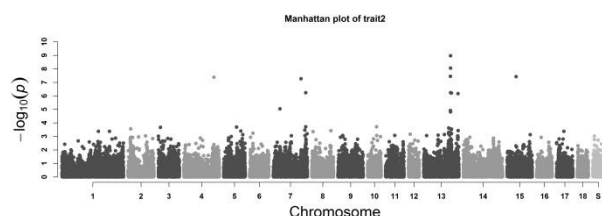
(1) Manhattan plot with p-values of GWAS results:

`CMplot(pig60K[, -c(4,6)], plot.type="m", band=0, LOG10=TRUE)`



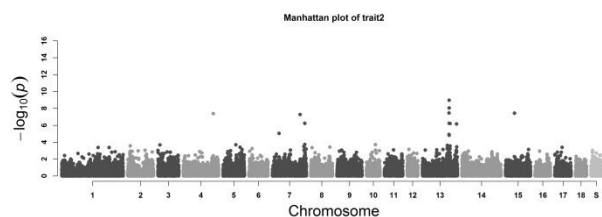
(2) Adjust the size of points and the colors and space between chromosomes:

`CMplot(pig60K[, -c(4,6)], plot.type="m", cex=1, band=1, col=c("grey30", "grey60"), LOG10=TRUE)`



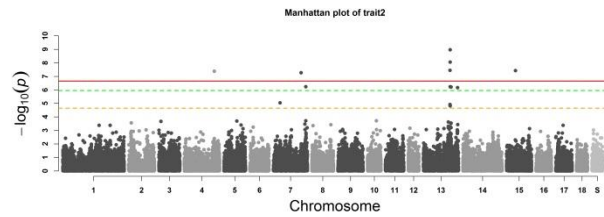
(3) Limit the range of Y-axis:

`CMplot(pig60K[, -c(4,6)], plot.type="m", band=1, col=c("grey30", "grey60"), ylim=c(0,16), LOG10=TRUE)`



(4) Add the significant line(or lines) and change the type, the color, the width of the line(lines):

`CMplot(pig60K[, -c(4,6)], plot.type="m", band=1, col=c("grey30", "grey60"), threshold=c(0.01,0.05,1), threshold.col=c("red", "green", "orange"), threshold.lty=c(1,2,2), threshold.lwd=c(2,2,2), amplify=FALSE, LOG10=TRUE)`



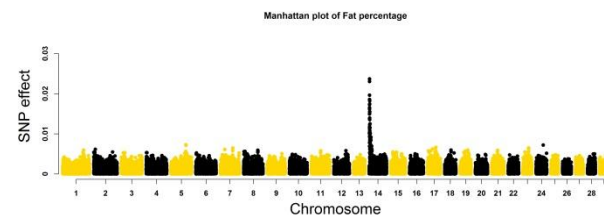
(5) Highlight the points which are significant:

`CMplot(pig60K[, -c(4,6)], plot.type="m", band=1, col=c("grey30", "grey60"), threshold=c(0.01, 0.05, 1), threshold.col=c("red", "green", "orange"), threshold.lty=c(1, 2, 2), threshold.lwd=c(2, 2, 2), amplify=T, signal.cex=1.5, signal.col="red", signal.pch=19, LOG10=TRUE)`



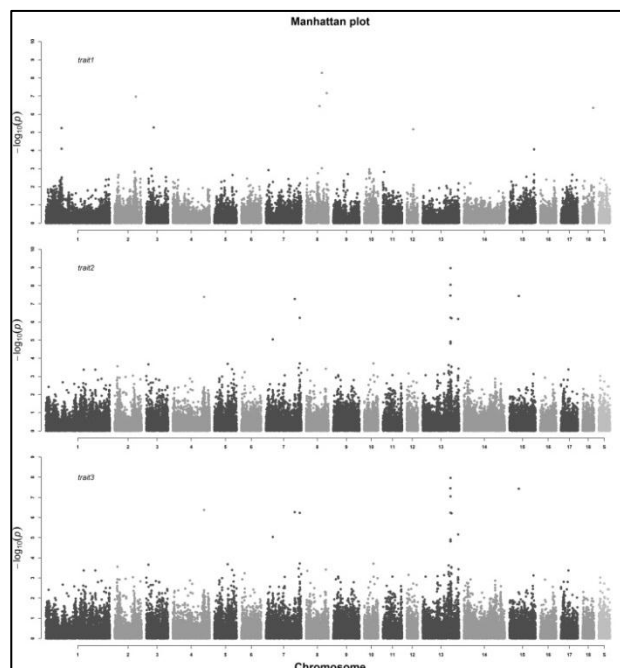
(6) Manhattan plot with SNP effects calculated by rrBLUP:

`CMplot(cattle50K[, -c(4,5)], plot.type="m", band=1, col=c("gold", "black"), LOG10=FALSE, ylab="SNP effect")`



(7) Plot multiple traits Manhattan in one track:

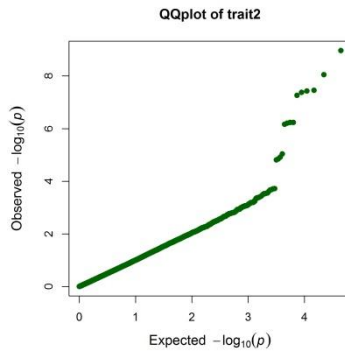
`CMplot(pig60K, plot.type="m", band=1, col=c("grey30", "grey60"), LOG10=TRUE, multitracks=TRUE)`



Part II Q-Q Plot

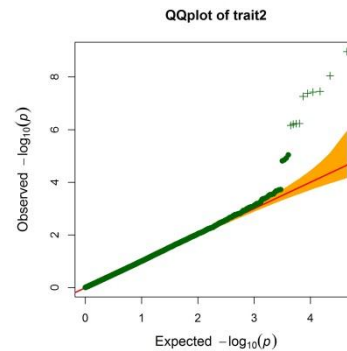
(1) Q-Q plot with p-values of GWAS results:

`CMplot(pig60K[, -c(4,6)], plot.type="q", col="darkgreen", LOG10=TRUE, threshold.col=NULL)`



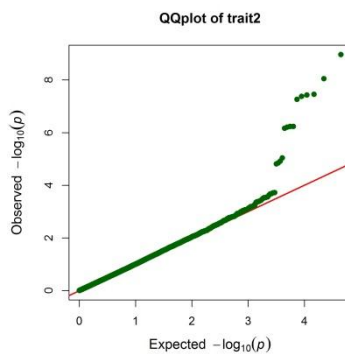
(4) Change the shape of significant points:

`CMplot(pig60K[, -c(4,6)], plot.type="q", col="darkgreen", LOG10=TRUE, threshold=0.05, threshold.col="red", signal.pch=3, conf.int=T, conf.int.col="orange")`



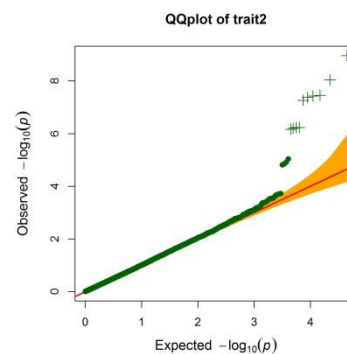
(2) Add the diagonal line on the plot:

`CMplot(pig60K[, -c(4,6)], plot.type="q", col="darkgreen", LOG10=TRUE, threshold.col="red")`



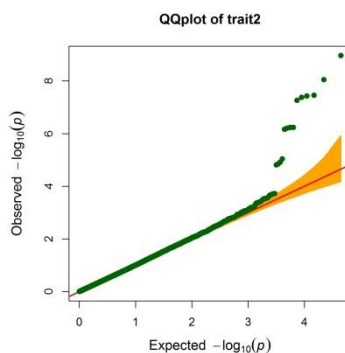
(5) Change the size of significant points:

`CMplot(pig60K[, -c(4,6)], plot.type="q", col="darkgreen", LOG10=TRUE, threshold=0.05, threshold.col="red", signal.pch=3, signal.cex=1.5, conf.int=T, conf.int.col="orange")`



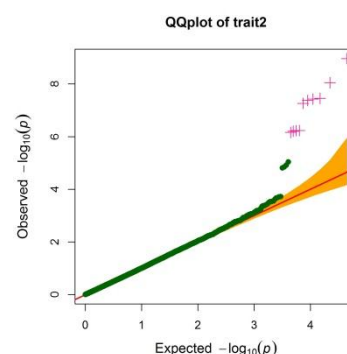
(3) Add confidence interval and change its color:

`CMplot(pig60K[, -c(4,6)], plot.type="q", col="blue", LOG10=TRUE, threshold.col="red", conf.int=TRUE, conf.int.col="orange")`



(6) Change the color of significant points:

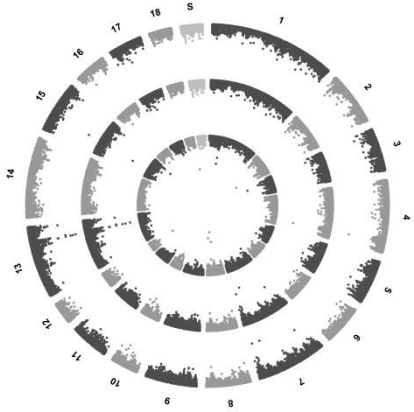
`CMplot(pig60K[, -c(4,6)], plot.type="q", col="darkgreen", threshold=0.05, threshold.col="red", signal.pch=3, signal.cex=1.5, signal.col="deeppink", conf.int=T, conf.int.col="orange")`



Part III Circular Manhattan Plot

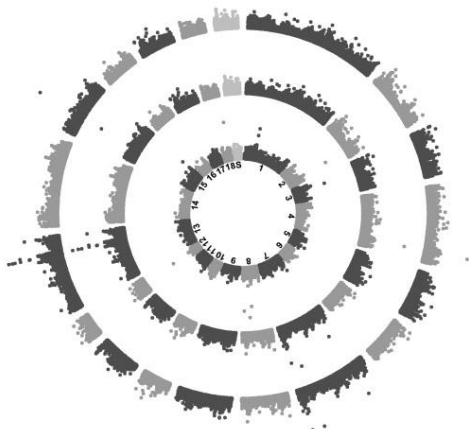
(1) Circular plot with p-values of GWAS results:

`CMplot(pig60K,plot.type="c",r=0.5,LOG10=TRUE,col=c("grey30","grey60",outward=FALSE))`



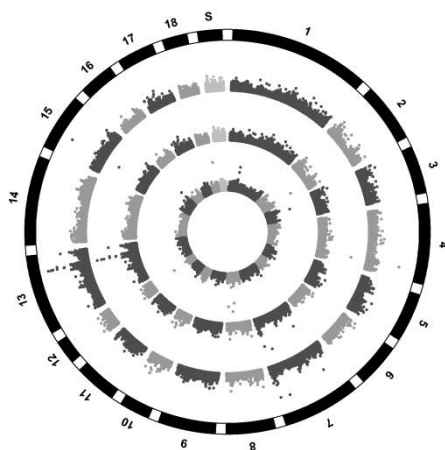
(2) Change the plot orientation:

`CMplot(pig60K,plot.type="c",r=1,LOG10=TRUE,col=c("grey30","grey60"),outward=TRUE)`



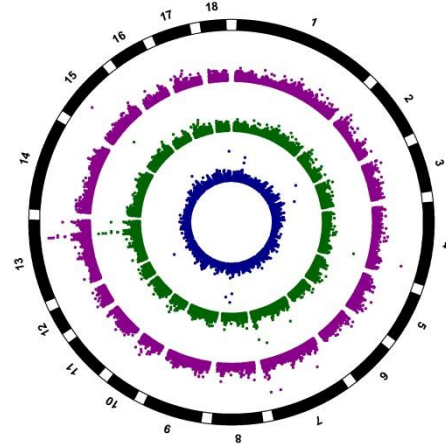
(3) Add the chromosome to outside:

`CMplot(pig60K,plot.type="c",r=1,LOG10=TRUE,col=c("grey30","grey60"),outward=TRUE,cir.chr=TRUE,cir.chr.h=1.3)`



