Second hight-tune coding sprint – 3-4 may, 2018, Paris

Thursday morning :
→ Brief introduction on the project status (Fleur Couvreux)
→ Available LES and standardization
(Fleur Couvreux & Marie-Pierre Lefebvre, 20min)
→ Available shared scripts : Description with illustrations (Frédéric Hourdin, 30 min)
→ News from Exeter (Daniel Williamson & Vicroria Volodina, 1hour ?)
→ Discussion on the work content and organization for the coming days

Thursday afternoon :
Work on the codes

Friday morining :
Debriefing (1hour max) + work

Friday afternoon :
Discussions (2 hours max) + work :
→ On the evolution of the shared htune scripts
→ On the next priorities in terms of parameterizations
→ On the coordination/collaboration with Exeter
→ From 1D to 3D

Step 1a : generating setup files

param2R.sh : define list of parameters and there range
create the R script ModelParam.R

Usage : ./param2R.sh LHCSIZE NLHC PARAM_FILE

Le nombre total de simulations sera NLHC * LHCSIZE

Ex : ./param2R.sh 30 3 LMDZ/param_cld

```
testcubeSCM <-
MakeRankExtensionCubes(n=LHCSIZE,m=NPARA,k=NLHC,w=0.2, FAC_t=0.05,Imax=500)
```

NLHC should be ≥ 3
How to choose the other parameters ?

Exemple of parameter file :

```
thermals_fact_epsilon 5.e-4 4.e-3 2.e-3 log
thermals_betalpha 0.5 1. 0.9 linear
thermals_afact 0.16666667 1 0.66666667 linear
thermals_ed_dz 0.0 0.2 0.07 linear
cld_lc_lsc 1e-4 1e-3 5e-4 linear
z0min 1e-4 1e-2 log
```

Color chart :
Model dependant
Question
To be done

Step 1b :Generating the Latin Hyper Cube sampling (htune_convertDesign.R)

creates design for the emulator using ModelParam.R
outputs : Par1D_Wave1.asc containing the parameter values
for SCM simulations
Wave1.RData containing normalized parameter values for
the SCM
Automatically run by param2R.sh after version 9

TBD : create a function that could be called from anywhere to
transformed normalized to non normalized

```
"t_IDs" "thermals_fact_epsilon" "thermals_betalpha" "thermals_afact" "thermals_ed_dz" "cld_lc_lsc" "z0min"
"SCM_1-001" 0.00176890073397987 0.569943311488411 0.846839357463217 0.119480651975609 0.000986134872906841 0.00884693657
"SCM_1-002" 0.00108746764500753 0.560571146170453 0.451489045169828 0.194212773878438 0.000441683268160559 0.00397120190
"SCM_1-003" 0.00155456791454632 0.539753121490746 0.672070160907955 0.0643667436670512 0.0002050994106289 0.001315390401
"SCM_1-004" 0.00127296623189608 0.811376507665652 0.362698160313403 0.0455267948114003 0.000137556258838158 0.00860164685
"SCM_1-005" 0.00282296499278743 0.776537479142038 0.541345494700146 0.0516816835710779 0.000836041868755128 0.00938256900
"SCM_1-006" 0.000703534480680766 0.707575917806632 0.494969216636829 0.125295180737351 0.000526144763897173 0.00981618580
"SCM_1-007" 0.00226220066528887 0.503595818478304 0.345543755397957 0.107448288596546 0.000271852736026049 0.00709725225
"SCM_1-008" 0.000733637018220402 0.520020785710464 0.888391049902339 0.129473727609341 0.000685064382920973 0.00223968070
"SCM_1-009" 0.0033826728034985 0.820045974121119 0.40142291370498 0.175838785067511 0.000410688407374546 0.00918312891663
"SCM_1-010" 0.0023751205892179 0.688135024998337 0.958075632592981 0.14920674375724 0.000315125980998855 0.00604585711137
"SCM_1-011" 0.00186617777261378 0.796525626323031 0.911967496953219 0.000492116739042103 0.000637655973976944 0.00520461
"SCM_1-012" 0.000769240624538988 0.592353973630816 0.830755332676756 0.0346160239803915 0.000351936126137152 0.007652537
"SCM_1-013" 0.0025434905173408 0.642925126082264 0.281404965902739 0.141914477616859 0.000564674943557475 0.00047102856
"SCM_1-014" 0.000607009116910186 0.766023993492126 0.638227289172274 0.171263636065026 0.000102190782425459 0.0049472118
"SCM_1-015" 0.00204437259635504 0.935195826704148 0.442018218255045 0.104677453530021 0.000245472253130283 0.00263371575
"SCM_1-016" 0.000654448162675327 0.661315408931114 0.187248302181684 0.0580605496279895 0.000664314172060695 0.002794382
```

