

S4 Class trajectories

Basis of object programming

```
setClass(Class="Trajectories",
  representation=representation(
    times = "numeric",
    traj = "matrix"
  ) )
```

4.2 Default constructor

```
new(Class="Trajectories")
```

```
An object of class "Trajectories"
Slot "times":
numeric(0)
```

```
Slot "traj":
<0 x 0 matrix>
```

As you can note, the result is not easy to read... It will be important to define a method to improve it. We'll deal with that in the section 5.2 page 20. It is possible to define an object by specifying the values of its slots (all of them, or partly). We must each time specify its name of the concerning field:

```
new(Class="Trajectories",times=c(1,3,4))
```

```
An object of class "Trajectories"
Slot "times":
[1] 1 3 4
```

```
Slot "traj":
<0 x 0 matrix>
```

```
new(Class="Trajectories",times=c(1,3),traj=matrix(1:4,ncol=2))
```

```
An object of class "Trajectories"
Slot "times":
[1] 1 3
```

```
Slot "traj":
  [,1] [,2]
[1,]   1   3
[2,]   2   4
```

An object can be stored in a variable like any other value of R. To illustrate our statements, we are going to build up a small example. Three hospitals take part to the study. The Piti e Salp etriere (which has not yet returned its data file, shame on them!), Cochin and Saint-Anne:

```
trajPitie <- new(Class="Trajectories")
trajCochin <- new(Class="Trajectories",
  times=c(1,3,4,5),
  traj=rbind (
```

```

        c(15,15.1, 15.2, 15.2),
        c(16,15.9, 16,16.4),
        c(15.2, NA, 15.3, 15.3),
        c(15.7, 15.6, 15.8, 16)
    ))

trajStAnne <- new(Class= "Trajectories",
  times=c(1: 10, (6: 16) *2),
  traj=rbind(
    matrix (seq (16,19, length=21), ncol=21, nrow=50, byrow=TRUE),
    matrix (seq (15.8, 18, length=21), ncol=21, nrow=30, byrow=TRUE)
  )+rnorm (21*80,0,0.2)
)

```

Reach a slot

```
trajCochin@times
```

```
[1] 1 3 4 5
```

```
trajCochin@times <- c(1,2,4,5)
trajCochin
```

An object of class "Trajectories"

Slot "times":

```
[1] 1 2 4 5
```

Slot "traj":

```

  [,1] [,2] [,3] [,4]
[1,] 15.0 15.1 15.2 15.2
[2,] 16.0 15.9 16.0 16.4
[3,] 15.2  NA 15.3 15.3
[4,] 15.7 15.6 15.8 16.0

```

As we will see thereafter, the use of the @ should be avoided. Indeed, it does not call upon the methods of checking. The use that we present here (posting of a field, and even worse, assignment of a value to a field) should thus be proscribed in most cases.

It is also possible to use the functions attr or attributes, but it is even worse: if one makes a simple typographic error, one modifies the structure of the object. And that is very very very bad!

4.4 Default values

```

setClass(Class = "TrajectoriesBis",
  representation=representation(
    time = "numeric",
    traj = "matrix"
  ),
  prototype=prototype(
    time = 1,
    traj = matrix (0)
  ) )

```

The empty object

```
identical(numeric(),integer())
```

```
[1] FALSE
```

4.7 to see an object

```
slotNames("Trajectories")
```

```
[1] "times" "traj"
```

```
getSlots ("Trajectories")
```

```
      times      traj  
"numeric" "matrix"
```

```
getClass ("Trajectories")
```

```
Class "Trajectories" [in ".GlobalEnv"]
```