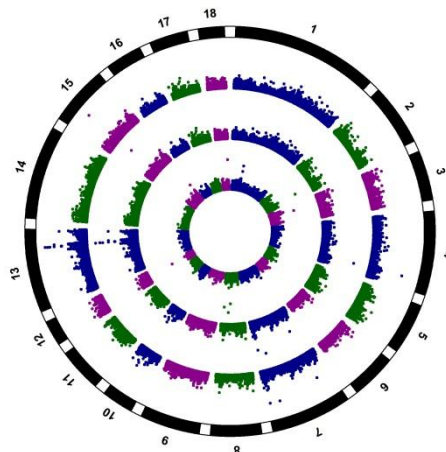
(4) Each circle with a distinctive color:

`CMplot(pig60K,plot.type="c",r=1,cir.chr.h=1.3,col=matrix(c("darkblue","darkgreen","darkmagenta"),nrow=3,byrow=T))`



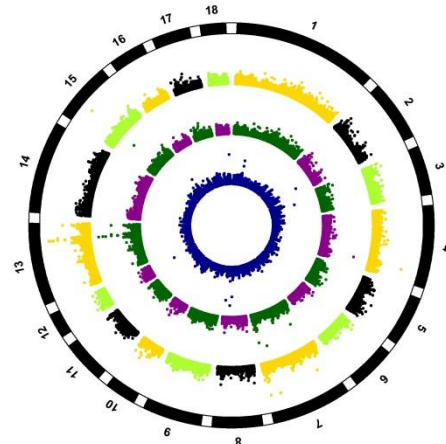
(5) The same CHR with same color:

`CMplot(pig60K,plot.type="c",r=1,LOG10=TRUE,cir.chr.h=1.3,col=c("darkblue","darkgreen","darkmagenta"))`



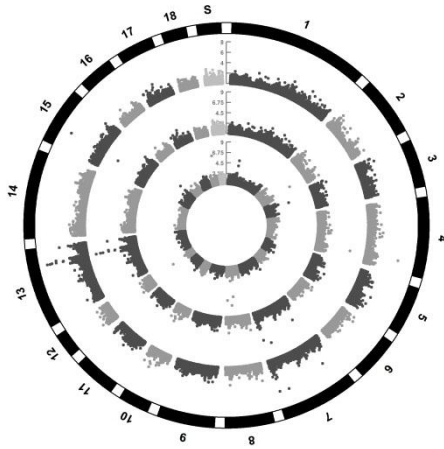
(6) Each circle with different number of colors:

`CMplot(pig60K,plot.type="c",r=1,LOG10=TRUE,cir.chr.h=1.3,col=matrix(c("darkblue",NA,NA,"darkgreen","darkmagenta",NA,"gold","black","greenyellow"),nrow=3,byrow=T))`

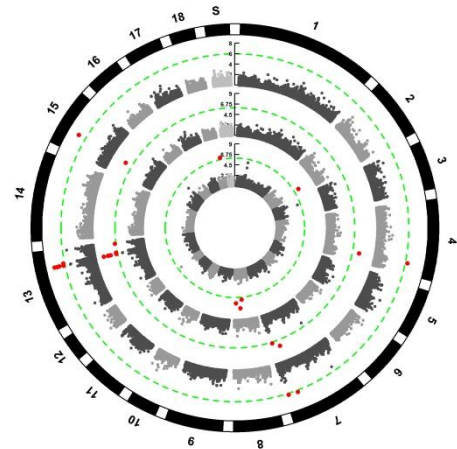


(7) Add the legend and adjust its property:

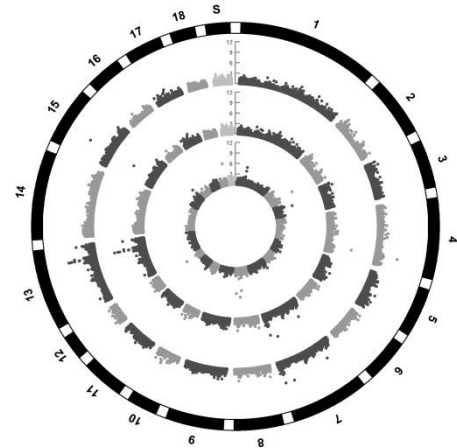
```
CMplot(pig60K,plot.type="c",col=c("grey30","grey60"),outward=TRUE,
cir.chr.h=1.3,cir.legend=TRUE,cir.legend.cex=0.5,cir.legend.col="grey45")
```