Step 2 : runing SCM (serie_LMDZ.sh)

The only model depedant script
Use : ./serie_LMDZ.sh [cas1] [cas2]
Run a series of the LMDZ SCM reading the parameters from Par1D_Wave1.asc
netcdf ouptut files should be put in
./WAVE1/[cas1] ...
with names SCM_1-101.nc given in Par1D_Wave1.asc
Also put the available LES 1D output files in WAVE1

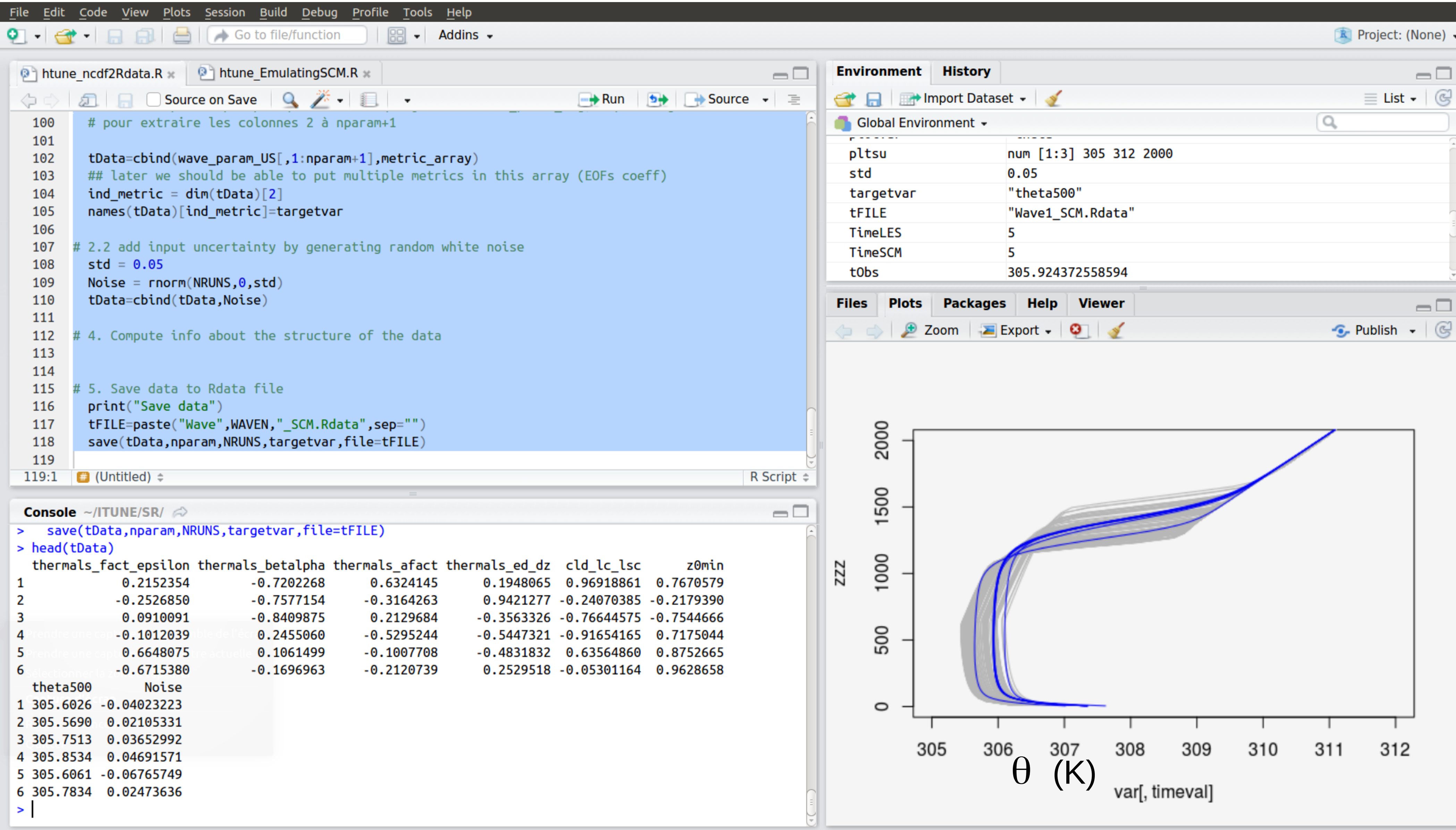
TBD : a version for CNRM
- Treat LES independently depending on the case considered ?
- Normalize the case names ?
- Share part of this script between models as well ?

```
WAVE1/

WAVE1/ayotte24SC:
LES0.nc   SCM_1-001.nc  SCM_1-013.nc  SCM_1-025.nc  SCM_1-037.nc  SCM_1-049.nc  SCM_1-061.nc  SCM_1-073.nc
LES10.nc  SCM_1-002.nc  SCM_1-014.nc  SCM_1-026.nc  SCM_1-038.nc  SCM_1-050.nc  SCM_1-062.nc  SCM_1-074.n
LES11.nc  SCM_1-003.nc  SCM_1-015.nc  SCM_1-027.nc  SCM_1-039.nc  SCM_1-051.nc  SCM_1-063.nc  SCM_1-075.nc
LES1.nc   SCM_1-004.nc  SCM_1-016.nc  SCM_1-028.nc  SCM_1-040.nc  SCM_1-052.nc  SCM_1-064.nc  SCM_1-076.nc