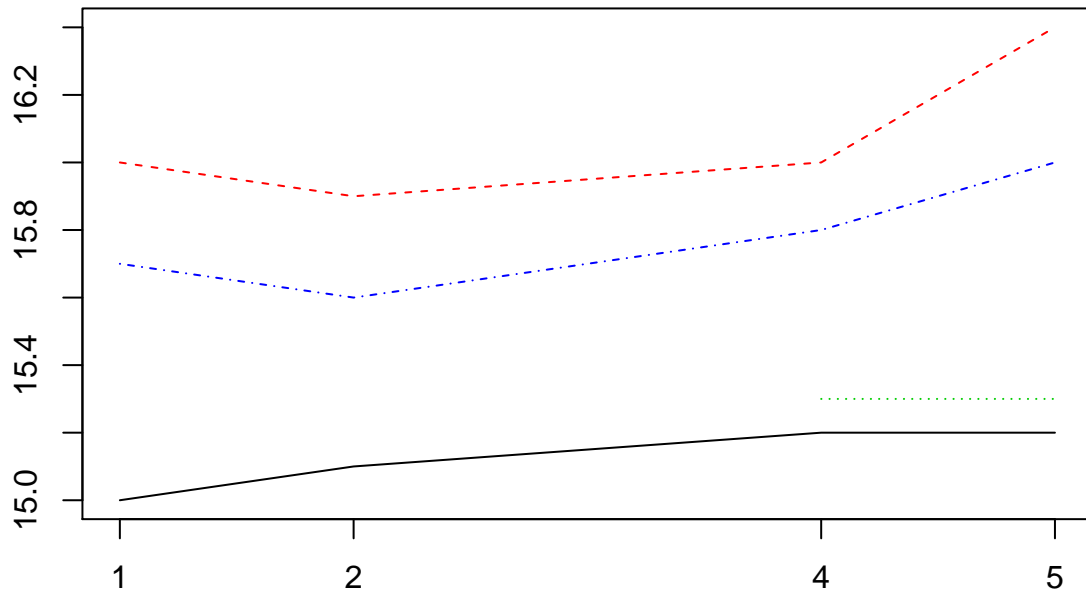
```
Slots:
```

```
Name:      times      traj  
Class: numeric matrix
```

SetMethod

```
setMethod("plot",  
  signature = "Trajectories", function (x, y,...) {  
    matplot(x@times,t(x@traj),xaxt="n",type="l",ylab= "", xlab="", pch=1)  
    axis(1,at=x@times)  
  })
```

```
plot(trajCochin)
```



```
setMethod ("print","Trajectories", function(x,...) {
  cat("*** Class Trajectories, method Print *** \n")
  cat("* Times ="); print (x@times)
  cat("* Traj = \n"); print (x@traj)
  cat("***** End Print (trajectories) ***** \n")
})
```

```
print(trajCochin)
```

```
*** Class Trajectories, method Print ***
* Times =[1] 1 2 4 5
* Traj =
  [,1] [,2] [,3] [,4]
[1,] 15.0 15.1 15.2 15.2
[2,] 16.0 15.9 16.0 16.4
[3,] 15.2  NA 15.3 15.3
[4,] 15.7 15.6 15.8 16.0
***** End Print (trajectories) *****
```

```
print(trajStAnne)
```

```
*** Class Trajectories, method Print ***
* Times = [1] 1 2 3 4 5 6 7 8 9 10 12 14 16 18 20 22 24 26 28 30 32
* Traj =
  [,1]      [,2]      [,3]      [,4]      [,5]      [,6]      [,7]
[1,] 15.94066 16.37060 15.97194 16.57515 16.92000 16.49317 17.07183
[2,] 15.84522 16.07448 15.90691 16.72437 16.61244 17.02844 17.50333
```

