Supplementary figures

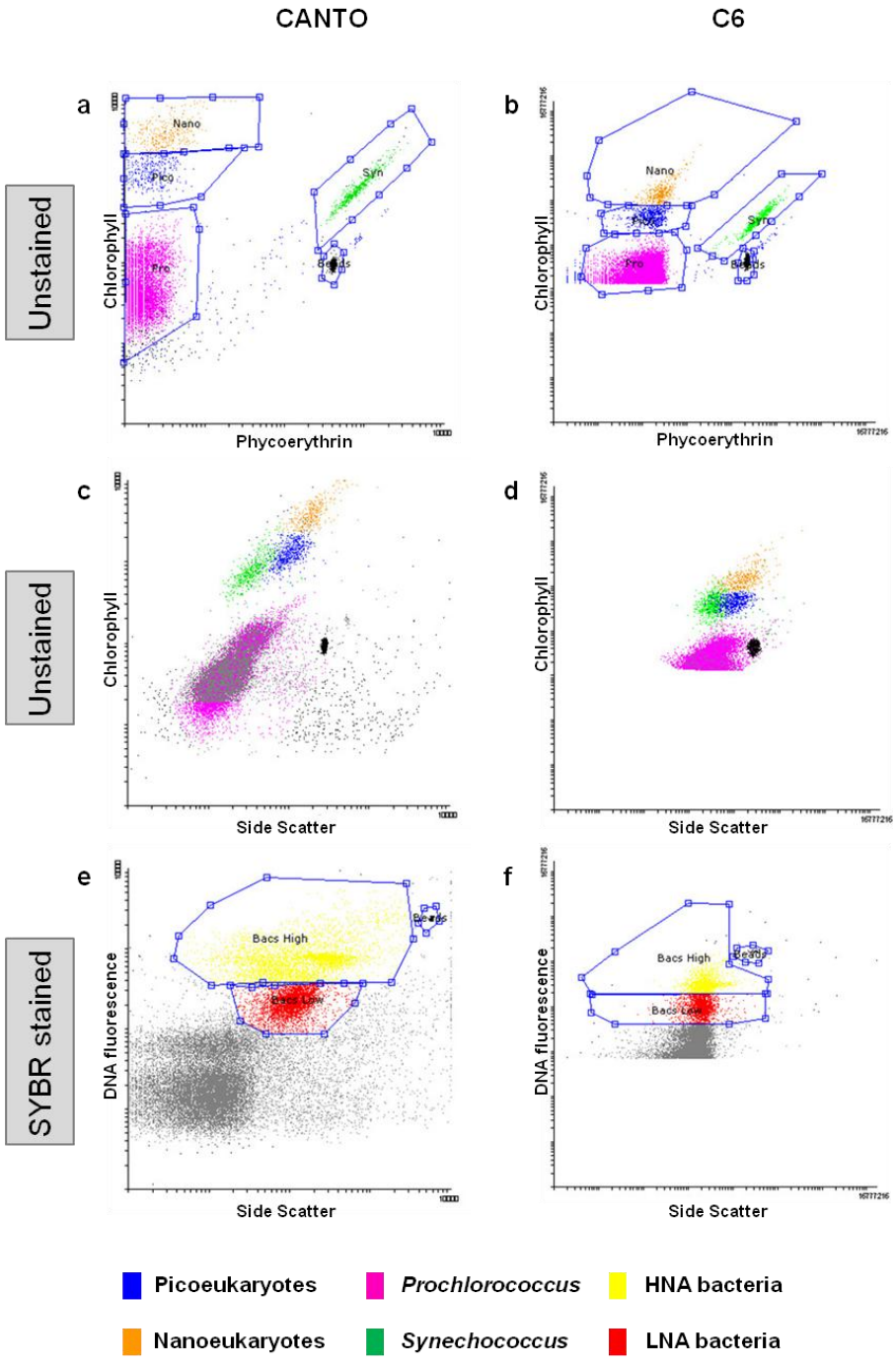

Figure S1. Cytograms of phycoerythrin *versus* chlorophyll fluorescence (**A, B**), side scatter *versus* chlorophyll fluorescence (**C, D**) and side scatter *versus* DNA fluorescence (**E, F**) for sample 137 (St. 100, 110 meters depth) for BD FACSCanto™ and BD Accuri™ C6 analyses, showing the gating windows: *Prochlorococcus* (pink), *Synechococcus* (green), picoeukaryotes (blue) nanoeukaryotes (orange), HNA bacteria (yellow) and LNA bacteria (red). Calibrations beads are marked in black.

