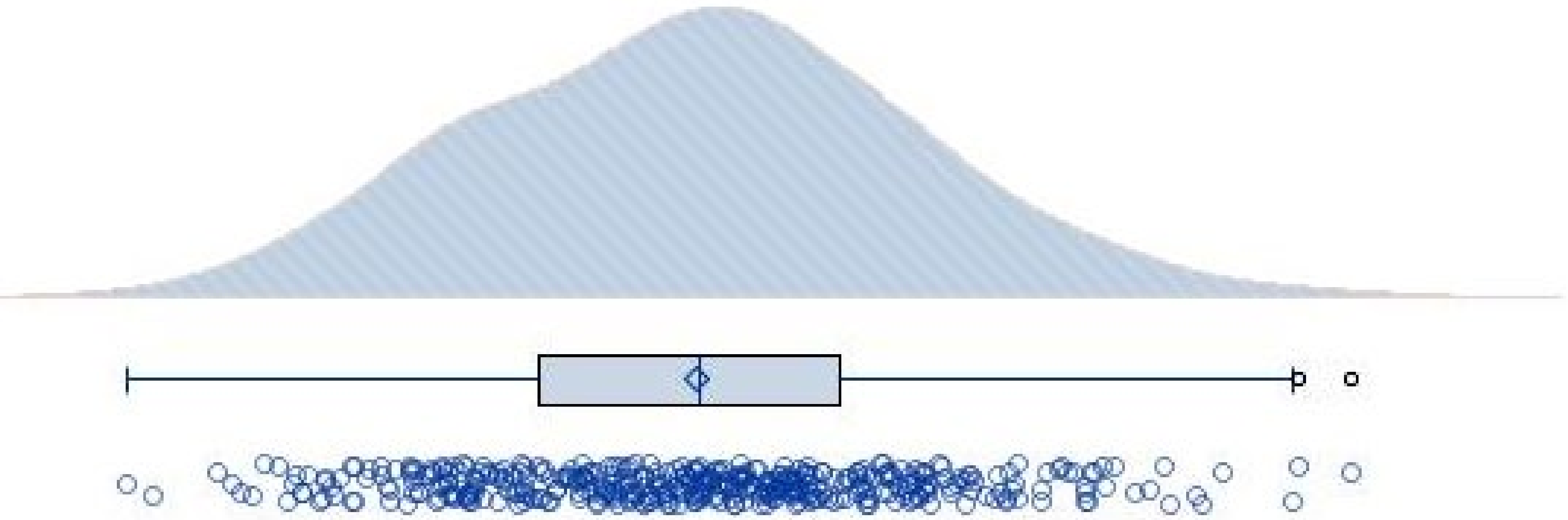
2022-12-08

EPS

Implementation of Raincloud Plot with SAS and its USE for Clinical Trial Data

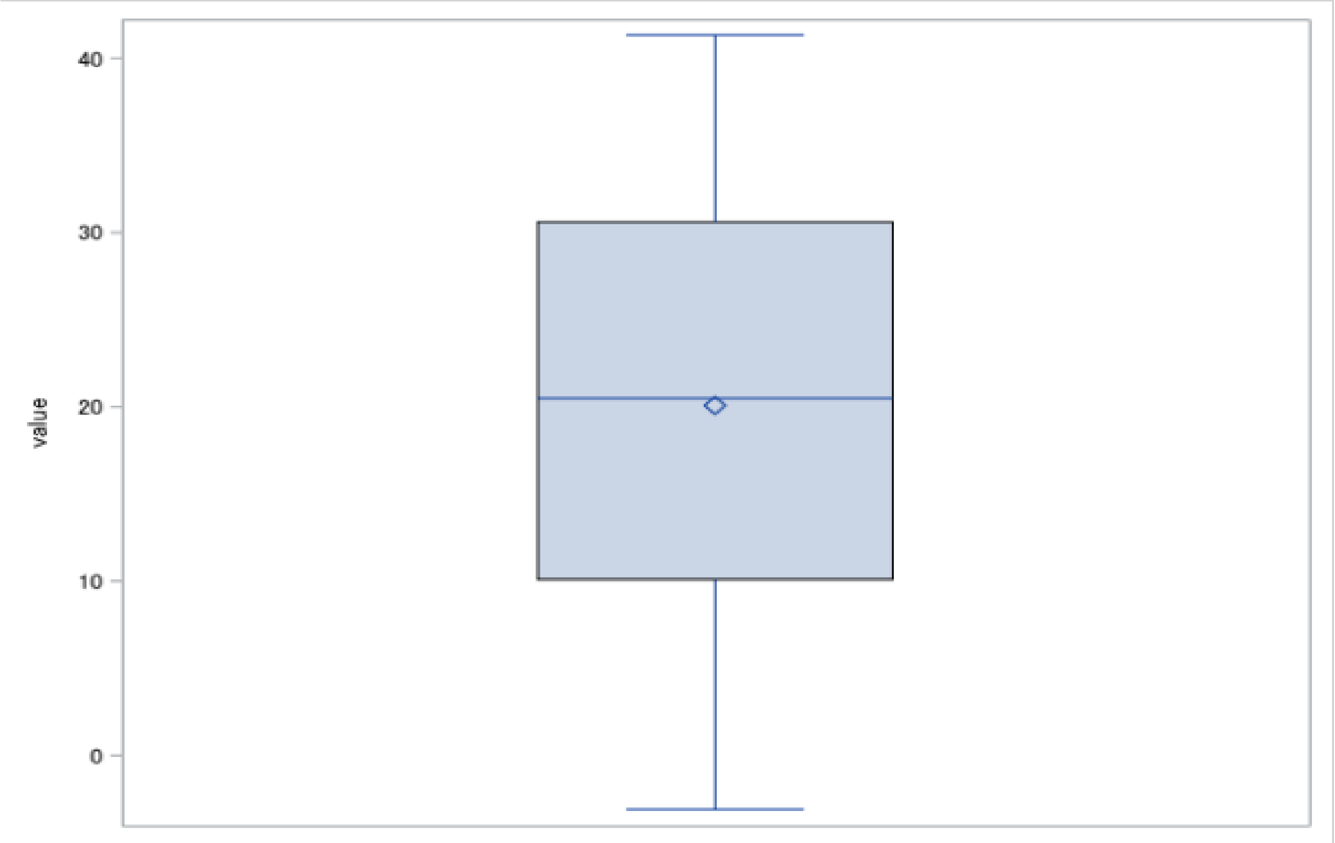
Yutaka Morioka
EPS Corporation

Abstract



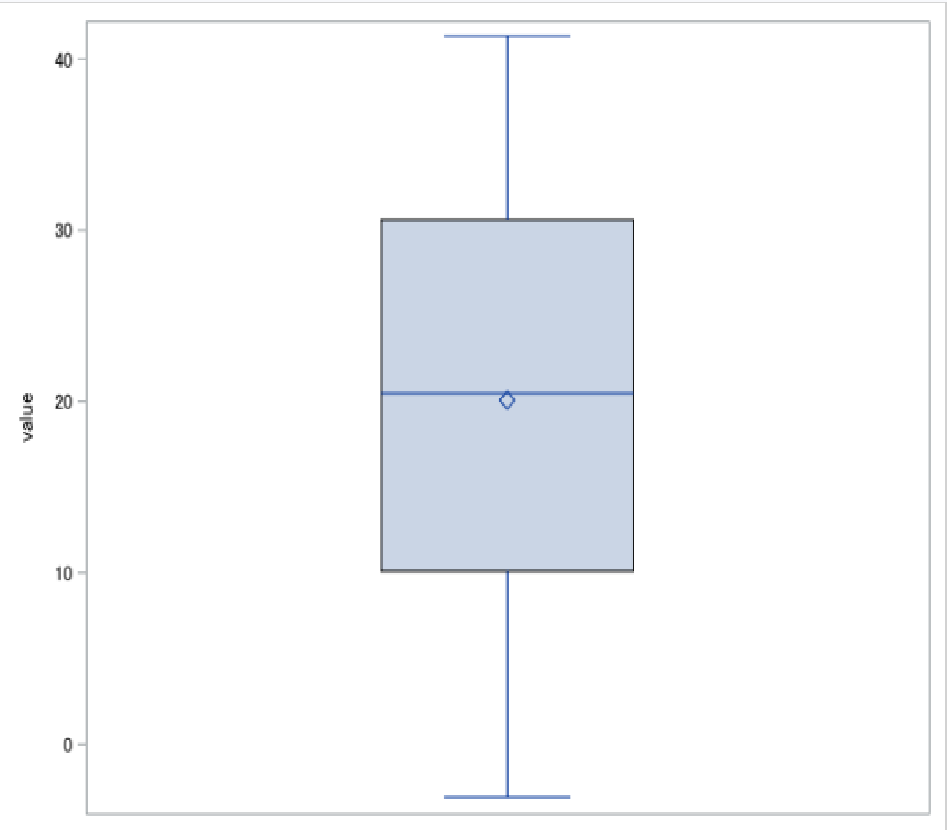
A boxplot is a standardized way of displaying the dataset based on the five-number summary.

Minimum (the lowest data point in the data set excluding any outliers)
Maximum (the highest data point in the data set excluding any outliers)
First quartile (Q1 or 25th percentile)
Median (*Q*2 or 50th percentile)
Third quartile (Q3 or 75th percentile)

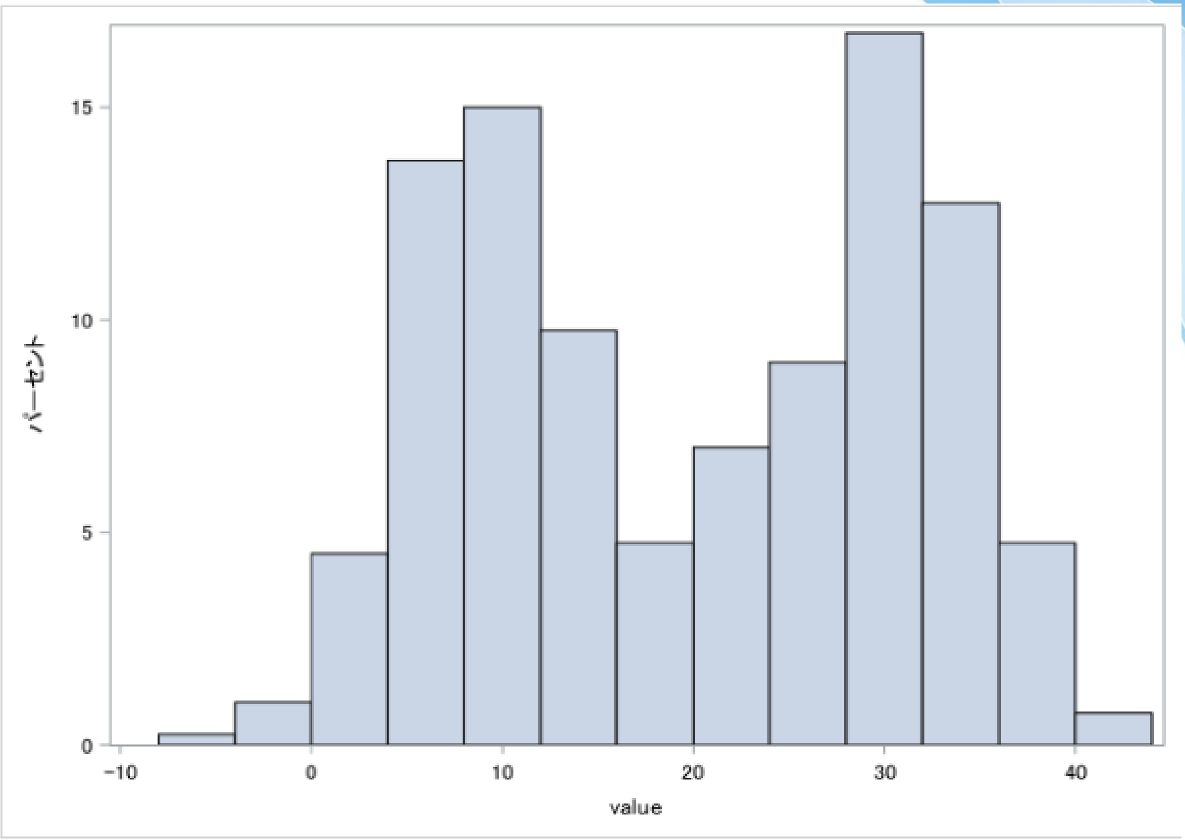


Copyright©EPS All rights reserved.