[3,] 15.68090 16.24215 16.17249 16.52928 16.44756 16.73344 17.12043
 [4,] 16.17541 15.83052 16.70172 16.32733 16.86571 16.94164 17.11331
 [5,] 15.99769 16.38370 16.19780 16.73180 16.44639 16.93544 16.96863
 [6,] 15.90100 16.38319 16.09291 16.14735 16.59090 16.70782 17.15678
 [7,] 15.90399 16.34898 16.45707 16.61896 16.80388 16.68634 16.95911
 [8,] 15.74246 16.32269 16.62443 16.23658 16.74780 16.89383 16.68129
 [9,] 16.03417 16.12953 16.14968 16.31820 16.39050 17.18055 16.80334
 [10,] 15.72257 15.82090 16.45464 16.64988 16.21688 16.62925 16.82722
 [11,] 15.89000 16.09472 16.17684 16.49700 16.56465 16.34589 17.14109
 [12,] 15.72978 16.32151 16.10500 16.32289 16.38130 16.61289 16.93287
 [13,] 15.60686 15.99770 16.23734 16.66425 16.32416 16.73829 17.22306
 [14,] 16.21937 16.19252 16.42501 16.51850 16.61301 17.04443 17.05956
 [15,] 16.29566 16.02280 16.51550 16.32608 16.81754 16.62980 16.79664
 [16,] 15.91393 16.47976 16.34813 16.42435 16.86298 16.62015 16.66390
 [17,] 15.61748 16.08828 15.88806 16.87747 16.23861 16.88505 16.77859
 [18,] 15.89124 16.13285 16.29412 16.43169 16.41677 16.76622 16.71144
 [19,] 15.55105 16.08394 16.45946 16.62987 16.68955 16.75451 17.37978
 [20,] 16.10151 15.97250 16.26310 16.45585 16.90156 16.72664 16.99532
 [21,] 15.58506 16.23791 16.45783 16.72973 16.43450 16.77397 16.73382
 [22,] 16.02212 16.09266 16.51944 16.44828 16.49177 16.57225 17.14516
 [23,] 15.77889 15.96814 16.47653 16.25557 16.78739 17.06443 16.77319
 [24,] 15.92242 16.16362 16.19926 16.52154 16.89758 16.64911 16.88417
 [25,] 15.98188 16.15358 16.03672 16.20997 16.36519 16.64089 16.90972
 [26,] 16.21492 16.04530 16.06381 16.25378 17.08601 16.77949 17.23548
 [27,] 16.03601 16.24276 16.23449 16.62890 16.57328 16.64895 17.00342
 [28,] 15.98925 16.30104 16.56103 16.35651 16.30480 16.59623 16.89108
 [29,] 16.14632 16.27262 16.03979 16.44323 16.97168 16.95325 16.92406
 [30,] 16.22899 16.53569 15.91684 16.41206 16.51429 16.66090 16.38314
 [31,] 15.91284 16.08441 16.27872 16.46069 16.61465 16.92630 16.80314
 [32,] 16.20698 16.26790 16.25665 16.58960 16.40170 16.81126 17.09043
 [33,] 16.40105 16.07359 16.34055 16.30570 16.43113 16.99920 16.45491
 [34,] 15.94303 16.11900 16.45910 16.30935 16.94515 16.79814 16.91216
 [35,] 16.04922 16.11154 16.10401 16.52501 16.77712 16.71921 17.26584
 [36,] 16.23021 16.48191 16.23797 16.45916 16.63444 16.62630 16.93358
 [37,] 15.88439 15.75610 16.29215 16.34679 16.47227 16.95070 17.25332
 [38,] 15.61698 16.06204 16.08909 16.24411 16.76071 16.92152 17.04819
 [39,] 16.10020 16.01345 16.04974 16.61379 16.54240 16.69230 17.02674
 [40,] 16.06543 16.06530 16.15585 16.29407 16.79616 16.95394 16.45958
 [41,] 15.56195 16.38726 16.14867 16.79661 16.48675 16.71197 16.95864
 [42,] 15.89138 15.94449 15.96778 16.57508 16.66292 16.73361 16.68906
 [43,] 16.06955 16.00506 16.65785 16.78769 17.16165 16.78539 16.96157
 [44,] 16.07143 16.12133 16.25007 16.13000 16.64006 16.69008 16.66109
 [45,] 16.21343 16.29961 16.16385 15.88180 16.54250 17.22377 16.76959
 [46,] 16.37449 16.13796 16.29028 16.43890 16.91205 16.78204 16.82161
 [47,] 16.00261 16.27611 15.94273 16.54831 16.73478 16.55975 17.05860
 [48,] 15.75461 16.28310 16.26936 16.45372 16.59497 16.89042 16.64450
 [49,] 16.13755 16.16421 16.74387 16.38175 16.45992 16.88754 16.84036
 [50,] 15.93768 16.70502 16.13052 16.76301 16.65180 16.63759 16.91849
 [51,] 15.33609 15.99861 15.87177 16.29034 16.32957 16.13995 16.52205
 [52,] 15.37708 16.03251 15.77491 16.00862 16.11181 16.20605 16.51250
 [53,] 15.96361 15.90164 15.86834 16.24907 16.16078 16.00546 16.28707
 [54,] 15.68715 15.51728 16.35213 16.60939 16.25922 16.34366 16.48723
 [55,] 15.51221 16.02811 16.01563 16.16768 16.09398 16.12610 16.35430
 [56,] 15.75520 16.05860 16.12971 15.94126 15.99040 16.34320 16.50387