Supplementary material

File S1. Example of input file for R routine.

channel	sample135_C6_PRO_5m	sample136_C6_PRO_50m	sample137_C6_PRO_110m	sample138_C6_PRO_130m	sample139_C6_PRO_170m
1	0	0	0	0	0
2	0	0	0	0	0
3	0	0	0	0	0
4	0	0	0	0	0
5	0	0	0	0	0
6	0	0	0	0	0
7	0	0	0	0	0
8	0	0	0	0	0
9	0	0	0	0	0
10	0	0	0	0	0
11	0	0	0	0	0
12	0	0	0	0	0
13	0	0	0	0	0
14	0	0	0	0	0
15	0	0	0	0	0
16	0	0	0	0	0
17	0	0	0	0	0
18	0	0	0	0	0
19	0	0	0	0	0
20	0	0	0	0	0
21	0	0	0	0	0
22	0	0	0	0	0
23	0	0	0	0	0
24	0	0	0	0	0
25	0	0	0	0	0
26	0	0	0	0	0
27	0	0	0	0	0
28	0	0	0	0	0
29	0	0	0	0	0
30	0	0	0	0	0
31	0	0	0	0	0
32	0	0	0	0	0
33	0	0	0	0	0
34	0	0	0	0	0
35	0	0	0	0	0
36	0	0	0	0	0
37	0	0	0	0	0
38	0	0	0	0	0

39	0	0	0	0	0
40	0	0	0	0	0
41	0	0	0	0	0
42	0	0	0	0	0
43	0	0	0	0	0
44	0	0	0	0	0
45	0	0	0	0	0
46	0	0	0	0	0
47	0	0	0	0	0
48	0	0	0	0	0
49	0	0	0	0	0
50	0	0	0	0	0
51	0	0	0	0	0
52	0	0	0	0	0
53	0	0	0	0	0
54	0	0	0	0	0
55	0	0	0	0	0
56	0	0	0	0	0
57	0	0	0	0	0
58	0	0	0	0	0
59	0	0	0	0	0
60	0	0	0	0	0
61	0	0	0	0	0
62	0	0	0	0	0
63	0	0	0	0	0
64	0	0	0	0	0
65	0	0	0	0	0
66	0	0	0	0	0
67	0	0	0	0	0
68	0	0	0	0	0
69	0	0	0	0	0
70	0	0	0	0	0
71	0	0	0	0	0
72	0	0	0	0	0
73	0	0	0	0	0
74	0	0	0	0	0
75	0	0	0	0	0
76	0	0	0	0	0
77	0	0	0	0	0
78	0	0	0	0	0
79	0	0	0	0	0
80	0	0	0	0	0
81	0	0	0	0	0

82	0	0	0	0	0
83	0	0	0	0	0
84	0	0	0	0	0
85	0	0	0	0	0
86	0	0	0	0	0
87	0	0	0	0	0
88	0	0	0	0	0
89	0	0	0	0	0
90	0	0	0	0	0
91	0	0	0	0	0
92	0	0	0	0	0
93	0	0	0	0	0
94	0	0	0	0	0
95	0	0	0	0	0
96	0	0	0	0	0
97	0	0	0	0	0
98	0	0	0	0	0
99	0	0	0	0	0
100	0	0	0	0	0
101	0	0	0	0	0
102	0	0	0	0	0
103	0	0	0	0	0
104	0	0	0	0	0
105	0	0	0	0	0
106	0	0	0	0	0
107	0	0	0	0	0
108	0	0	0	0	0
109	0	0	0	0	0
110	0	0	0	0	0
111	0	0	0	0	0
112	0	0	0	0	0
113	0	0	0	0	0
114	0	0	0	0	0
115	0	0	0	0	0
116	0	0	0	0	0
117	0	0	0	0	0
118	0	0	0	0	0
119	0	0	0	0	0
120	0	0	0	0	0
121	0	0	0	0	0
122	0	0	0	0	0
123	0	0	0	0	0
124	0	0	0	0	0