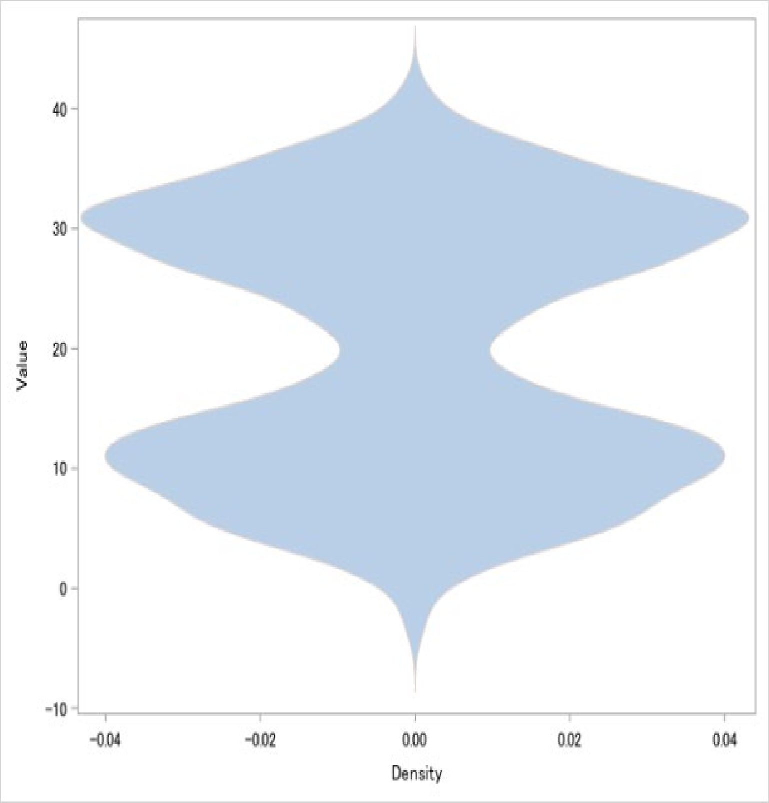
We cannot always grasp the data distribution from Boxplot .



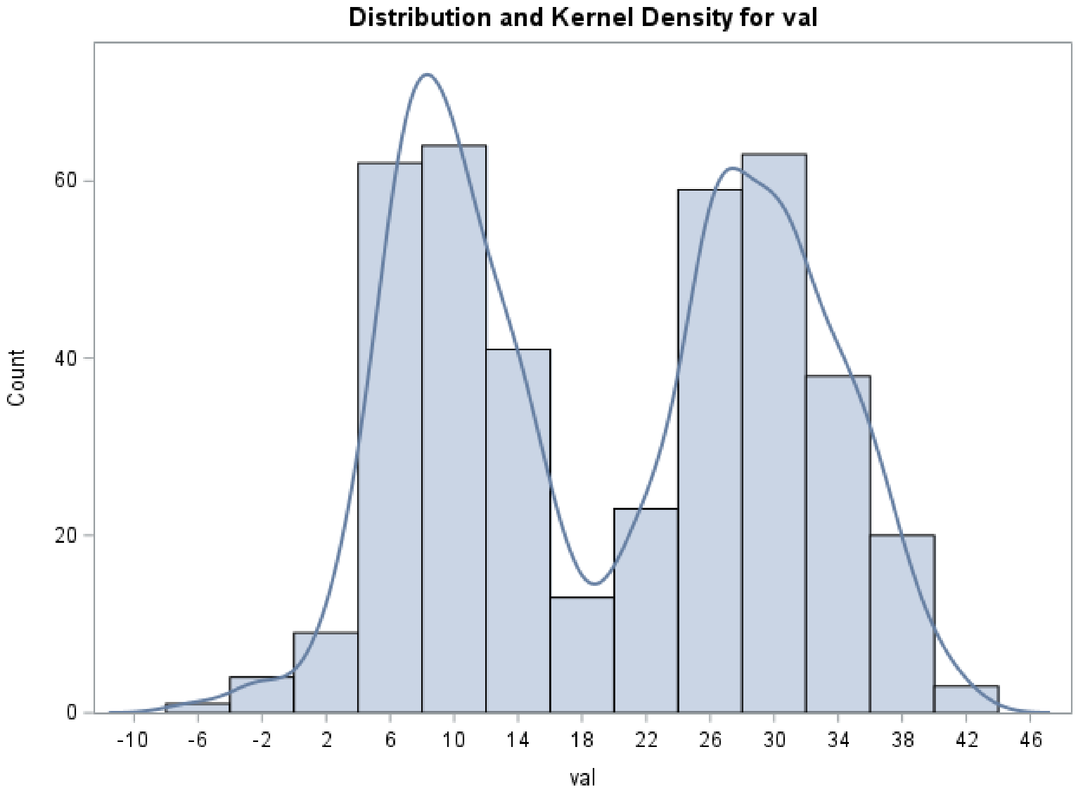
=



Violin plot

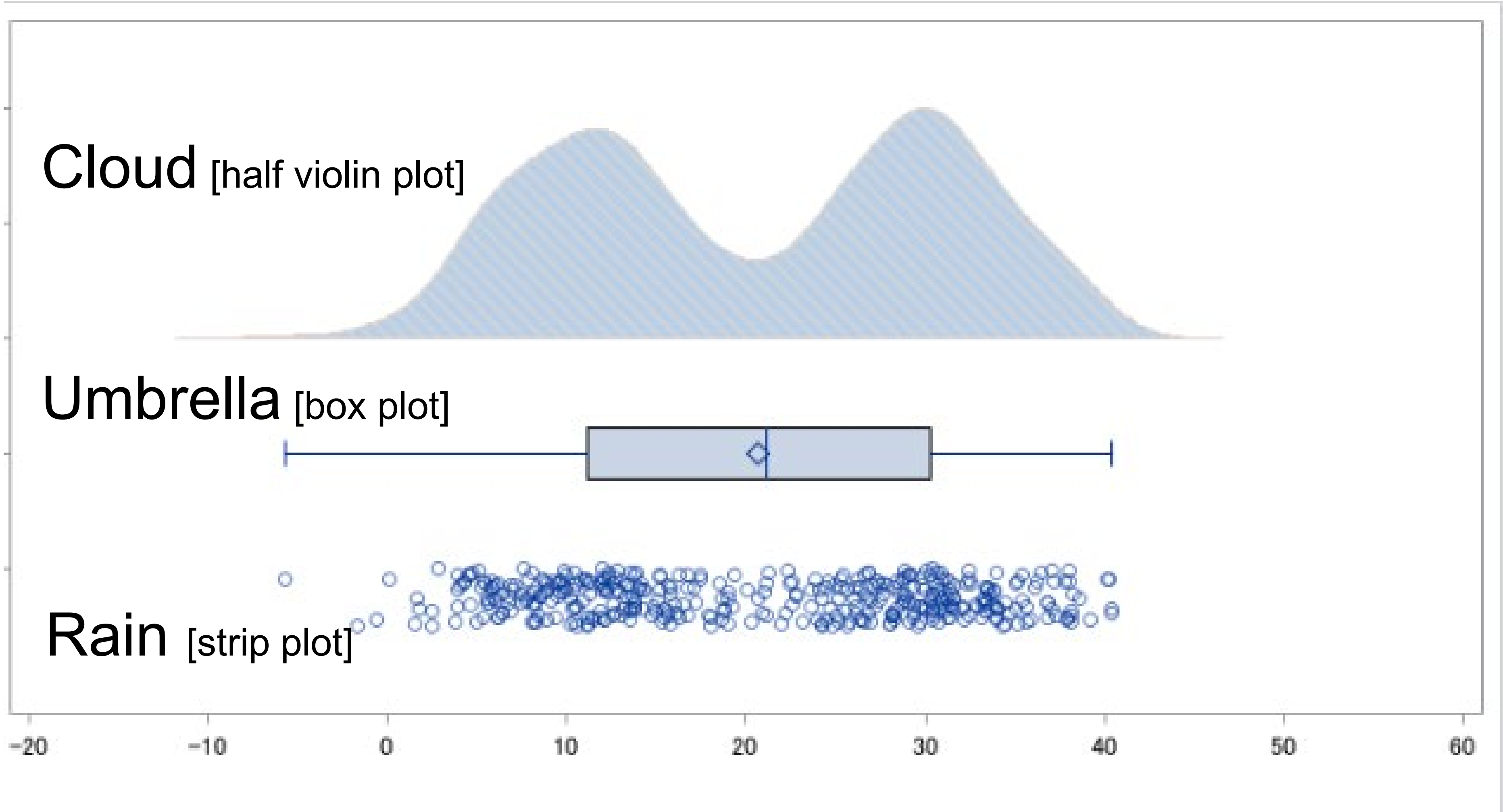


Kernel density estimation



Copyright©EPS All rights reserved.

Raincloud plot

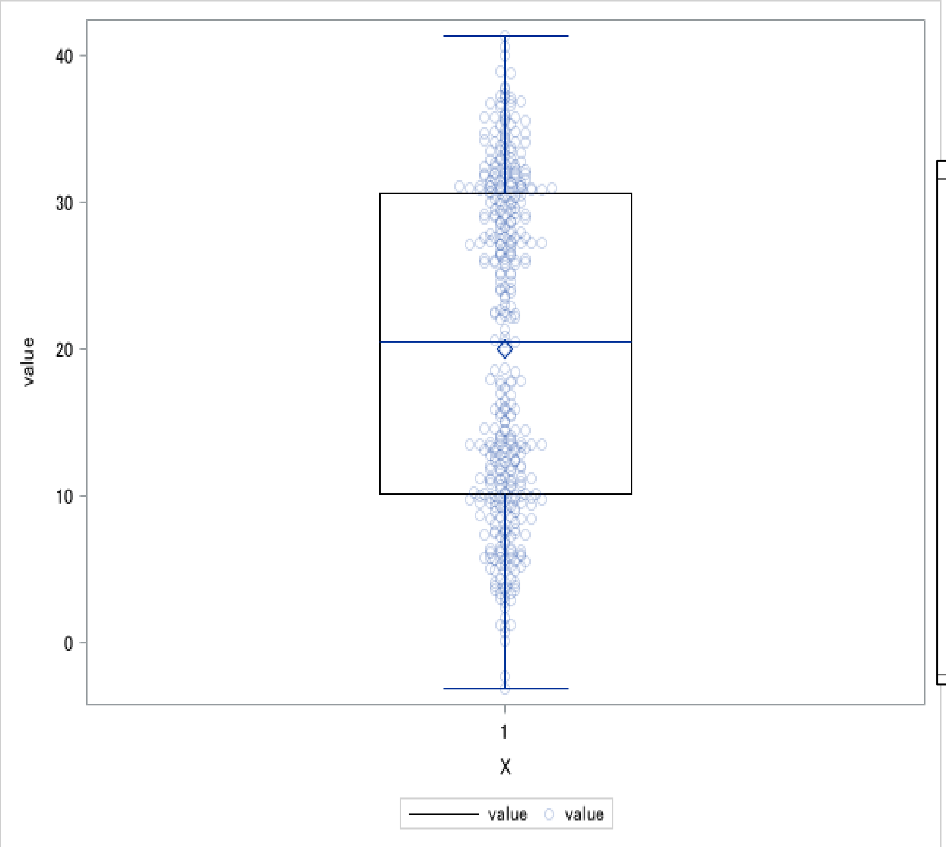


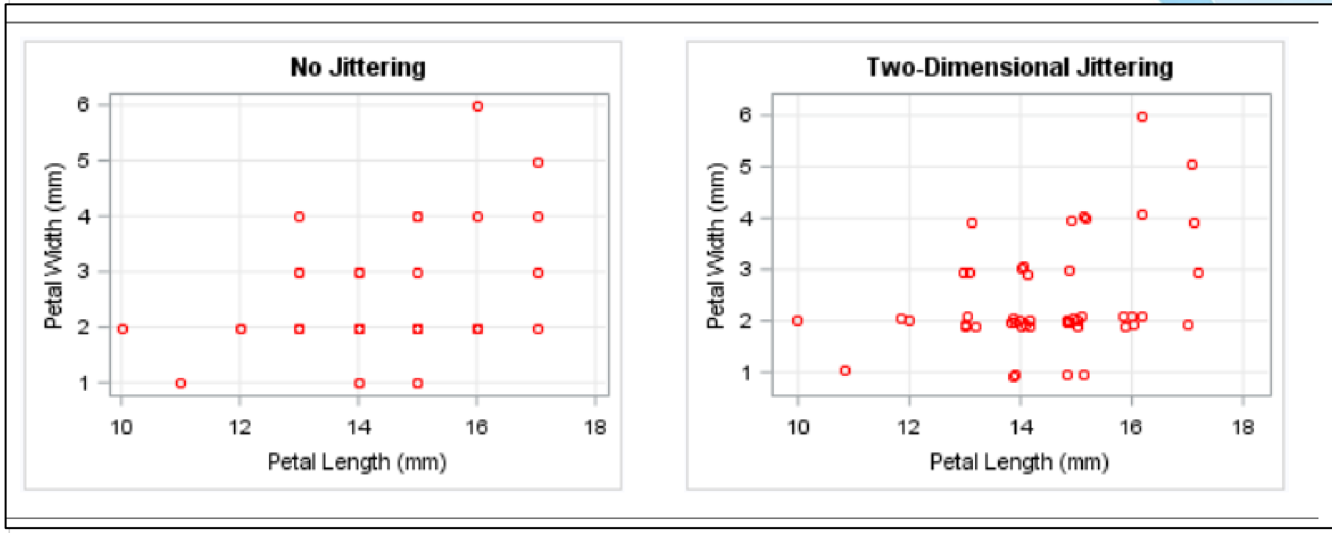
Cloud [half violin plot]

Umbrella [box plot]

Rain [strip plot]

Bee swarm





```
PROC SGPLOT DATA=sample;
  SCATTER X=x1 Y=y1 / JITTER JITTERWIDTH=0.5;
RUN;
```