[57,]	15.64732	15.77579	16.27846	16.10376	16.13912	16.68411	16.47917
[58,]	16.04479	16.24279	16.40783	16.22729	16.01706	16.31237	16.42717
[59,]	15.52863	15.91091	16.26704	16.16211	16.55745	16.48797	16.48813
[60,]	15.38678	15.77647	15.95259	15.93837	16.23937	16.33528	16.07129
[61,]	15.71879	15.92180	15.83570	15.99188	16.04356	16.17078	16.42266
[62,]	15.64521	16.30204	15.98720	15.79513	16.28750	16.41660	16.54536
[63,]	15.78009	15.63436	16.12201	15.79805	16.48069	16.43843	16.38963
[64,]	15.73799	16.08602	15.87095	16.32744	16.35268	16.33262	16.19015
[65,]	15.55596	15.93564	15.96166	16.22837	16.34002	16.09572	16.77339
[66,]	15.87507	16.13145	15.77354	15.97671	16.63226	16.31733	16.68313
[67,]	15.45644	15.90544	15.88096	16.29987	16.35458	16.43457	16.29192
[68,]	15.58365	16.04680	15.72489	16.27036	16.26764	15.91629	16.53119
[69,]	15.55734	15.94089	15.89880	15.93897	16.30168	16.23194	16.45116
[70,]	16.11436	15.75444	16.01801	16.26650	16.19541	16.48255	16.12262
[71,]	15.81722	15.90689	16.49289	16.13752	16.30950	16.47569	16.58739
[72,]	16.02008	15.48480	16.13047	16.04942	16.26443	16.46262	16.24720
[73,]	15.58021	15.75062	16.14582	16.24208	16.12618	16.31491	16.46374
[74,]	15.82746	15.75881	15.78366	16.14776	16.23730	16.40894	16.63669
[75,]	15.78631	15.91629	16.05117	16.44427	15.95297	16.28094	16.80142
[76,]	15.60614	15.78384	15.88938	16.36403	16.61373	16.48548	16.55102
[77,]	15.63404	15.66391	16.05014	15.99383	16.13414	16.53170	16.38578
[78,]	15.71780	15.87725	16.10227	16.23365	16.56348	16.09055	16.67873
[79,]	15.92158	15.70039	16.02549	15.96751	16.19238	16.65337	16.52557
[80,]	15.66035	15.49298	15.84145	16.00493	16.55368	16.58801	16.73786
	[,8]	[,9]	[,10]	[,11]	[,12]	[,13]	[,14]
[1,]	17.48396	17.50436	17.37876	17.22337	17.44530	17.66591	17.66483
[2,]	17.22441	17.39662	17.47697	17.74179	17.77420	17.57337	18.04315
[3,]	17.04639	17.63507	17.23200	17.18153	17.80713	17.71904	18.17542
[4,]	16.99044	16.98985	17.34749	17.59516	17.61780	17.76925	17.72011
[5,]	16.98703	17.40625	17.46990	17.28431	17.67014	17.68880	17.94435
[6,]	17.12607	17.39744	17.66958	17.34287	17.34776	17.87652	17.81975
[7,]	16.90492	17.29956	17.05062	17.69273	17.50388	17.89774	17.82991
[8,]	17.01645	17.36202	17.19036	17.43401	17.53496	17.84314	17.66207
[9,]	17.30576	17.11499	17.50078	17.45655	17.58321	17.73316	18.01914
[10,]	17.00625	17.38197	17.25701	17.43014	17.67582	18.24591	18.05322
[11,]	17.00074	17.01267	17.18055	17.65844	17.71279	17.65047	17.72668
[12,]	17.30006	17.25641	17.54219	17.93009	17.45494	18.17360	18.12703
[13,]	17.08255	17.38956	17.39682	17.57186	17.40982	17.58320	18.35211
[14,]	16.94723	17.44892	17.39400	17.53792	17.26202	17.79756	18.21017
[15,]	16.98094	17.20500	17.63632	17.71834	17.57051	17.69523	17.93068
[16,]	17.03759	17.24702	17.13786	17.24131	17.63211	17.82088	18.03466
[17,]	17.36543	17.28256	17.45038	17.08738	17.45937	17.93742	17.98668
[18,]	16.61303	17.06912	17.24816	17.85925	17.74064	18.17311	18.09871
[19,]	16.87798	17.41533	16.89435	17.68694	17.79631	18.04850	17.85163
[20,]	17.09716	17.14982	17.29030	17.48823	17.65088	17.59892	18.11739
[21,]	17.23961	17.11287	17.46941	17.28219	17.96477	17.86831	17.81406
[22,]	17.07890	17.09798	17.37359	17.56018	17.50365	17.49926	17.91037
[23,]	17.16110	17.07115	17.20768	17.31425	17.92802	17.71434	17.88143
[24,]	17.23561	16.78648	17.29964	17.52545	17.64717	17.77768	17.96929
[25,]	17.10625	17.24320	17.21591	17.66879	17.74000	17.70948	17.87576
[26,]	16.96188	17.11078	17.65362	17.37574	17.60982	17.92870	17.91262
[27,]	17.09760	17.26855	17.16030	17.35980	17.70990	18.02087	17.88607
[28,]	16.86845	17.49036	17.25652	17.29351	17.87811	17.78938	17.92709
[29,]	17.00371	17.07017	17.25309	17.48210	17.66539	17.68647	17.86177