125	0	0	0	0	0
126	0	0	0	0	0
127	0	0	0	0	0
128	0	0	0	0	0
129	0	0	0	0	0
130	0	0	0	0	0
131	0	0	0	0	0
132	0	0	0	0	0
133	0	0	0	0	0
134	0	0	0	0	0
135	0	0	0	0	0
136	0	0	0	0	0
137	0	0	0	0	0
138	0	0	0	0	0
139	0	0	0	0	0
140	0	0	0	0	0
141	0	0	0	0	0
142	0	0	0	0	0
143	0	0	0	0	0
144	0	0	0	0	0
145	0	0	0	0	0
146	0	0	0	0	0
147	0	0	0	0	0
148	0	0	0	0	0
149	0	0	0	0	0
150	0	0	0	0	0
151	0	0	0	0	0
152	0	0	0	0	0
153	0	0	0	0	0
154	0	0	0	0	0
155	0	0	0	0	0
156	0	0	0	0	0
157	0	0	0	0	0
158	0	0	0	0	0
159	0	0	0	0	0
160	0	0	0	0	0
161	0	0	0	0	0
162	0	0	0	0	0
163	0	0	0	0	0
164	0	0	0	0	0
165	0	0	0	0	0
166	0	0	0	0	0
167	0	0	0	0	0

168	0	0	0	0	0
169	0	0	0	0	0
170	0	0	0	0	0
171	0	0	0	0	0
172	0	0	0	0	0
173	0	0	0	0	0
174	0	0	0	0	0
175	0	0	0	0	0
176	0	0	0	0	0
177	0	0	0	0	0
178	0	0	0	0	0
179	0	0	0	0	0
180	0	0	0	0	0
181	0	0	0	0	0
182	0	0	0	0	0
183	0	0	0	0	0
184	0	0	0	0	0
185	0	0	0	0	0
186	0	0	0	0	0
187	0	0	0	0	0
188	0	0	0	0	0
189	0	0	0	0	0
190	0	0	0	0	0
191	0	0	0	0	0
192	0	0	0	0	0
193	0	0	0	0	0
194	0	0	0	0	0
195	0	0	0	0	0
196	442	440	303	0	0
197	1104	1055	963	0	0
198	1647	1615	1519	9	0
199	1664	1604	1696	203	0
200	1385	1362	1440	545	0
201	1158	1123	1241	851	0
202	933	939	1048	882	0
203	748	760	898	716	0
204	590	591	745	543	0
205	479	474	661	421	0
206	376	362	566	322	0
207	286	283	529	239	0
208	203	198	494	184	0
209	147	149	488	135	0
210	103	102	478	100	0

211	80	77	463	71	0
212	56	56	475	53	0
213	43	41	486	36	0
214	35	30	491	22	0
215	24	21	489	16	0
216	16	12	492	13	1
217	10	7	488	12	6
218	9	5	501	12	9
219	7	5	491	9	11
220	6	6	509	7	10
221	3	5	507	6	10
222	2	4	511	7	9
223	2	3	491	7	8
224	3	3	472	5	8
225	2	3	450	5	9
226	2	2	435	5	8
227	1	1	408	8	9
228	0	0	384	9	8
229	0	0	357	11	9
230	1	0	343	10	7
231	2	1	330	11	8
232	2	1	302	11	8
233	1	1	271	14	8
234	1	0	237	16	6
235	1	1	224	18	6
236	1	0	205	17	6
237	0	0	190	20	7
238	1	0	180	22	7
239	1	0	175	31	7
240	1	0	166	34	7
241	0	0	160	35	8
242	0	0	156	43	11
243	0	0	157	48	14
244	0	0	149	58	16
245	0	0	140	59	20
246	0	0	143	66	22
247	0	0	148	71	24
248	0	0	148	84	25
249	0	0	142	91	33
250	0	0	134	94	40
251	0	0	136	101	46
252	0	0	131	112	48
253	0	0	124	126	54