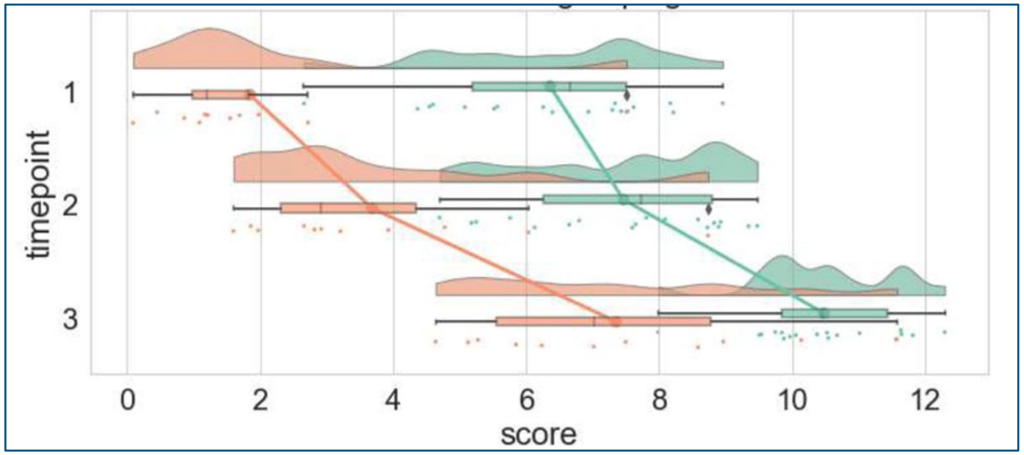
Jittering strip plot

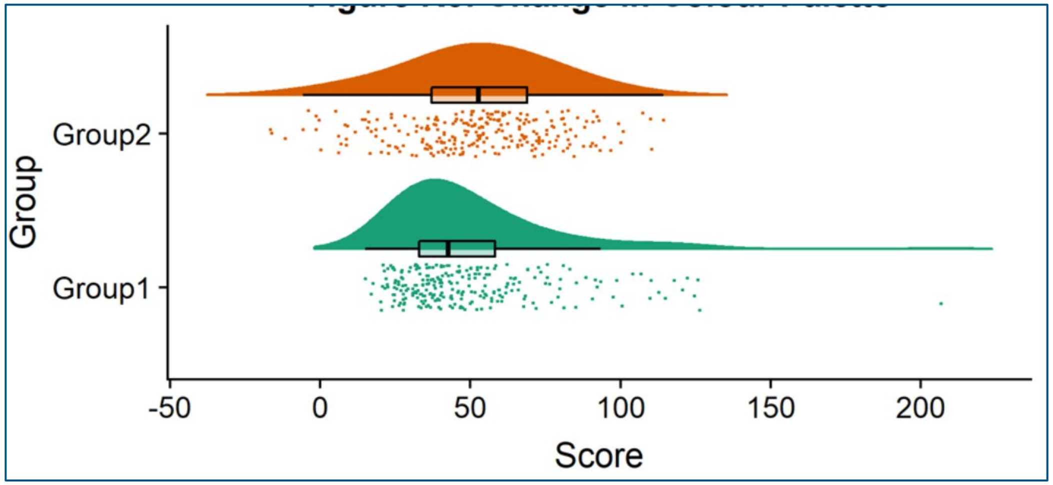
Raincloud plots: a multi-platform tool for robust data visualization

（ Micah Allen et al. ）

https://wellcomeopenresearch.org/articles/4-63

Raincloud plot was proposed by Micah Allen et al. at Aarhus University.
Many examples of implementation in R, Python, and Matlab were presented.
In the spirit of Open Science, they have introduced their technology widely on the Web and SNS, and We have made the most of the feedback.





Test data

```
data test;
call streaminit(1080);
GROUP="A";
do id=1 to 200;

val=rand("normal",10,5);output;
end;
do id=201 to 400;

val=rand("normal",30,5);output;
end;
GROUP="B";
do id=1 to 200;

val=rand("normal",18,5);output;
end;
do id=201 to 400;

val=rand("normal",22,5);output;
end;
run;
```

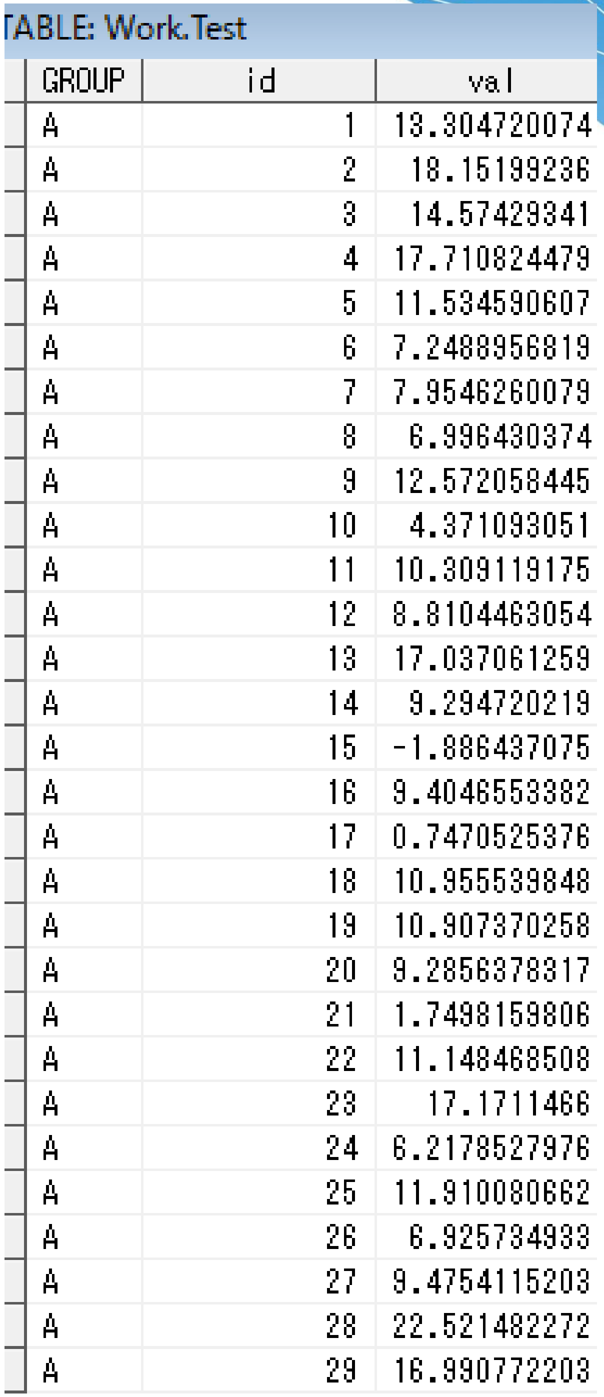


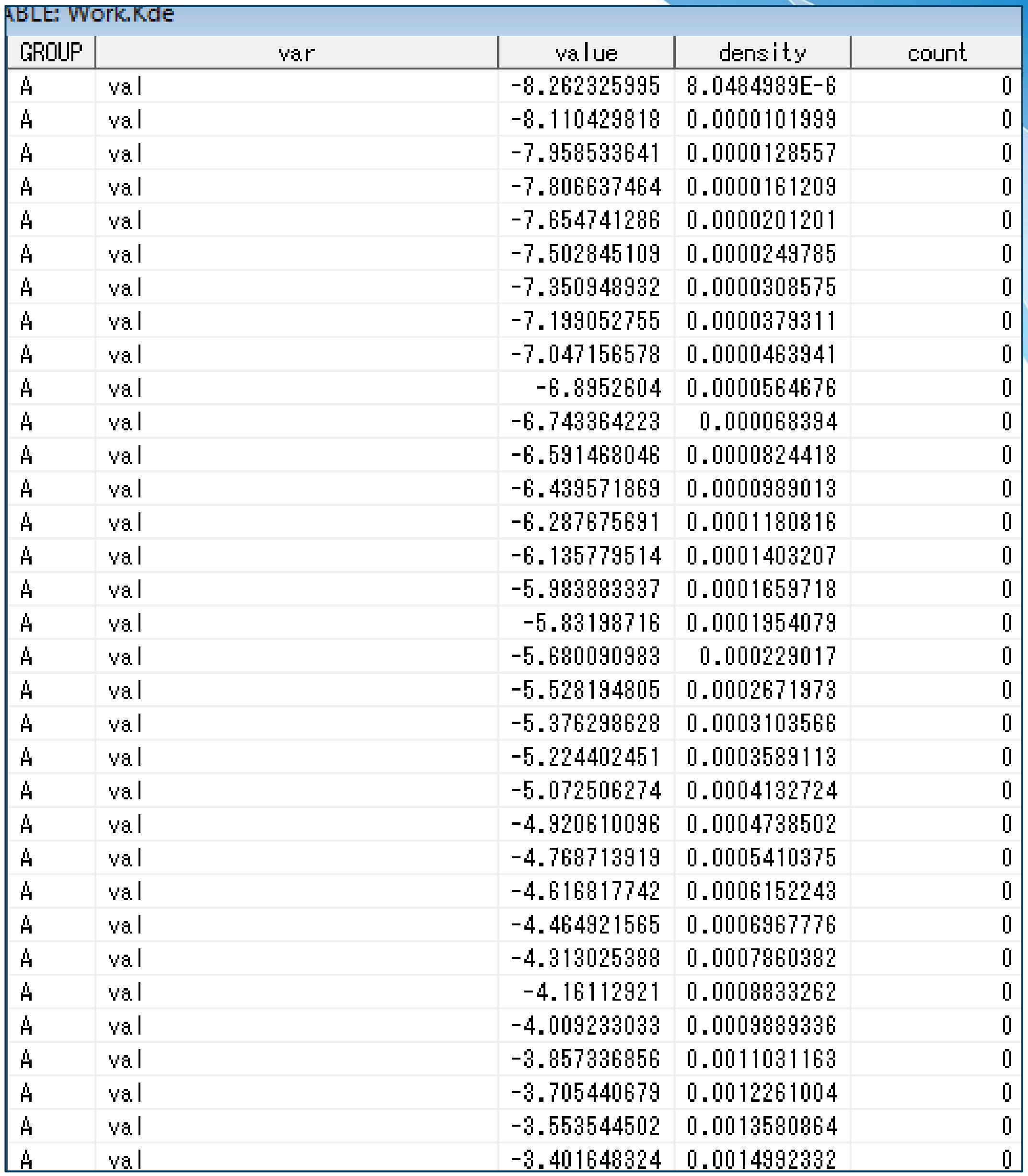
TABLE: Work.Test

GROUP	id	val
A	1	13.304720074
A	2	18.15199236
A	3	14.57429341
A	4	17.710824479
A	5	11.534590607
A	6	7.2488956819
A	7	7.9546260079
A	8	6.996430374
A	9	12.572058445
A	10	4.371093051
A	11	10.309119175
A	12	8.8104463054
A	13	17.037061259
A	14	9.294720219
A	15	-1.886437075
A	16	9.4046553382
A	17	0.7470525376
A	18	10.955539848
A	19	10.907370258
A	20	9.2856378317
A	21	1.7498159806
A	22	11.148468508
A	23	17.1711466
A	24	6.2178527976
A	25	11.910080662
A	26	6.925734933
A	27	9.4754115203
A	28	22.521482272
A	29	16.990772203

Kernel density estimation　（Violin plot）

```
proc kde data = test;
   univar val / out = kde;
   by GROUP;
run;
```

```
data wk2;
   set kde test;
run;
proc sort data = wk2;
      by group;
run;
```



ABLE: Work.Kde

GROUP	var	value	density	count
A	val	-8.262325995	8.0484989E-6	0
A	val	-8.110429818	0.0000101999	0
A	val	-7.958533641	0.0000128557	0
A	val	-7.806637464	0.0000161209	0
A	val	-7.654741286	0.0000201201	0
A	val	-7.502845109	0.0000249785	0
A	val	-7.350948932	0.0000308575	0
A	val	-7.199052755	0.0000379311	0
A	val	-7.047156578	0.0000463941	0
A	val	-6.8952604	0.0000564676	0
A	val	-6.743364223	0.000068394	0
A	val	-6.591468046	0.0000824418	0
A	val	-6.439571869	0.0000989013	0
A	val	-6.287675691	0.0001180816	0
A	val	-6.135779514	0.0001403207	0
A	val	-5.983883337	0.0001659718	0
A	val	-5.83198716	0.0001954079	0
A	val	-5.680090983	0.000229017	0
A	val	-5.528194805	0.0002671973	0
A	val	-5.376298628	0.0003103566	0
A	val	-5.224402451	0.0003589113	0
A	val	-5.072506274	0.0004132724	0
A	val	-4.920610096	0.0004738502	0
A	val	-4.768713919	0.0005410375	0
A	val	-4.616817742	0.0006152243	0
A	val	-4.464921565	0.0006967776	0
A	val	-4.313025388	0.0007860382	0
A	val	-4.16112921	0.0008833262	0
A	val	-4.009233033	0.0009889336	0
A	val	-3.857336856	0.0011031163	0
A	val	-3.705440679	0.0012261004	0
A	val	-3.553544502	0.0013580864	0
A	val	-3.401648324	0.0014992332	0

Graph Template

```
ods graphics on / height = 3in width = 12in;
proc template;
   define statgraph RCP;
      begingraph;
         layout overlay
            / xaxisopts = (label = ' ' type= linear
                       linearopts = (viewmin = -20 viewmax = 60
                       tickvaluesequence = (start = -20 end = 60 increment = 10)))
              yaxisopts = (label = ' ');
           bandplot x = value limitlower = 0 limitupper = density/ display = all;
            boxplot y = val x = eval(-0.02 + coalesce(0, val))
               / orient   = horizontal
                 boxwidth = 0.3;
            scatterplot x = val
                 y = eval(-0.05 + 0.01*cdf(‘NORMAL’, rannor(1234))+coalesce(0,val))
               / markerattrs = (symbol = circle size = 8 transparency = 0.4);
         endlayout;
      endgraph;
   end;
run;
```