[30,]	17.19410	17.30385	17.51684	17.48697	17.71513	17.55548	17.72931
[31,]	17.15201	17.16466	17.22261	17.24954	17.62076	17.48135	17.98332
[32,]	17.28595	17.30870	17.43400	17.57784	17.55427	17.63195	17.91096
[33,]	17.07974	17.21444	17.48103	17.48165	17.79387	18.04811	17.95618
[34,]	16.99507	16.93674	17.44995	17.47498	17.27262	17.90245	18.18082
[35,]	17.07366	17.27648	17.58601	17.34078	17.88823	17.77074	17.68469
[36,]	16.94326	17.23121	17.09636	17.29388	17.47831	17.79563	18.20750
[37,]	17.34256	17.39137	17.19035	17.92936	17.88974	17.84578	17.85267
[38,]	17.19342	17.31437	17.43461	17.37648	17.72329	17.65509	17.96061
[39,]	16.68574	17.18757	17.33733	17.56018	17.81491	17.36694	18.02593
[40,]	17.48398	17.13825	17.37041	17.25944	17.65336	17.94499	18.19934
[41,]	16.88452	17.17177	17.61102	17.09397	17.78087	17.80858	17.90379
[42,]	17.10856	17.03399	17.20827	17.60071	17.66175	17.92538	18.02282
[43,]	16.81492	17.19074	17.29314	17.51147	17.75506	17.90790	17.73228
[44,]	17.30778	16.91758	17.34784	17.76509	17.81853	17.91633	18.22720
[45,]	17.10049	17.00776	17.24252	17.32764	17.57746	17.74317	17.93138
[46,]	17.04906	17.32314	17.34232	17.81780	17.54573	17.87139	17.95367
[47,]	17.21766	16.94668	17.73538	17.43444	17.67104	17.72020	17.95209
[48,]	16.67385	17.28911	17.43107	17.64450	17.18738	17.96742	18.11133
[49,]	17.06402	17.10431	17.35137	17.32107	17.92150	17.89736	17.79085
[50,]	17.08328	17.18187	17.15348	17.21469	17.71139	17.83425	18.14377
[51,]	16.95530	16.89268	16.96249	16.73968	17.19426	17.26276	17.51236
[52,]	16.61759	16.54241	16.81227	16.64805	17.03055	17.07464	17.10007
[53,]	16.42042	16.64274	16.86975	17.04261	16.70980	17.38905	17.17296
[54,]	16.67493	17.08767	16.76539	16.83087	17.04800	17.33469	17.41008
[55,]	16.66575	17.02838	16.56953	16.72096	16.78842	17.35782	17.04693
[56,]	16.95586	17.14469	17.03427	16.87975	16.51031	16.93796	17.34443
[57,]	16.64358	16.47934	16.64613	16.84907	16.94122	17.12200	17.42366
[58,]	16.41776	16.83159	16.80743	17.11130	17.14167	16.96434	16.98418
[59,]	16.75399	16.58340	16.47801	16.79078	16.72500	17.00827	17.36569
[60,]	16.33119	16.78036	16.54329	16.90104	16.99806	17.31182	17.03884
[61,]	16.68809	16.73333	16.83493	16.77820	16.67412	16.92281	17.52046
[62,]	16.32436	16.69582	16.45528	17.07564	16.80986	17.38214	17.13138
[63,]	16.82340	16.56721	16.93037	17.17733	16.92747	16.77622	17.31862
[64,]	16.56026	16.69674	16.80356	16.62846	17.15094	16.97955	16.96661
[65,]	16.56500	16.77759	16.74841	16.91139	17.17850	17.27621	17.28501
[66,]	16.35624	16.62391	16.60917	16.99876	16.68829	16.93388	16.89730
[67,]	16.86596	16.62297	16.84879	17.14337	16.95854	16.95836	17.22205
[68,]	16.17558	17.04465	16.83711	16.94213	16.80481	17.16537	17.37016
[69,]	16.58388	16.51190	16.41800	17.00190	17.43139	17.23211	17.52996
[70,]	16.78115	16.65265	17.46724	16.75947	16.83742	16.87664	17.19452
[71,]	16.58065	16.49918	16.38266	16.86838	17.11736	17.41803	17.05809
[72,]	16.36251	16.75161	16.76720	16.73859	16.67112	17.17959	17.20792
[73,]	16.34201	16.65259	16.83277	16.58838	16.67147	16.71597	17.38733
[74,]	16.56135	16.31776	16.49950	16.74098	17.08860	16.87980	17.26702
[75,]	16.46652	16.76184	16.85600	16.82253	17.12986	17.02882	17.39014
[76,]	16.52707	17.01833	16.75993	16.77475	16.77601	17.13195	17.38891
[77,]	16.82523	16.70486	16.92345	16.67955	17.19047	17.21116	17.35858
[78,]	16.51308	16.33064	16.49206	17.06133	17.17027	17.11320	17.16148
[79,]	16.34236	16.97463	17.00880	16.63051	16.98817	17.61176	17.15732
[80,]	16.62846	16.80115	16.61882	16.63861	17.45786	17.12304	17.38127
	[,15]	[,16]	[,17]	[,18]	[,19]	[,20]	[,21]
[1,]	18.20646	18.13294	18.62800	18.18054	18.64998	18.83873	18.63089
[2,]	17.90484	18.30409	18.69688	18.35309	18.60245	18.96069	19.11889