254	0	0	112	130	56
255	0	0	107	126	67
256	0	0	107	127	68
257	0	0	106	127	77
258	0	0	97	125	75
259	0	0	85	121	74
260	0	0	68	114	75
261	0	0	56	120	73
262	0	0	51	117	79
263	0	0	48	123	75
264	0	0	46	108	73
265	0	0	39	107	71
266	0	0	34	93	68
267	0	0	27	89	64
268	0	0	19	75	61
269	0	0	13	73	60
270	0	0	10	73	59
271	0	0	11	65	55
272	0	0	10	56	48
273	0	0	8	43	43
274	0	0	6	39	36
275	0	0	5	35	36
276	0	0	3	27	32
277	0	0	3	26	28
278	0	0	3	18	22
279	0	0	3	17	18
280	0	0	3	13	20
281	0	0	2	11	18
282	0	0	3	9	18
283	0	0	3	6	13
284	0	0	4	5	10
285	0	0	3	5	7
286	0	0	2	5	9
287	0	0	2	5	10
288	0	0	1	3	12
289	0	0	2	2	8
290	0	0	1	2	6
291	0	0	1	3	4
292	0	0	1	3	4
293	0	0	0	2	3
294	0	0	0	1	2
295	0	0	0	1	1
296	0	0	0	1	2

297	0	0	0	1	2
298	0	0	0	0	1
299	0	0	0	0	2
300	0	0	0	0	3
301	0	0	0	0	3
302	0	0	0	0	2
303	0	0	0	0	1
304	0	0	0	0	1
305	0	0	0	0	2
306	0	0	0	0	1
307	0	0	0	0	0
308	0	0	0	0	0
309	0	0	0	0	0
310	0	0	0	0	0
311	0	0	0	0	0
312	0	0	0	0	0
313	0	0	0	0	0
314	0	0	0	0	0
315	0	0	0	0	0
316	0	0	0	0	0
317	0	0	0	0	0
318	0	0	0	0	0
319	0	0	0	0	0
320	0	0	0	0	0
321	0	0	0	0	0
322	0	0	0	0	0
323	0	0	0	0	0
324	0	0	0	0	0
325	0	0	0	0	0
326	0	0	0	0	0
327	0	0	0	0	0
328	0	0	0	0	0
329	0	0	0	0	0
330	0	0	0	0	0
331	0	0	0	0	0
332	0	0	0	0	0
333	0	0	0	0	0
334	0	0	0	0	0
335	0	0	0	0	0
336	0	0	0	0	0
337	0	0	0	0	0
338	0	0	0	0	0
339	0	0	0	0	0

340	0	0	0	0	0
341	0	0	0	0	0
342	0	0	0	0	0
343	0	0	0	0	0
344	0	0	0	0	0
345	0	0	0	0	0
346	0	0	0	0	0
347	0	0	0	0	0
348	0	0	0	0	0
349	0	0	0	0	0
350	0	0	0	0	0
351	0	0	0	0	0
352	0	0	0	0	0
353	0	0	0	0	0
354	0	0	0	0	0
355	0	0	0	0	0
356	0	0	0	0	0
357	0	0	0	0	0
358	0	0	0	0	0
359	0	0	0	0	0
360	0	0	0	0	0
361	0	0	0	0	0
362	0	0	0	0	0
363	0	0	0	0	0
364	0	0	0	0	0
365	0	0	0	0	0
366	0	0	0	0	0
367	0	0	0	0	0
368	0	0	0	0	0
369	0	0	0	0	0
370	0	0	0	0	0
371	0	0	0	0	0
372	0	0	0	0	0
373	0	0	0	0	0
374	0	0	0	0	0
375	0	0	0	0	0
376	0	0	0	0	0
377	0	0	0	0	0
378	0	0	0	0	0
379	0	0	0	0	0
380	0	0	0	0	0
381	0	0	0	0	0
382	0	0	0	0	0

383	0	0	0	0	0
384	0	0	0	0	0
385	0	0	0	0	0
386	0	0	0	0	0
387	0	0	0	0	0
388	0	0	0	0	0
389	0	0	0	0	0
390	0	0	0	0	0
391	0	0	0	0	0
392	0	0	0	0	0
393	0	0	0	0	0
394	0	0	0	0	0
395	0	0	0	0	0
396	0	0	0	0	0
397	0	0	0	0	0
398	0	0	0	0	0
399	0	0	0	0	0
400	0	0	0	0	0
401	0	0	0	0	0
402	0	0	0	0	0
403	0	0	0	0	0
404	0	0	0	0	0
405	0	0	0	0	0
406	0	0	0	0	0
407	0	0	0	0	0
408	0	0	0	0	0
409	0	0	0	0	0
410	0	0	0	0	0
411	0	0	0	0	0
412	0	0	0	0	0
413	0	0	0	0	0
414	0	0	0	0	0
415	0	0	0	0	0
416	0	0	0	0	0
417	0	0	0	0	0
418	0	0	0	0	0
419	0	0	0	0	0
420	0	0	0	0	0
421	0	0	0	0	0
422	0	0	0	0	0
423	0	0	0	0	0
424	0	0	0	0	0
425	0	0	0	0	0