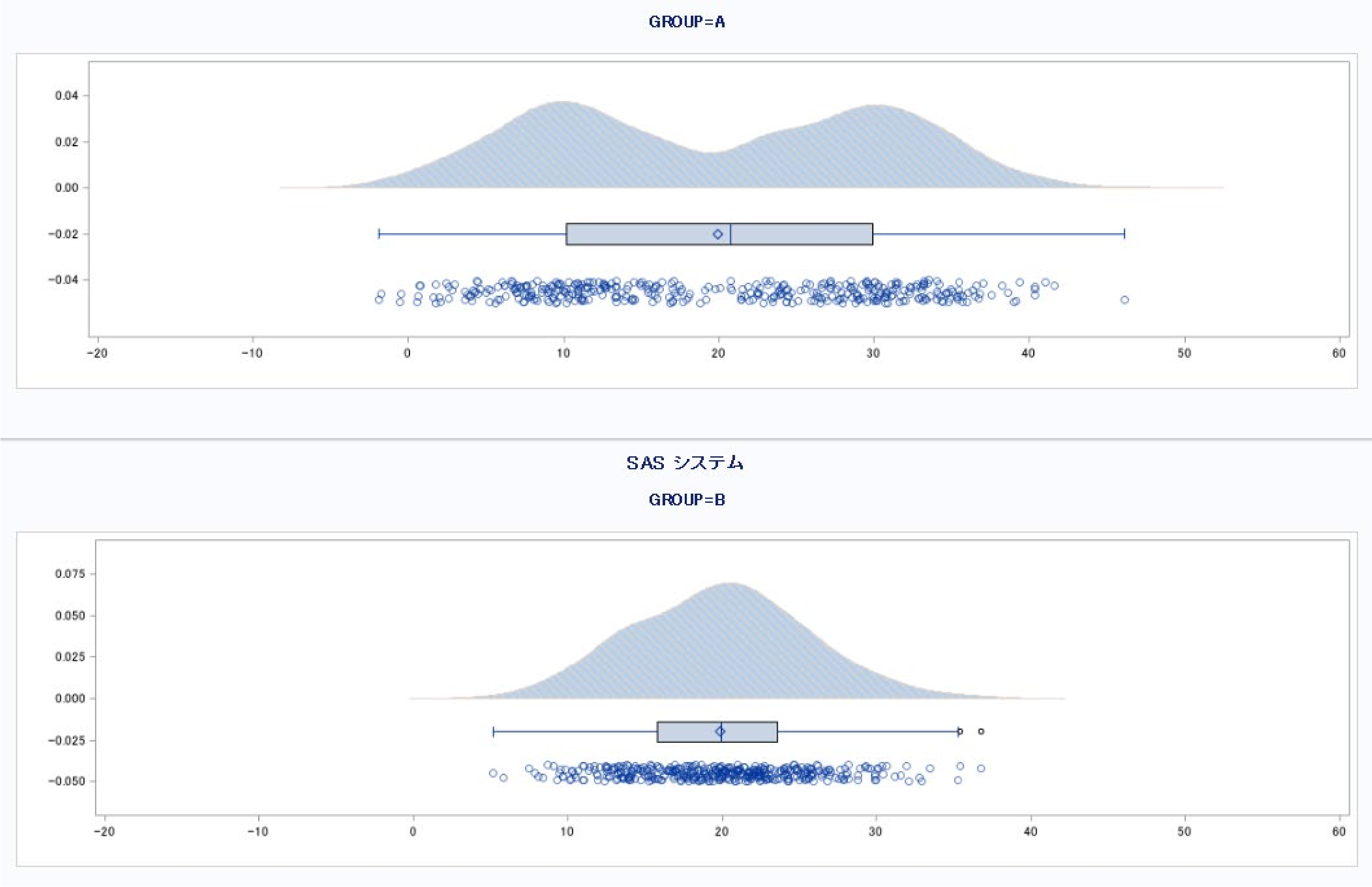
Axis definition

Violin plot

Box plot

Stripplot (with random jittering)

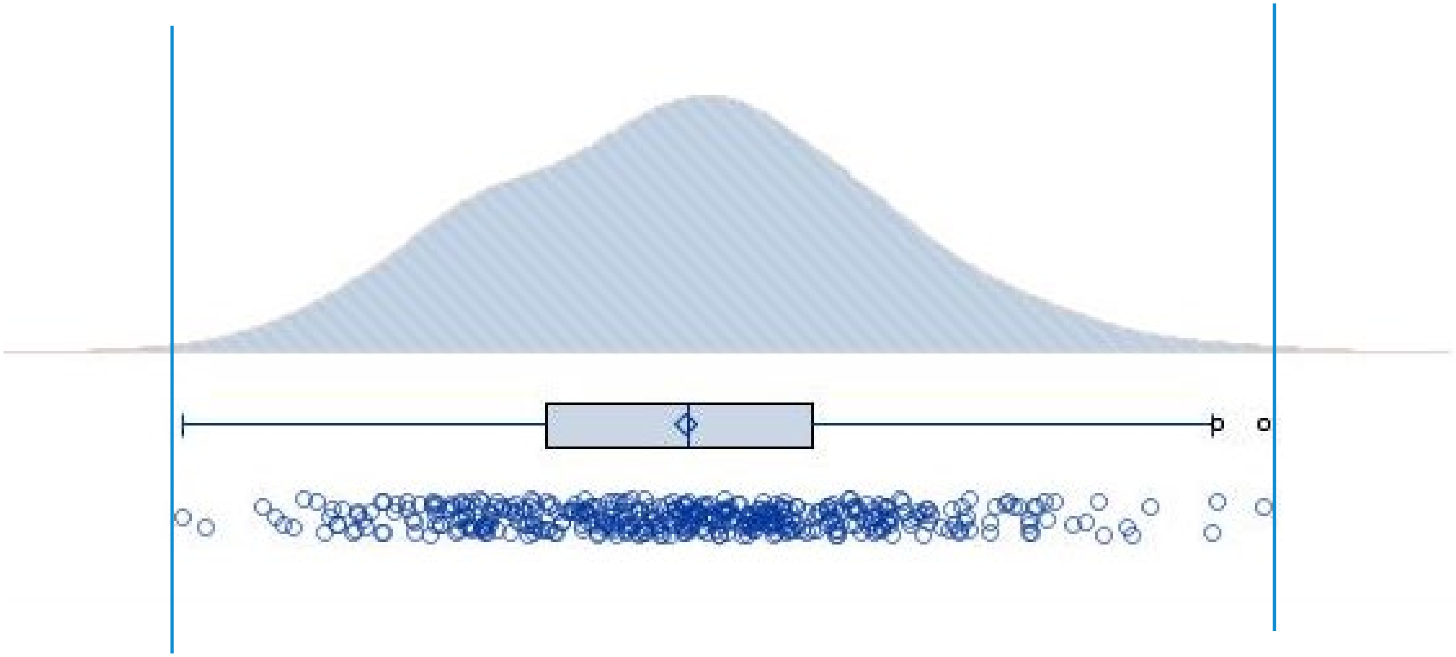
```
ods html path = "&outpath" file = 'test.html';
 proc sgrender data = wk2 template = RCP;
    by group;
 run;
 ods html close;
```



GROUP=A

SAS システム

GROUP=B

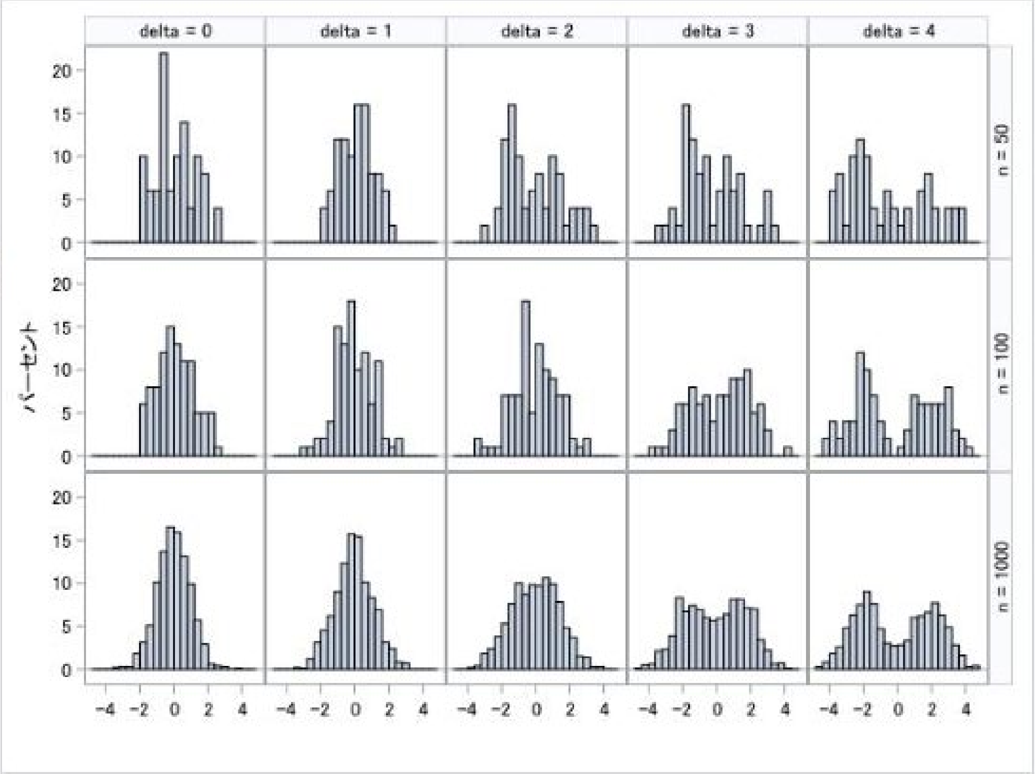


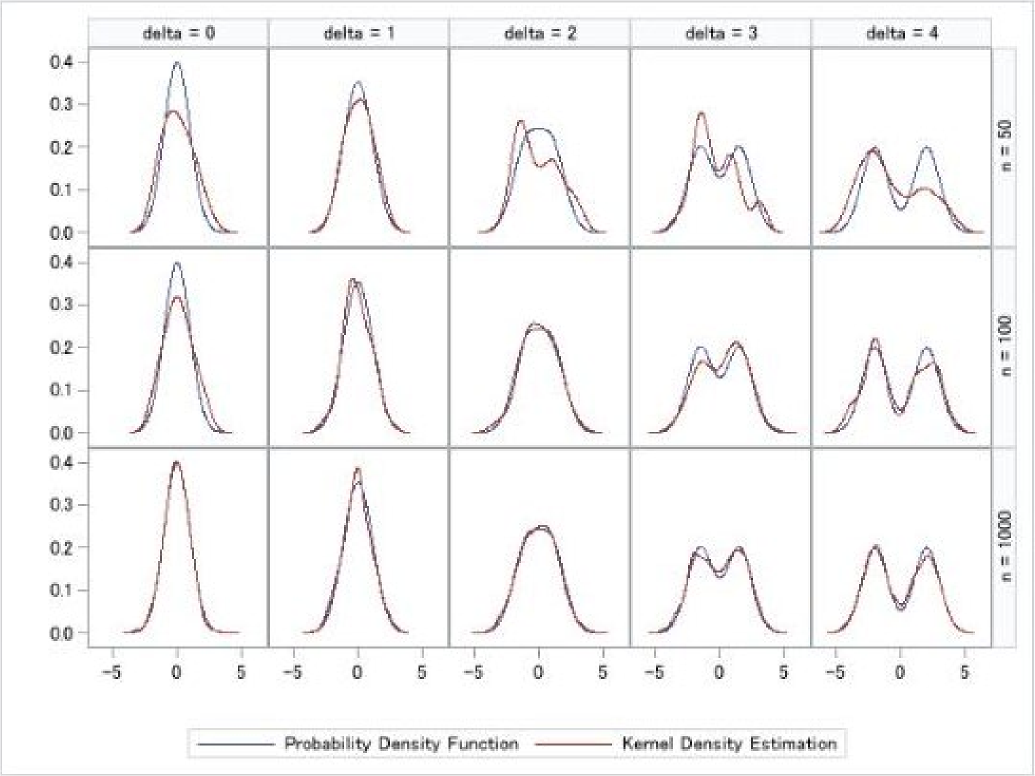
Kernel density estimation also estimates points beyond the range of real data
Proc KDE + **truncate** option ⇒ Cut beyond the range of the actual data.

```
proc kde data = test;
   univar val / out = kde truncate;
   by GROUP;
run;
```

Kernel density estimation
– simulation 1

If N is less than 50 or so, the accuracy of determining single/multiple peaks does not seem to be good.