[3,] 17.93954 18.33958 18.23305 18.77900 19.21866 18.37344 19.02267
 [4,] 18.20608 18.58654 18.24729 18.33911 18.95590 19.15097 19.09491
 [5,] 17.74588 18.22315 18.24171 18.54946 18.85306 18.50167 18.80680
 [6,] 18.49513 18.03641 18.40448 18.31583 18.50501 18.95346 18.97185
 [7,] 18.23869 18.27632 18.51861 18.06885 18.21056 18.84095 18.94938
 [8,] 18.09739 18.43140 18.32103 18.71784 18.88667 19.05190 19.26620
 [9,] 18.19787 18.36258 18.83124 18.41776 18.72882 18.82945 18.83506
 [10,] 18.07369 18.25642 18.69099 18.53488 18.48269 18.49539 19.02619
 [11,] 18.10206 18.00555 18.29537 18.55405 18.79810 18.85152 19.10512
 [12,] 18.27125 18.30886 18.33681 18.49044 18.51224 18.83349 18.72125
 [13,] 18.05322 18.31137 18.47921 18.45503 18.54328 18.77539 19.03367
 [14,] 18.21029 18.66853 18.06769 18.28774 18.51074 18.86216 19.01203
 [15,] 18.27780 17.99602 18.34650 18.71998 18.60351 19.01797 19.19101
 [16,] 18.29395 18.02704 18.45455 18.43315 18.76243 18.79511 18.96548
 [17,] 18.15847 18.51067 18.51411 18.30724 18.80373 18.89281 18.94800
 [18,] 18.01325 18.34463 18.70469 18.38314 18.86794 18.64289 18.80667
 [19,] 18.16558 18.15827 18.23913 18.44377 18.57729 18.81017 18.97469
 [20,] 18.12890 18.34122 18.41134 18.78335 18.94816 18.99492 18.79734
 [21,] 18.16224 18.23680 18.48432 18.63962 18.73877 18.95340 19.02896
 [22,] 18.35527 18.09835 18.73180 18.65011 18.59721 18.67808 18.83773
 [23,] 18.36041 18.01846 18.57183 18.74727 18.67269 19.21169 19.03419
 [24,] 18.29004 17.90268 18.51354 18.69542 18.74712 19.04839 19.20430
 [25,] 18.16612 17.88639 18.21780 18.51520 18.62238 18.94517 19.27352
 [26,] 18.35578 18.36804 18.11583 18.52534 18.45945 18.63027 19.23611
 [27,] 18.00121 18.45066 18.55683 18.31157 18.90110 18.55071 18.95383
 [28,] 18.20541 18.10571 18.07446 18.39704 18.63096 18.34723 19.13532
 [29,] 18.41810 18.02396 18.22748 18.61068 18.87697 19.01490 18.93415
 [30,] 17.98527 18.17157 18.21975 18.38010 18.73591 18.90738 19.09821
 [31,] 18.15466 18.26936 18.34426 18.45122 18.74879 18.85575 18.87921
 [32,] 18.31629 18.19698 18.38480 18.47551 18.38035 18.72810 19.09545
 [33,] 18.11160 18.49904 18.25899 18.57698 18.72132 18.89613 19.02769
 [34,] 18.04507 18.07243 18.35386 18.54295 18.78151 18.75091 18.83876
 [35,] 18.36036 18.49717 18.37705 18.43225 18.72037 18.51548 18.96351
 [36,] 18.32115 18.57901 18.77855 18.76078 18.79583 19.12095 19.14017
 [37,] 17.92143 18.59127 18.46572 18.61481 18.82260 18.76277 18.89224
 [38,] 18.04414 18.15435 18.60597 18.85862 18.72075 19.16837 19.32138
 [39,] 18.27973 18.18306 18.50891 18.69780 18.81829 18.96879 19.21904
 [40,] 18.18882 18.46755 18.00906 18.69092 19.18029 18.90033 18.99740
 [41,] 18.12439 18.30193 18.50925 18.37899 18.82897 18.54237 18.86119