**(10) Highlight the significant points:**

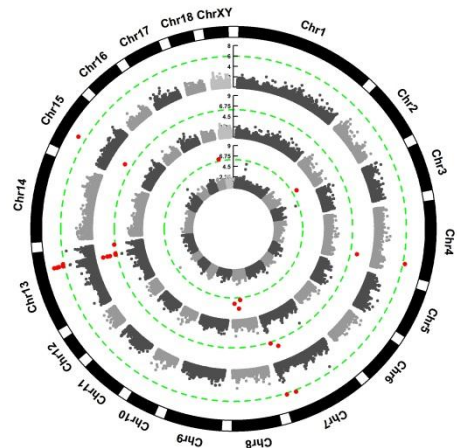
```
CMplot(pig60K,plot.type="c",col=c("grey30","grey60"),cir.chr.h=1.3,cir.legend.cex=0.5,cir.legend.col="black",threshold=0.05,threshold.col="green",threshold.lwd=2,amplify=T,signal.col="red",signal.cex=1.5)
```

**(8) Limit the same range of Y-axis of each circle:**

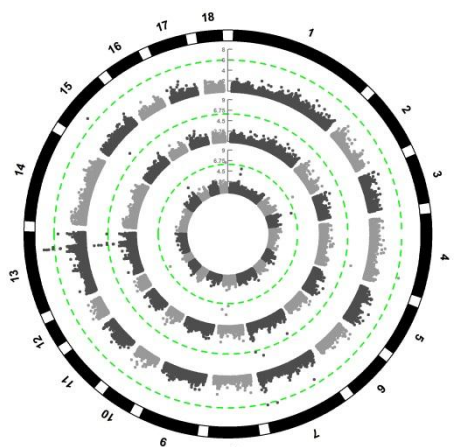
```
CMplot(pig60K,plot.type="c",col=c("grey30","grey60"),cir.chr.h=1.3,cir.legend.cex=0.5,cir.legend.col="grey45",ylim=c(0,12))
```

**(11) Change the labels of chromosomes for circle:**

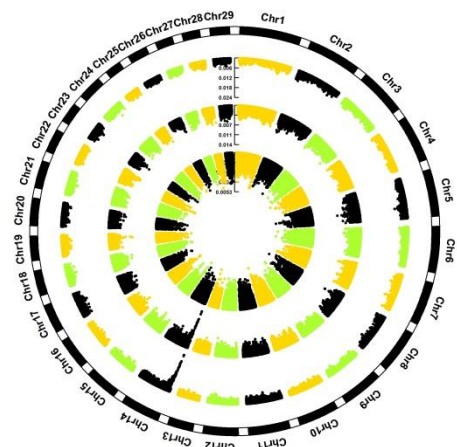
```
CMplot(pig60K,plot.type="c",col=c("grey30","grey60"),cir.chr.h=1.3,cir.legend.cex=0.5,cir.legend.col="black",threshold=0.05,threshold.col="green",threshold.lwd=2,amplify=T,signal.col="red",signal.cex=1.5,chr.labels=paste("Chr",c(1:18,"XY"),sep=""))
```

**(9) Add the significant line and adjust its property:**

```
CMplot(pig60K,plot.type="c",col=c("grey30","grey60"),cir.chr.h=1.3,cir.legend.cex=0.5,cir.legend.col="grey45",threshold=0.05,threshold.col="green",threshold.lwd=2)
```

**(12) Plot with SNP effects calculated by rrBLUP:**

```
CMplot(cattle50K,plot.type="c",col=c("orange","black"),cir.chr.h=1.3,cir.legend.cex=0.5,cir.legend.col="black",LOG10=FALSE,cir.chr.labels=paste("Chr",c(1:29),sep=""),outward=FALSE)
```