426	0	0	0	0	0
427	0	0	0	0	0
428	0	0	0	0	0
429	0	0	0	0	0
430	0	0	0	0	0
431	0	0	0	0	0
432	0	0	0	0	0
433	0	0	0	0	0
434	0	0	0	0	0
435	0	0	0	0	0
436	0	0	0	0	0
437	0	0	0	0	0
438	0	0	0	0	0
439	0	0	0	0	0
440	0	0	0	0	0
441	0	0	0	0	0
442	0	0	0	0	0
443	0	0	0	0	0
444	0	0	0	0	0
445	0	0	0	0	0
446	0	0	0	0	0
447	0	0	0	0	0
448	0	0	0	0	0
449	0	0	0	0	0
450	0	0	0	0	0
451	0	0	0	0	0
452	0	0	0	0	0
453	0	0	0	0	0
454	0	0	0	0	0
455	0	0	0	0	0
456	0	0	0	0	0
457	0	0	0	0	0
458	0	0	0	0	0
459	0	0	0	0	0
460	0	0	0	0	0
461	0	0	0	0	0
462	0	0	0	0	0
463	0	0	0	0	0
464	0	0	0	0	0
465	0	0	0	0	0
466	0	0	0	0	0
467	0	0	0	0	0
468	0	0	0	0	0

469	0	0	0	0	0
470	0	0	0	0	0
471	0	0	0	0	0
472	0	0	0	0	0
473	0	0	0	0	0
474	0	0	0	0	0
475	0	0	0	0	0
476	0	0	0	0	0
477	0	0	0	0	0
478	0	0	0	0	0
479	0	0	0	0	0
480	0	0	0	0	0
481	0	0	0	0	0
482	0	0	0	0	0
483	0	0	0	0	0
484	0	0	0	0	0
485	0	0	0	0	0
486	0	0	0	0	0
487	0	0	0	0	0
488	0	0	0	0	0
489	0	0	0	0	0
490	0	0	0	0	0
491	0	0	0	0	0
492	0	0	0	0	0
493	0	0	0	0	0
494	0	0	0	0	0
495	0	0	0	0	0
496	0	0	0	0	0
497	0	0	0	0	0
498	0	0	0	0	0
499	0	0	0	0	0
500	0	0	0	0	0

405

File S2. R routine to correct abundance when populations are partly in noise.

R code

The code below describes how to implement an R routine to correct the abundance of picoplanktonic populations based on their red fluorescence distribution. All libraries used here are freely available from R repositories. The input file used in this examples is named as Pro_C6.txt (See input file example File S1). This file has been created by exporting FL3 (chlorophyll) histogram from the Flowing Software (http://www.flowingsoftware.com) combining different samples into a single file. The first column contains the channel number and each following column corresponds to a different sample with rows corresponding to cell counts in each channel. Such a file could be created with any flow cytometry software. After running the `cyto_plot` function, a pdf output file is created named "Pro_C6.txt 1.0 .pdf" which contains all histograms from the input file (see File S3) and the file statistics (sample, uncorrected and corrected total cell abundance) are available as a data frame in the R session (see example at bottom of this file)

```
# Example of use of cyto_plot function (run first the R code below to define the necessary functions)
stats_Pro_C6<-cyto_plot("Pro_C6.txt", decades_C6, channel_min_C6, xmin_C6, xmax_C6)

# Example of statistics output

  sample                 cell_tot       cell_tot_correc
1 sample135_C6_PRO_5m    134            cells in noise
2 sample136_C6_PRO_50m   111            cells in noise
3 sample137_C6_PRO_110m  13072          20240
4 sample138_C6_PRO_130m  3598           no correction
5 sample139_C6_PRO_170m  2211           no correction
```