 [42,] 18.21386 18.54070 18.39750 18.50666 18.64454 18.79222 19.28649
 [43,] 17.84572 18.14399 18.28572 18.42026 18.60898 18.46347 19.10975
 [44,] 18.45238 17.95924 18.27036 18.42014 18.77155 18.78633 19.18258
 [45,] 18.04196 18.63978 18.38713 18.85302 19.03231 18.72391 18.98263
 [46,] 17.70515 18.18938 18.40433 18.98530 18.68621 18.89623 18.98590
 [47,] 17.90335 18.73980 18.19617 18.99565 18.86518 18.77167 18.89281
 [48,] 18.61359 18.30943 18.24187 18.60479 18.95372 19.23885 19.14907
 [49,] 18.21949 18.11146 18.44240 18.44611 19.05751 18.64579 19.00023
 [50,] 18.49658 18.19978 18.64751 18.34804 18.52912 18.53643 18.45057
 [51,] 16.94858 17.83190 17.48924 17.69680 17.61827 17.77555 18.64401
 [52,] 17.43907 16.92105 17.67761 17.64597 17.58517 17.77055 17.97014
 [53,] 16.96451 17.29014 17.46518 17.63921 17.98598 18.03469 18.02411
 [54,] 17.34576 17.50644 17.74925 17.81855 17.88254 17.82430 18.16206
 [55,] 17.31051 17.67666 17.29119 17.83901 17.69261 17.96928 17.64529
 [56,] 17.33638 17.12916 17.49859 17.53689 17.71900 17.79391 18.15585


```

[57,] 17.22085 17.35699 17.32456 17.50316 17.39390 17.89196 18.39967
[58,] 17.02157 17.84904 17.74472 17.67480 17.46985 18.15430 18.06692
[59,] 17.10653 17.55887 17.73730 17.26314 17.81525 18.09626 17.98616
[60,] 17.29059 17.45238 17.71028 17.74317 17.94589 17.82886 17.83551
[61,] 17.75010 17.36083 17.49446 18.00114 17.50899 18.01499 17.96915
[62,] 17.66286 17.53715 17.51052 17.89996 17.71252 17.94040 18.13767
[63,] 17.56511 16.78475 17.63428 17.58358 17.82931 18.10733 17.89009
[64,] 17.17315 17.49107 17.51993 17.65867 17.33790 18.28280 18.08198
[65,] 17.51700 17.57209 17.45555 17.60583 18.04029 18.02720 17.91788
[66,] 17.56215 17.42467 17.78734 17.20976 17.64414 18.13627 17.94638
[67,] 17.32313 17.33298 17.55111 17.51142 17.85728 18.00448 17.97965
[68,] 17.33630 17.63470 17.51608 17.29333 17.62180 17.74703 18.53779
[69,] 17.55522 17.88892 17.68549 17.81761 17.54922 17.65032 18.18780
[70,] 17.27240 17.40728 17.37196 17.72733 17.98161 17.66627 18.01940
[71,] 17.51739 17.31853 17.42671 17.23256 17.61802 17.75926 18.03326
[72,] 17.60651 17.37234 17.65371 17.82761 17.42817 17.79455 17.98281
[73,] 17.39658 17.51164 17.46104 17.49267 17.69640 17.75575 17.74596
[74,] 16.95818 17.58399 17.60953 17.46675 17.47199 18.32744 18.02201
[75,] 17.23008 17.28418 17.94136 17.57026 17.61758 18.04759 18.07796
[76,] 17.18753 17.40416 17.21424 17.75199 17.73267 17.91675 17.59217
[77,] 17.33654 17.50627 17.51180 17.44885 17.82322 18.04010 18.23415
[78,] 17.33623 17.44473 17.60311 17.98616 17.61377 17.75421 18.17772
[79,] 17.39357 17.67384 17.52311 17.00698 18.12858 18.43742 18.31258
[80,] 17.40548 17.62495 17.55422 17.52761 17.59191 17.80935 17.48302
***** End Print (trajectories) *****