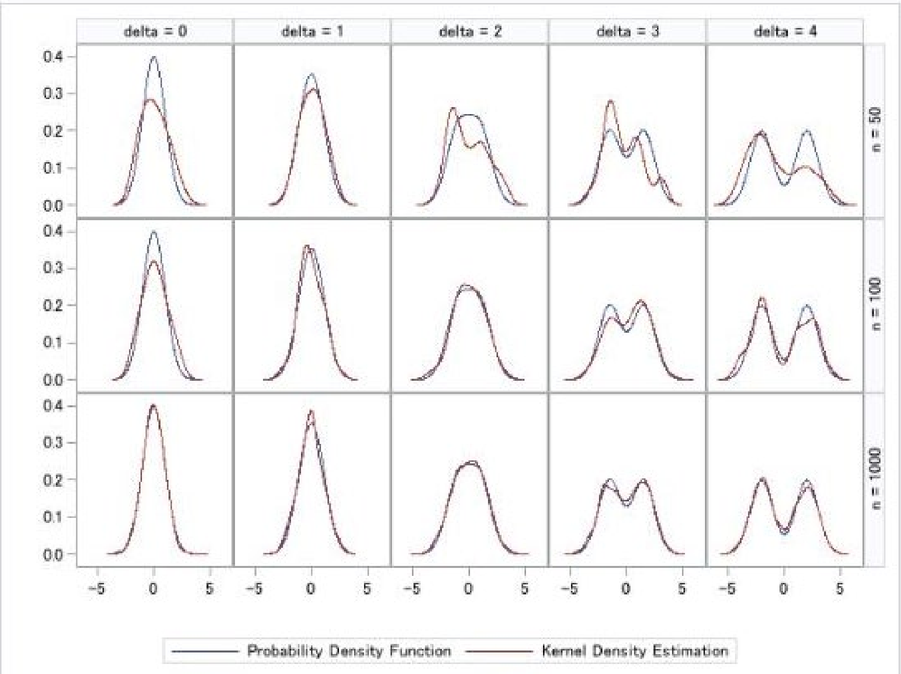
Kernel density estimation
– simulation 2

Differences in methods used to compute the bandwidth .

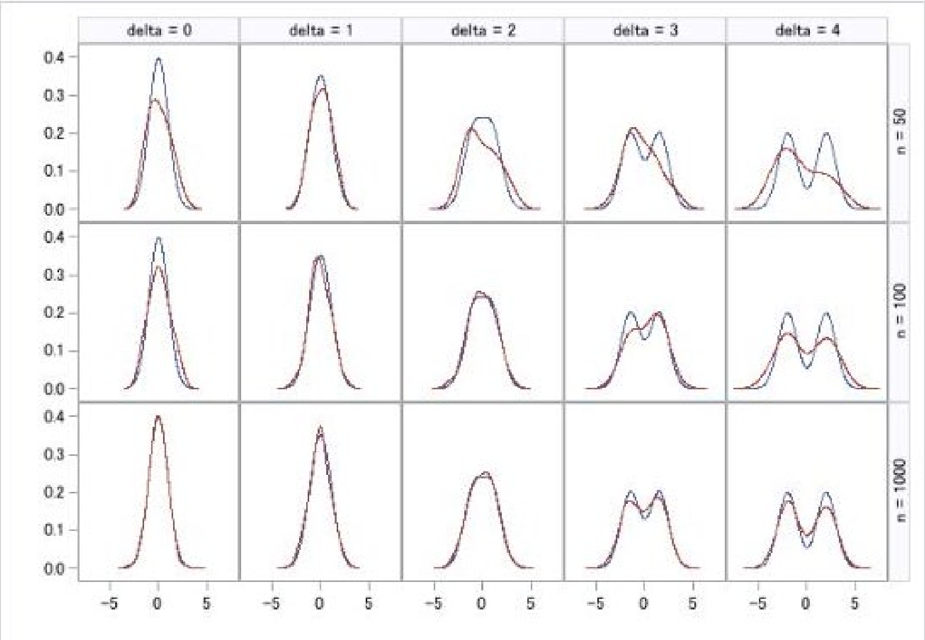
```
proc kde data = test;
   univar val / out = kde method = os;
run;
```

Oversmoothed Estimation（More smoothing）

Sheather-Jones Plug In（default）

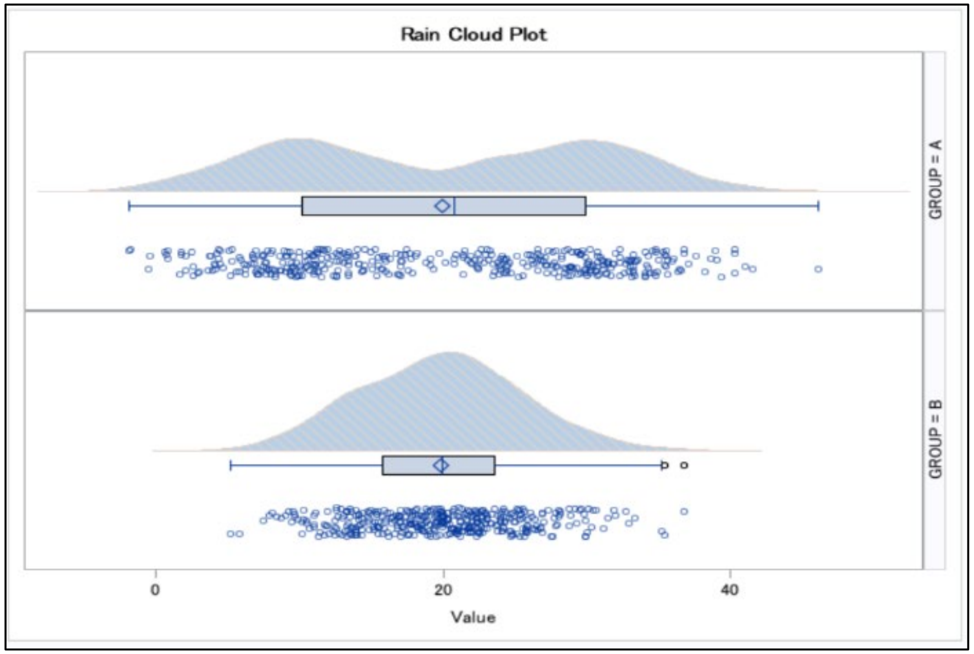


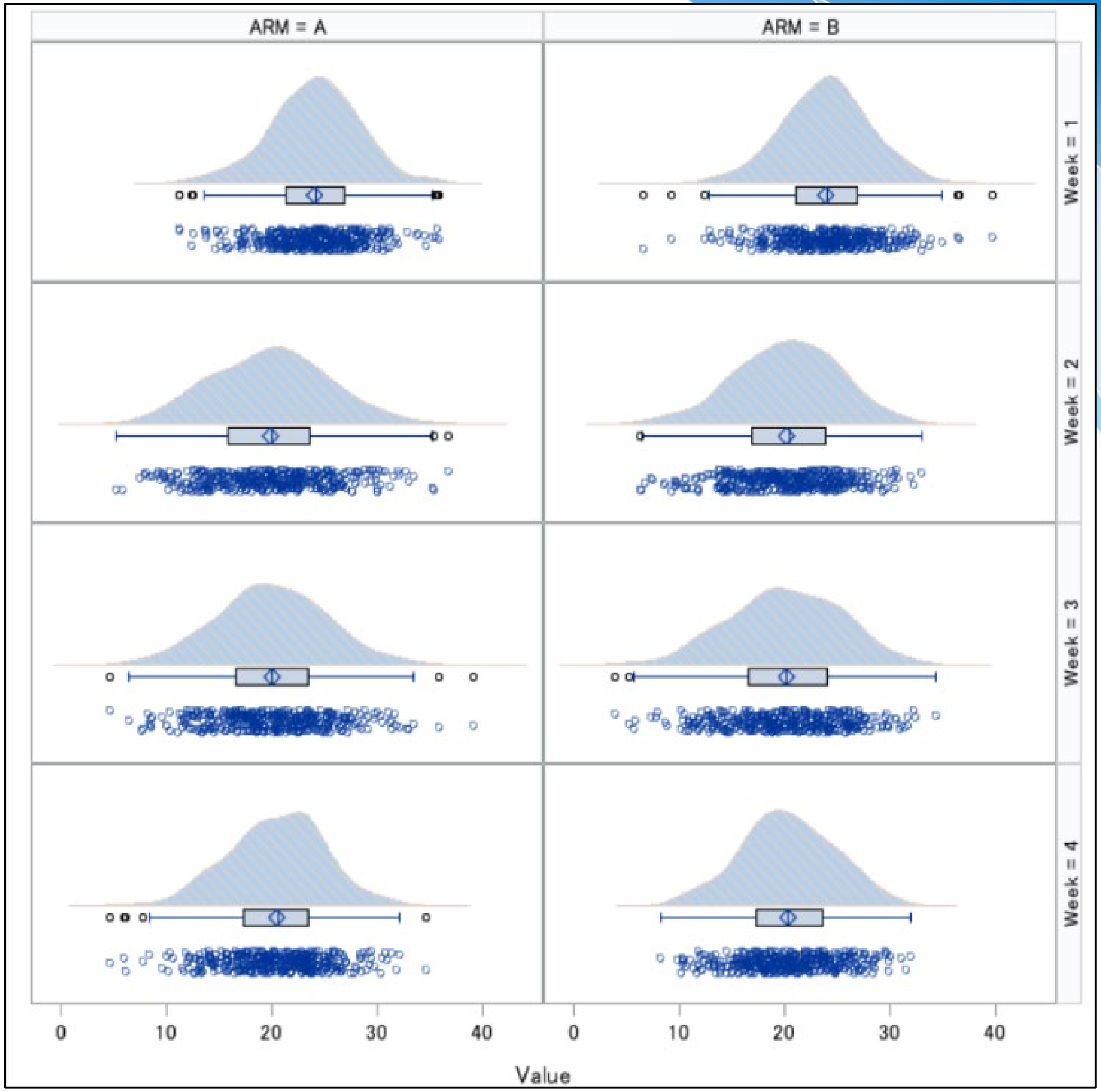
method=SJPI (default)



method=OS

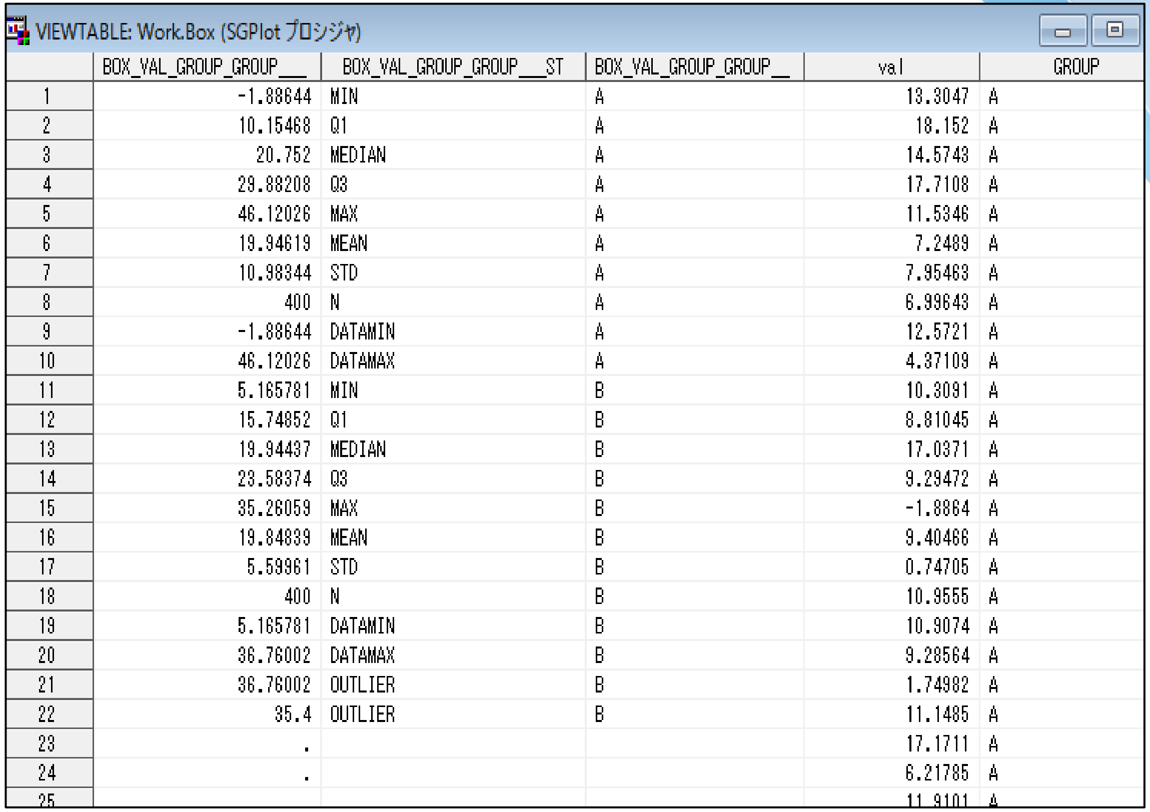
Multi Panel Raincloud plot





Tips

```
ods _all_ close;
ods output SGPlot=box;
proc sgplot data=test;
  vbox val/group=group;
run;
ods html;
```



VIEWTABLE: Work.Box (SGPlot プロシジャ)

	BOX_VAL_GROUP_GROUP___	BOX_VAL_GROUP_GROUP___ST	BOX_VAL_GROUP_GROUP__	val	GROUP
1	-1.88644	MIN	A	13.3047	A
2	10.15468	Q1	A	18.152	A
3	20.752	MEDIAN	A	14.5743	A
4	29.88208	Q3	A	17.7108	A
5	46.12026	MAX	A	11.5346	A
6	19.94619	MEAN	A	7.2489	A
7	10.98344	STD	A	7.95463	A
8	400	N	A	6.99643	A
9	-1.88644	DATAMIN	A	12.5721	A
10	46.12026	DATAMAX	A	4.37109	A
11	5.165781	MIN	B	10.3091	A
12	15.74852	Q1	B	8.81045	A
13	19.94437	MEDIAN	B	17.0371	A
14	23.58374	Q3	B	9.29472	A
15	35.26059	MAX	B	-1.8864	A
16	19.84839	MEAN	B	9.40466	A
17	5.59961	STD	B	0.74705	A
18	400	N	B	10.9555	A
19	5.165781	DATAMIN	B	10.9074	A
20	36.76002	DATAMAX	B	9.28564	A
21	36.76002	OUTLIER	B	1.74982	A
22	35.4	OUTLIER	B	11.1485	A
23	.			17.1711	A
24	.			6.21785	A
25				11.9101	A

Get only the parameters of boxplot as a data set. No plot is created.