Parameters in CMplot

Parameters	Description
Pmap	a dataframe, at least four columns. The first column is the name of SNP, the second column is the chromosome of SNP, the third column is the position of SNP, and the remaining columns are the P-value of each trait(Note:each trait a column).
col	a vector or a matrix, if "col" equals to a vector, each circle use the same colors, it means that the same chromosome is drewed in the same color, the colors are not fixed, one, two, three or more colors can be used, if the length of the "col" is shorter than the length the chromosome, then colors will be applied circularly. if "col" equals to a matrix, the row is the number of circles(traits), the columns are the colors that users want to use for different circles, so each circle can be plotted in different number of colors, the missing value can be replaced by NA. For example: col=matrix(c("grey30","grey60",NA,"red","blue","green","orange",NA,NA),3,3,byrow=T).
pch	a number, the type for the points, is the same with "pch" in <plot>.
band	a number, the space between chromosomes, the default is 1(if the band equals to 0, then there would be no space between chromosome).
cir.band	a number, the space between circles, the default is 1.
H	a number, the height for each circle, each circle represents a trait, the default is 1.
ylim	a vector, the range of Y-axis when plotting the two type of Manhattans, is the same with "ylim" in <plot>.
cex.axis	a number, controls the size of numbers of X-axis and the size of labels of circle plot.
plot.type	a character or vector, only "c", "m", "q" or "b" can be used. if plot.type="c", only circle-Manhattan plot will be plotted; if plot.type="m",only Manhattan plot will be plotted; if plot.type="q",only Q-Q plot will be plotted;if plot.type="b", both circle-Manhattan, Manhattan and Q-Q plots will be plotted; if plot.type=c("m","q"), Both Manhattan and Q-Q plots will be plotted.
multitracks	a logical,if multitracks=FALSE, plotting multiple rectangular Manhattan plots on multiple tracks, if it is TRUE, all Manhattan plots will be plotted in only one track.
cex	a number or a vector, the size for the points, is the same with "size" in <plot>, and if it is a vector, the first number controls the size of points in circle plot(the default is 0.5), the second number controls the size of points in Manhattan plot(the default is 1), the third number controls the size of points in Q-Q plot(the default is 1)
r	a number, the radius for the circle(the inside radius), the default is 1.
xlab	a character, the labels for x axis.
ylab	a character, the labels for y axis.
xaxs	a character, The style of axis interval calculation to be used for the x-axis. Possible values are "r", "i", "e", "s", "d". The styles are generally controlled by the range of data or xlim, if given.
yaxs	a character, The style of axis interval calculation to be used for the y-axis. See xaxs above..
outward	logical, if outward=TRUE,then all points will be plotted from inside to outside.
threshold	a number or vector, users can adjust the significant level they want, more than one significant line can be added on the plots, if threshold=0 or NULL, then the threshold line will not be added, the default is 0.01.
threshold.col	a character or vector, the colour for the line of threshold levels.
threshold.lwd	a number or vector, the width for the line of threshold levels.

Parameters	Description
threshold.lty	a number or vector, the type for the line of threshold levels.
amplify	logical, CMplot can amplify the significant points, if amplify=T, then the points greater than the minimal significant level will be highlighted, the default: amplify=TRUE.
signal.cex	a number, if amplify=TRUE, users can set the size of significant points.
signal.pch	a number, if amplify=TRUE, users can set the shape of significant points.
signal.col	a character, if amplify=TRUE, users can set the colour of significant points, if signal.col=NULL, then the colors of significant points will not be changed.
cir.chr	logical, a boundary represents chromosome, the default is TRUE.
cir.chr.h	a number, the width for the boundary, if cir.chr=FALSE, then this parameter will be useless.
cir.chr.col	a character, the colour for the boundary, only one colour is allowed, if cir.chr.col=NULL, then the default colours are the same with the parameter "col".
chr.labels	a vector, the labels for the chromosomes of circle-Manhattan plot.
cir.legend	logical, whether to add the legend of each circle.
cir.legend.cex	a number, the size of the number of legend.
cir.legend.col	a character, the color of the axis of legend.
LOG10	logical, whether to change the p-value into log ₁₀ (p-value).
box	logical, this function draws a box around the current Manhattan plot.
conf.int	logical, whether to draw the confidence interval on QQ-plot.
conf.int.col	a character, the color of the confidence interval on QQ-plot.
plot0	logical, users can choose whether to plot the chromosome (written as 0) which has no clear chromosome number in all plots.
file.output	a logical, users can choose whether to output the plot results.
file	a character, users can choose the different output formats of plot, so for, ".jpg", ".pdf", ".tiff" can be selected by users.
dpi	a number, the picture element for .jpg and .tiff files. The default is 300.