```

For Cochín, the result is correct. For Saint-Anne, print will display too much information. So we need a second method.

show is the default method used to show an object when its name is write in the console. We thus define it by taking into account the size of the object: if there are too many trajectories, show post only part of them.

```

setMethod("show", "Trajectories", function(object) {
  cat("*** Class Trajectories, method Show *** \n")
  cat("* Times ="); print(object@times)
  nrowShow <- min(10, nrow(object@traj))
  ncolShow <- min(10, ncol(object@traj))
  cat("* Traj (limited to a matrix 10x10) = \n")
  print(formatC(object@traj[1:nrowShow,1:ncolShow]),quote=FALSE)
  cat("***** End Show (trajectories) ***** \n")
})

```

trajStAnne

```

*** Class Trajectories, method Show ***
* Times = [1] 1 2 3 4 5 6 7 8 9 10 12 14 16 18 20 22 24 26 28 30 32
* Traj (limited to a matrix 10x10) =
  [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
[1,] 15.94 16.37 15.97 16.58 16.92 16.49 17.07 17.48 17.5 17.38
[2,] 15.85 16.07 15.91 16.72 16.61 17.03 17.5 17.22 17.4 17.48
[3,] 15.68 16.24 16.17 16.53 16.45 16.73 17.12 17.05 17.64 17.23
[4,] 16.18 15.83 16.7 16.33 16.87 16.94 17.11 16.99 16.99 17.35
[5,] 16 16.38 16.2 16.73 16.45 16.94 16.97 16.99 17.41 17.47
[6,] 15.9 16.38 16.09 16.15 16.59 16.71 17.16 17.13 17.4 17.67
[7,] 15.9 16.35 16.46 16.62 16.8 16.69 16.96 16.9 17.3 17.05
[8,] 15.74 16.32 16.62 16.24 16.75 16.89 16.68 17.02 17.36 17.19

```

```

[9,] 16.03 16.13 16.15 16.32 16.39 17.18 16.8 17.31 17.11 17.5
[10,] 15.72 15.82 16.45 16.65 16.22 16.63 16.83 17.01 17.38 17.26
***** End Show (trajectories) *****

```

A small problem must still be regulated. We saw in section 4.6 page 17 that `new` should be usable without any argument. However, it is no longer true:

```
new("Trajectories")
```

```

*** Class Trajectories, method Show ***
* Times =numeric(0)
* Traj (limited to a matrix 10x10) =

Error in object@traj[1:nrowShow, 1:ncolShow]: subscript out of bounds

```

Taking in account an empty object

Indeed, `new` creates an object, then display it using `show`. In the case of `new` without any argument, the empty object is send to `show`. However, `show` as we conceived it cannot treat the empty object.

More generally, all our methods must take into account the fact that they may have to deal with the empty object:

```

setMethod("show","Trajectories", function(object){
  cat("*** Class Trajectories, method Show *** \n")
  cat("* Times = "); print (object@times)

  nrowShow <- min(10,nrow(object@traj))
  ncolShow <- min(10,ncol(object@traj))
  cat("* Traj (limited to a matrix 10x10) = \n")

  if(length(object@traj) != 0){
    print(formatC(object@traj[1:nrowShow,1:ncolShow]),quote=FALSE)
  } else {}
  cat("***** End Show (trajectories) ***** \n")
}
)

new("Trajectories")

```

```

*** Class Trajectories, method Show ***
* Times = numeric(0)
* Traj (limited to a matrix 10x10) =
***** End Show (trajectories) *****

```