Dataset for Multi Panel Raincloud plot

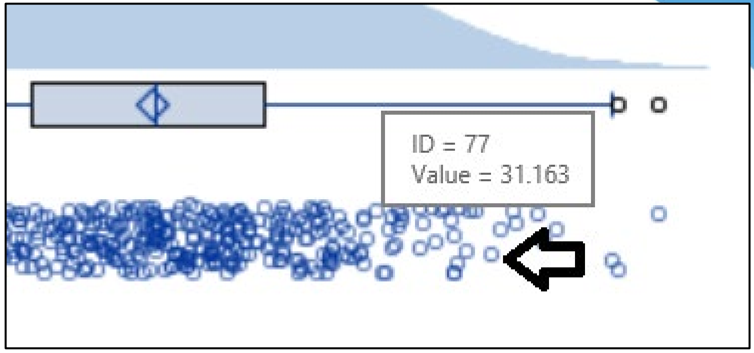
```
data wk2;
set kde(in=ina) /*violin*/
    box(in=inb where=(^missing(BOX_VAL_GROUP_GROUP____Y))) /*box*/
    test(in=inc) /*sprit*/;
    call streaminit(20220901);
    if ina then low=0;
    if missing(BOX_VAL_GROUP_GROUP___ST) then BOX_VAL_GROUP_GROUP___ST="DUMMY";
    if ^missing(BOX_VAL_GROUP_GROUP___GP) then group=BOX_VAL_GROUP_GROUP___GP;
    if inb then dummy_x=-0.01;
    if inc then do;
      dummy_y=-0.05;
      random=rand("uniform")*0.01;
      if ranuni(777)<0.5 then dummy_y=dummy_y - random; /
      else dummy_y=dummy_y + random;
    end;
run;
```

Graph Template --- Multi Panel Raincloud plot

```
ods graphics on /imagemap=on tipmax=5000;
ods html path=“xxxxxxxxxxxxx" file="test.html";
proc template ;
 define statgraph RCP ;
  begingraph;
   entrytitle "Rain Cloud Plot";
    layout datalattice rowvar=group/columnaxisopts=(label=“Value”)
                 rowaxisopts=(display=none);
     layout prototype;
      bandplot x=value limitupper=density  limitlower=low  ;
      boxplotparm y=BOX_VAL_GROUP_GROUP____Y x=dummy_x
                 stat=BOX_VAL_GROUP_GROUP___ST /boxwidth=0.3 orient = horizontal  ;
      scatterplot x=val y=dummy_y/ jitter=auto jitteropts=(axis=Y width=1)
      markerattrs=(symbol=circle size=8 transparency=0.4)
      rolename=(tip1=ID tip2=VAL) tip=(tip1 tip2) tiplabel=(tip1="ID" tip2="Value") ;
     endlayout;
    endlayout;
  endgraph;
 end;
run;
proc sgrender data=wk2 template=RCP ;
run;
ods html close;
```

Imagemap –Graph Cursor Action

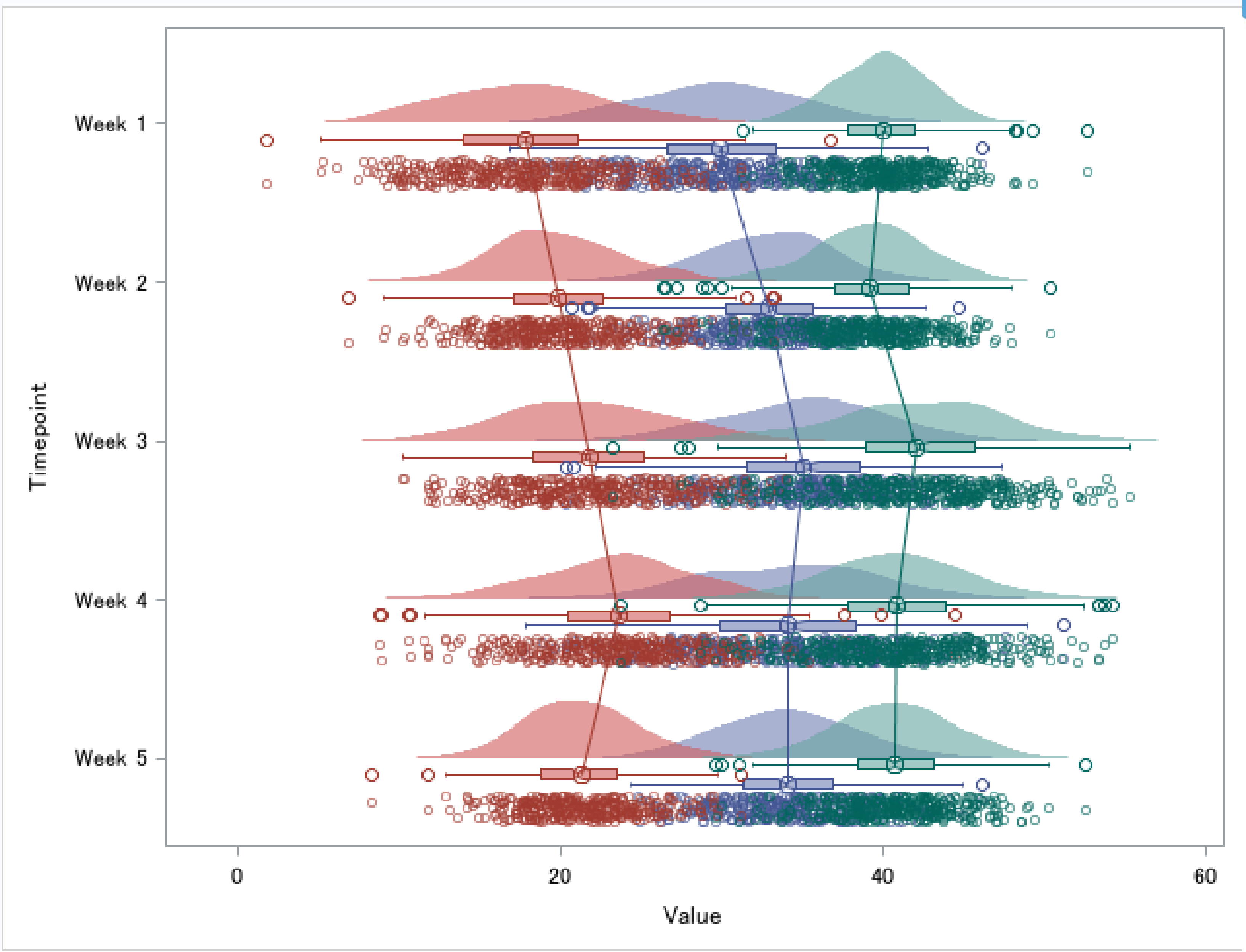
```
ods graphics on /imagemap=on
tipmax=5000;
```


ID = 77
Value = 31.163

```
scatterplot x=val y=dummy_y/ jitter=auto jitteropts=(axis=Y width=1)
markerattrs=(symbol=circle size=8 transparency=0.4)
rolename=(tip1=ID tip2=VAL) tip=(tip1 tip2)
tiplabel=(tip1=“ID”   tip2="Value") ;
```

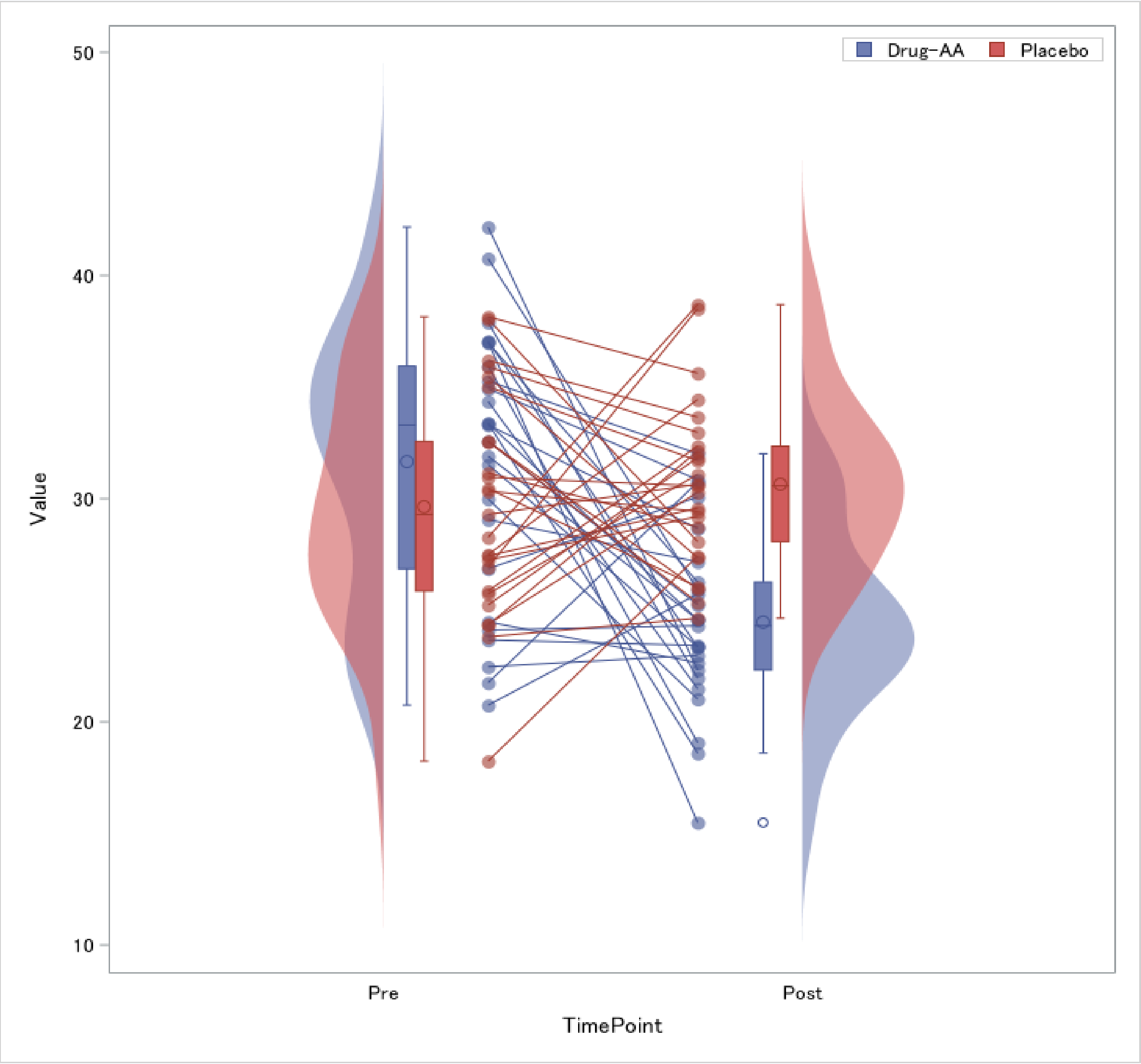
When the mouse cursor is hovered over the scatterplot, the data can be popped out according to the settings in rolename and tip. Useful for tracing each data.