R code

```
# Install libraries
library("ggplot2")
library("reshape2")
library("plyr")
library("scales")
require(grid)

# Set the working directory where the files are located
setwd ("C:/My Documents/cytometry data/")

# Define basic parameters
decades_Canto = 5
decades_C6 = 7
channel_min_Canto = 100
channel_min_C6 = 214
xmin_Canto = 10
xmin_C6 = 1000
xmax_Canto = 10000
xmax_C6 = 100000
channel_max = 500
point <- format_format(big.mark = "", decimal.mark = ".", scientific = TRUE)
# ----------------------------------------------------------
# cell_correct(channel, cell_number, cell_smooth)
# Arguments
#       channel : vector containing the channels (from 1 to 500 in the present case)
#       cell_number : vector containing cell abundance in each channel
#       cell_smooth : vector containing smoothed cell abundance in each channel
# Description
# This function determines in which case we are ("no correction", "cells in noise" or "correction")and
return the corrected cell abundance in the latter case.

cell_correct<-function(channel, cell_number, cell_smooth)
{ df<-data.frame(channel, cell_number, cell_smooth)  # create a data frame
i_min<-which.min(channel)                            # determine the minimum channel
i_max<-which.max(channel)                            # determine the maximum channel
i_cell_max<-which.max(cell_smooth)                   # determine in which channel is the histogram mode
```

```
# "no correction" : cell abundance in the first channel is 5 times lowerthan abundance at the maximum
of the histogram
if (cell_smooth[i_cell_max]>5*cell_smooth[i_min])    {cell_correct<-"no correction"}
# "cells in noise" : maximum of cell abundance is in the first channel
  else {if (i_cell_max==i_min)
        {cell_correct<-"cells in noise"}
# "correction" : all the other cases, we then apply a correction by computing the total cell abundance
as twice the number of cells in the channels right of the histogram maximum
  else
        {cell_correct<-2*sum(cell_number[i_cell_max:i_max])}
        }
  return (cell_correct)
}

# ------------------------------------------------------
#cyto_plot(file_name,decades,channel_min,xmin,xmax)
# Arguments
#       file_name : name of input file containing the different samples (see File S1)
#       decades : number of logarithmic decades of the flow cytometer (e.g. 7 for C6)
#       channel_min : threshold channel for the histogram (depends on fcm acquisition settings)
#       xmin : linear value corresponding to the threshold channel
#       xmax : linear value corresponding to the maximum channel
# Description
# This function plotsa set of histograms for the input samples,saves the graphics as a pdf file and
compute the total cell abundance indicating whether a corrections is needed or not.  It returns a
dataframe containing three columns : sample, cell_tot, cell_tot_correc (see top of this file for an
example)

cyto_plot<-function(file_name,decades,channel_min,xmin,xmax)
{       channel_max = 500 # this the number of channels provided as output of the Flowing Software
        histo<- read.delim(file_name)
        histo<- histo[histo$channel>=channel_min,]
        histo_melt<-  melt(histo, id.vars=c("channel"),variable.name = "sample", value.name =
"cell_number")
```

```
# smooth histogram using default R smoothing function
        histo_melt<- ddply(histo_melt,c("sample"), transform,
cell_smooth=as.vector(smooth(cell_number)))
# normalize histogram so that maximum abundance is equal to 1
        histo_melt<- ddply(histo_melt,c("sample"), transform, cell_norm=cell_smooth/max(cell_smooth))
# transform log channel to linear scale for plotting
        if (decades==5)
                {histo_melt<- ddply(histo_melt,c("sample"), transform,
fluo=(10^5)^(channel/channel_max))}
        else
                {histo_melt<- ddply(histo_melt,c("sample"), transform,
fluo=(10^7)^(channel/channel_max))}
# plots histograms using 5 columns
        histo_plot<-ggplot(histo_melt, aes(fluo,cell_norm)) + geom_line() + theme_bw () + facet_wrap(~
sample, nrow=21, ncol=5) + xlab("Chlorophyll")+ylab("Relative cell number") +
scale_x_log10(limits=c(xmin,xmax), labels=point)
# save plots as pdf
ggsave(plot=histo_plot, filename=paste(file_name," 1.0 .pdf",sep=""),width = 15, height = 4, scale=2,
units="cm")
# compute uncorrected and corrected total cell number calling the cell_correct function defined above
        stats<-ddply(histo_melt,c("sample"),summarise,
cell_tot=sum(cell_number),cell_tot_correc=cell_correct(channel,cell_number,cell_smooth))
        print(paste("# of decades:",decades,"minimum channel : ",channel_min, "xmin : ",   xmin, " xmax
: ",    xmax))
        print (paste("File : ",file_name))
        stats
        return (stats)
}
```

File S3. Example of output file for R routine.