Raincloud plot for Repeated Measure



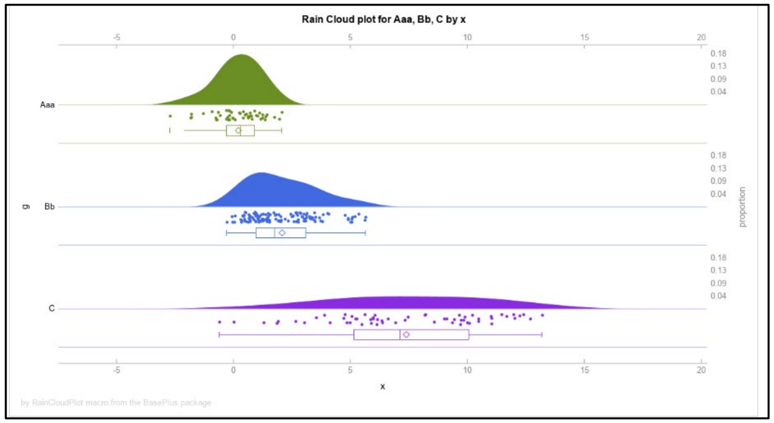
Paired Raincloud plot

```
.
.
.
bandplot y=value  limitupper=-0.2  limitlower=density1
/ group=TREAT  fillattrs=(transparency=0.4) tip=none;
bandplot y=value  limitupper=density2  limitlower=0.2
/ group=TREAT  fillattrs=(transparency=0.4) tip=none;
boxplot y = val x =box_x
/boxwidth = 1 group=TREAT groupdisplay=cluster
clusterwidth=0.1 name="box";
seriesplot x=series_x y=val
/display=all group=TREAT_ID
markerattrs=(symbol=circlefilled size=8 transparency=0.4)
lineattrs=(thickness=1 pattern=solid )
linecolorgroup=TREAT markercolorgroup=TREAT;
.
.
.
```

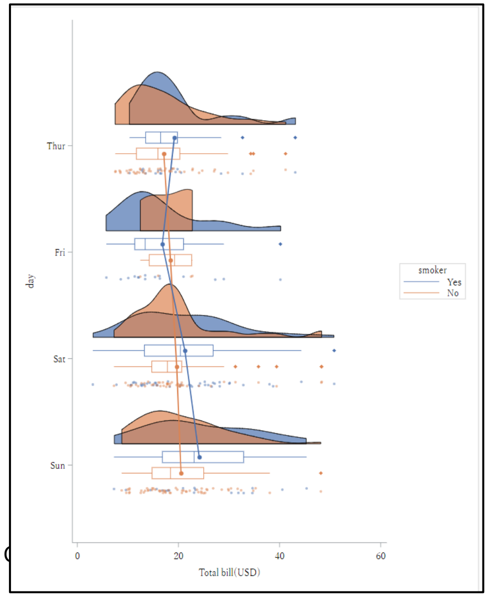


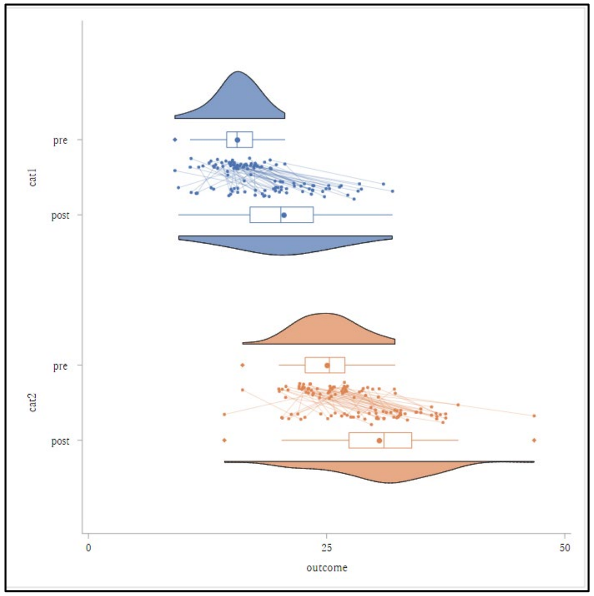
Programs already released by SAS users

Bart Jablonski, “The BasePlus package [ver. 1.17]”, May .12, 2022. https://github.com/yabwon/SAS_PACKAGES/blob/main/packages/baseplus.md#raincloudplot-macro. (Accessed Dec 05, 2022)



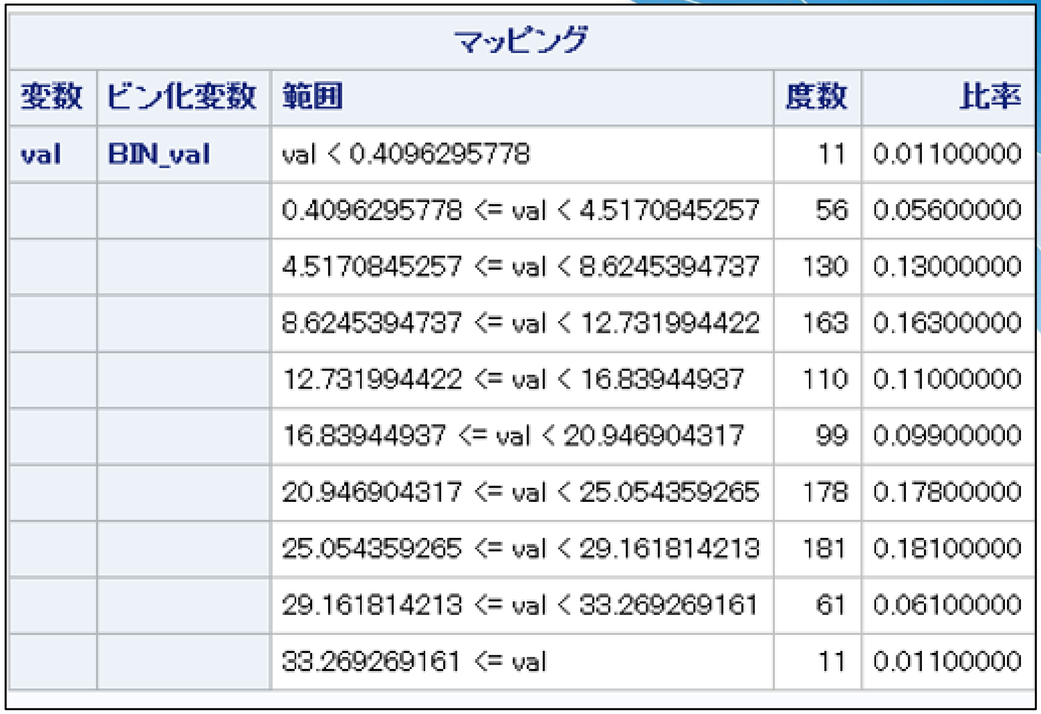
SupermanJP, “SAS_Plotter”, Jul .3, 2022.
https://github.com/Superman-jp/SAS_Plotter (Accessed Dec 05, 2022)





Binning

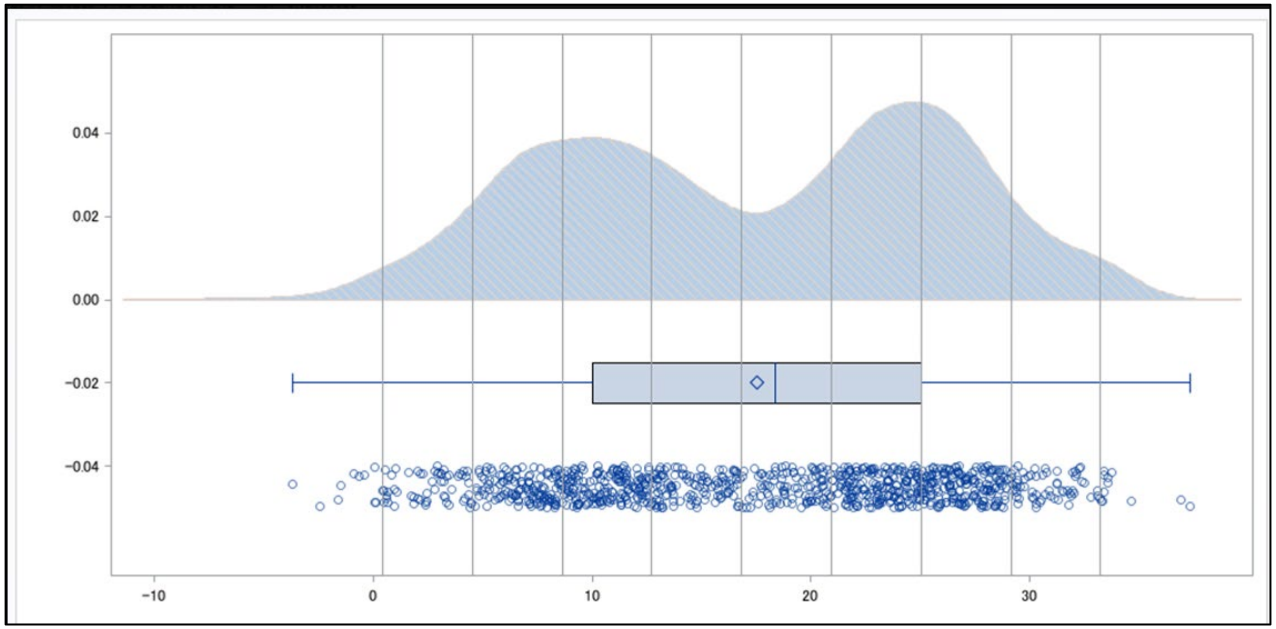
```
ods output mapping=bin;
proc hpbin
data=test numbin=10  bucket   ;
   input val;
 run;
```



マッピング				
変数	ビン化変数	範囲	度数	比率
val	BIN_val	val < 0.4096295778	11	0.01100000
		0.4096295778 <= val < 4.5170845257	56	0.05600000
		4.5170845257 <= val < 8.6245394737	130	0.13000000
		8.6245394737 <= val < 12.731994422	163	0.16300000
		12.731994422 <= val < 16.83944937	110	0.11000000
		16.83944937 <= val < 20.946904317	99	0.09900000
		20.946904317 <= val < 25.054359265	178	0.17800000
		25.054359265 <= val < 29.161814213	181	0.18100000
		29.161814213 <= val < 33.269269161	61	0.06100000
		33.269269161 <= val	11	0.01100000

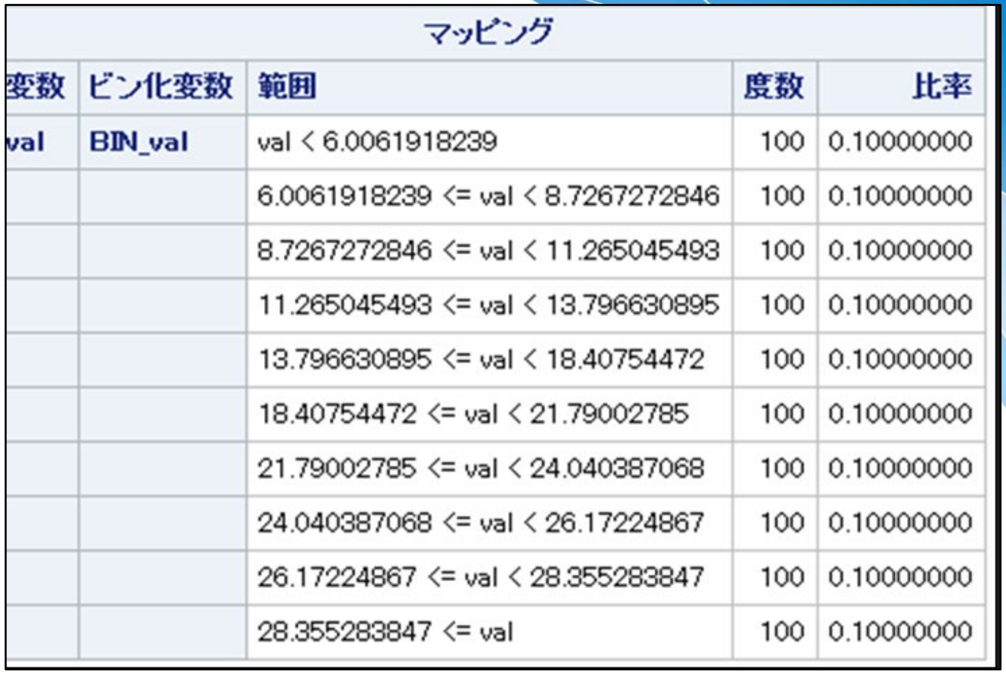
QUANTILE Binning

(Max – Min) / (number of bin)



Binning

```
ods output mapping=bin;
proc hpbin data=test numbin=10 quantile   ;
   input val;
 run;
```

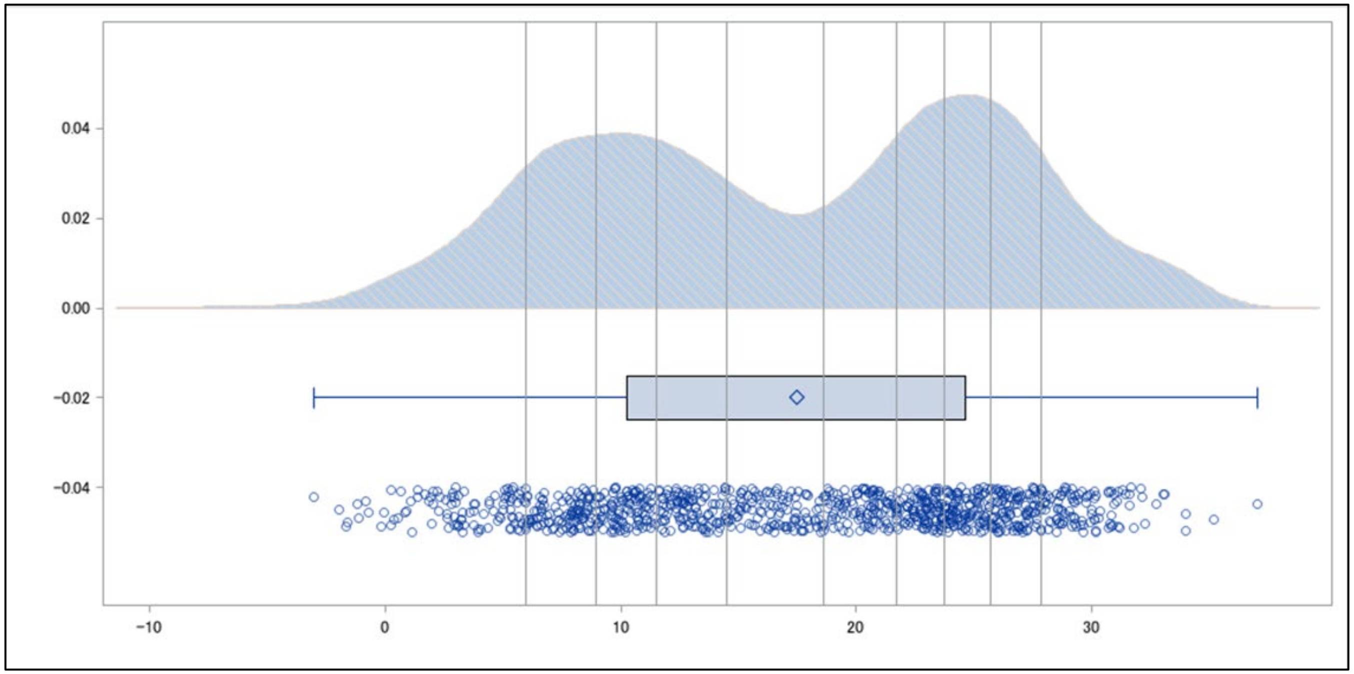


マッピング

変数	ビン化変数	範囲	度数	比率
val	BIN_val	val < 6.0061918239	100	0.10000000
		6.0061918239 <= val < 8.7267272846	100	0.10000000
		8.7267272846 <= val < 11.265045493	100	0.10000000
		11.265045493 <= val < 13.796630895	100	0.10000000
		13.796630895 <= val < 18.40754472	100	0.10000000
		18.40754472 <= val < 21.79002785	100	0.10000000
		21.79002785 <= val < 24.040387068	100	0.10000000
		24.040387068 <= val < 26.17224867	100	0.10000000
val		26.17224867 <= val < 28.355283847	100	0.10000000
		28.355283847 <= val	100	0.10000000

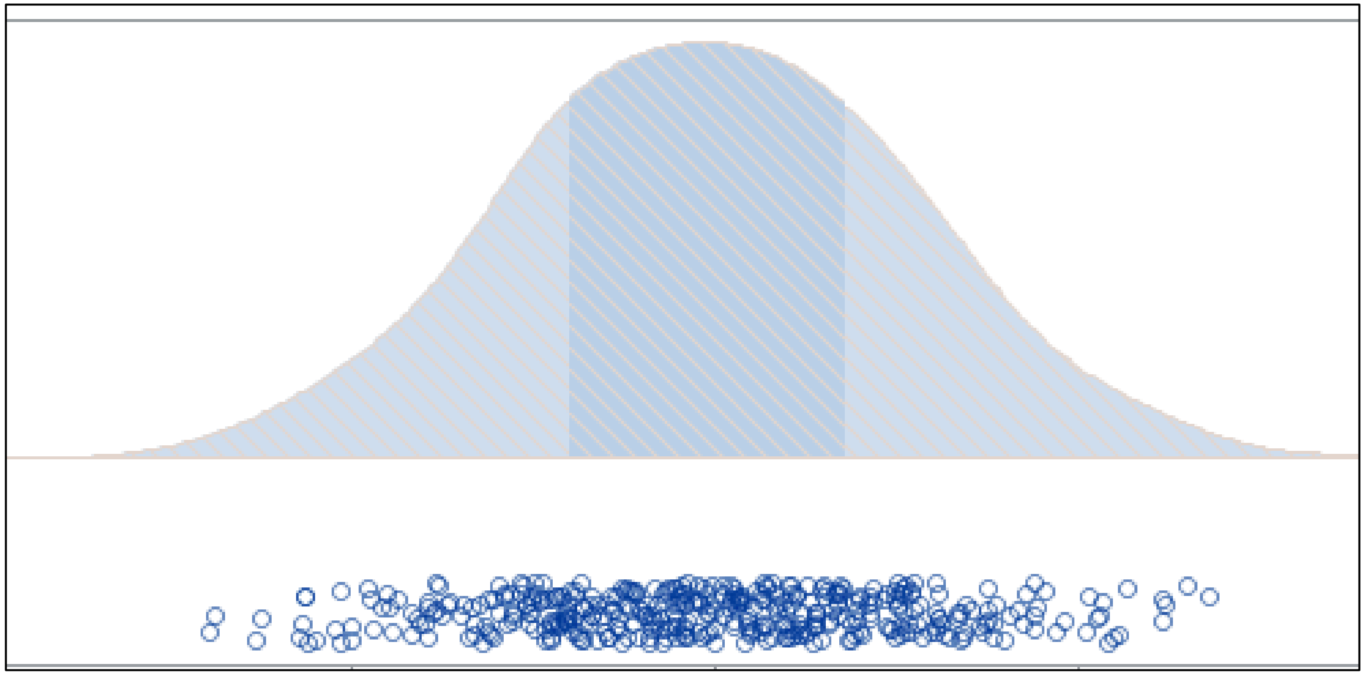
Quantile Binning

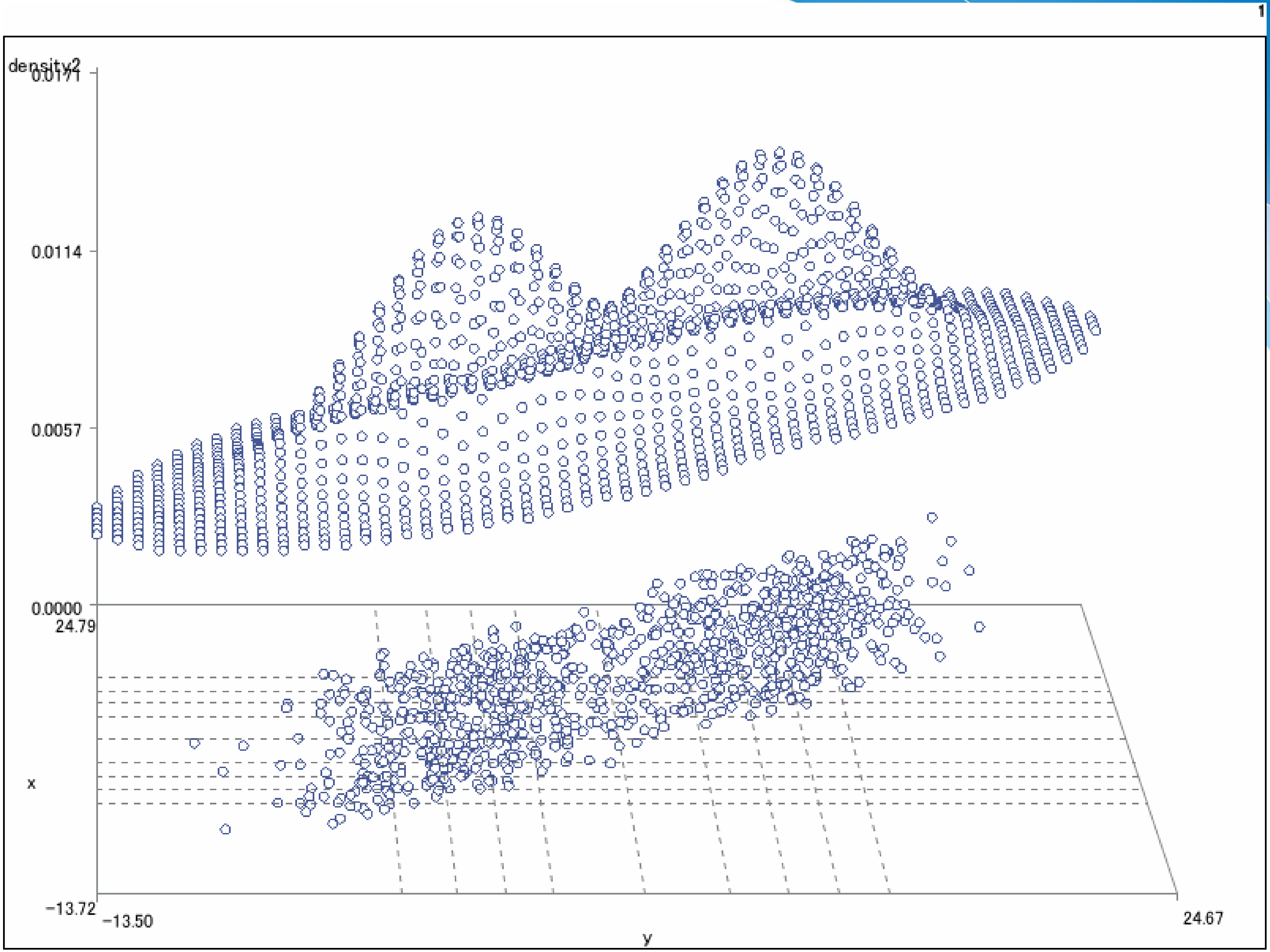
Use Quantile



Faded raincloud plots

```
bandplot x = value1 limitlower = 0 limitupper = density
               / display = all fillattrs=(transparency=0.3);
bandplot x = value2 limitlower = 0 limitupper = density
               / display = all fillattrs=(transparency=0);
```




1
density2
0.0171
0.0114
0.0057
0.0000
24.79
x
−13.72
−13.50
y
24.67

Bivariable Raincloud plot

```
proc kde data=xxx ;
  bivar  Var1 Var2/ out=kde;
run;
```

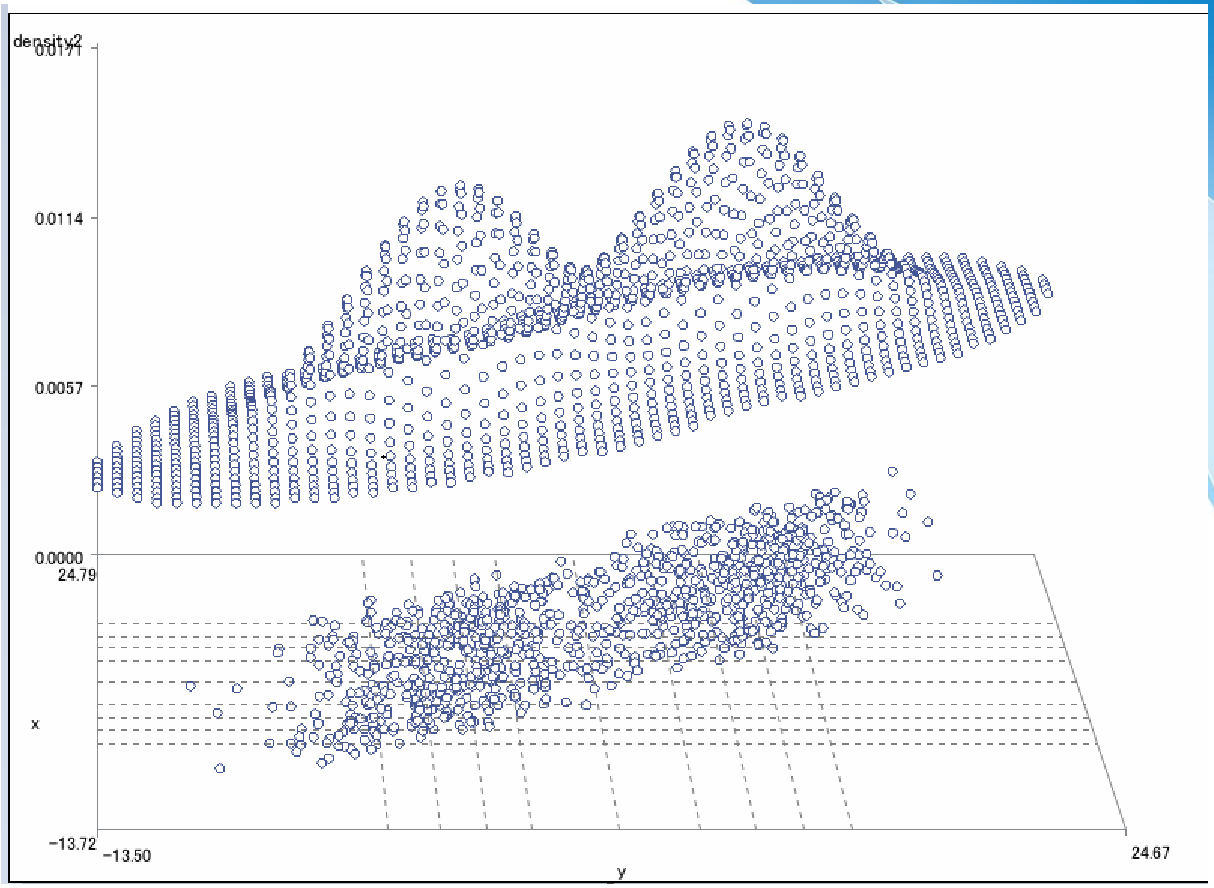
x =var1

y =var2

z= kde

Rain z-axis is dummy adjustment with random

Located on the bottom is Quantile binning(10)



All code:

http://sas-tumesas.blogspot.com/2022/10/23d-raincloud-plot.html

REFERENCES

[1] Micah Allen, Davide Poggiali, Kirstie Whitaker, Tom Rhys Marshall, Rogier Kievit,
“Raincloud plots: a multi-platform tool for robust data visualization” , PeerJ, Aug .23, 2018
https://peerj.com/preprints/27137/ (Accessed Dec 05, 2022)

[2] Micah Allen, Davide Poggiali, Kirstie Whitaker, Tom Rhys Marshall, Jordy van Langen, Rogier A. Kievit,
“Raincloud plots: a multi-platform tool for robust data visualization” , Wellcome Open Research, Jan .21 ,2021
https://wellcomeopenresearch.org/articles/4-63 (Accessed Dec 05, 2022)

[3] Bart Jablonski, “The BasePlus package [ver. 1.17]” , May .12, 2022.
https://github.com/yabwon/SAS_PACKAGES/blob/main/packages/baseplus.md#raincloudplot-macro.
(Accessed Dec 05, 2022)

[4] SupermanJP, “SAS_Plotter” , Jul .3, 2022.
https://github.com/Superman-jp/SAS_Plotter (Accessed Dec 05, 2022)

Thank you for your attention