LES2.nc   SCM_1-005.nc  SCM_1-017.nc  SCM_1-029.nc  SCM_1-041.nc  SCM_1-053.nc  SCM_1-065.nc  SCM_1-077.nc
LES3.nc   SCM_1-006.nc  SCM_1-018.nc  SCM_1-030.nc  SCM_1-042.nc  SCM_1-054.nc  SCM_1-066.nc  SCM_1-078.nc
LES4.nc   SCM_1-007.nc  SCM_1-019.nc  SCM_1-031.nc  SCM_1-043.nc  SCM_1-055.nc  SCM_1-067.nc  SCM_1-079.nc
LES5.nc   SCM_1-008.nc  SCM_1-020.nc  SCM_1-032.nc  SCM_1-044.nc  SCM_1-056.nc  SCM_1-068.nc  SCM_1-080.nc
LES6.nc   SCM_1-009.nc  SCM_1-021.nc  SCM_1-033.nc  SCM_1-045.nc  SCM_1-057.nc  SCM_1-069.nc  SCM_1-081.nc
LES7.nc   SCM_1-010.nc  SCM_1-022.nc  SCM_1-034.nc  SCM_1-046.nc  SCM_1-058.nc  SCM_1-070.nc  SCM_1-082.nc
LES8.nc   SCM_1-011.nc  SCM_1-023.nc  SCM_1-035.nc  SCM_1-047.nc  SCM_1-059.nc  SCM_1-071.nc  SCM_1-083.nc
LES9.nc   SCM_1-012.nc  SCM_1-024.nc  SCM_1-036.nc  SCM_1-048.nc  SCM_1-060.nc  SCM_1-072.nc  SCM_1-084.nc

WAVE1/sanduref:
LES0.nc       SCM_1-006.nc  SCM_1-012.nc  SCM_1-018.nc  SCM_1-024.nc  SCM_1-030.nc  SCM_1-036.nc  SCM_1-042
SCM_1-001.nc  SCM_1-007.nc  SCM_1-013.nc  SCM_1-019.nc  SCM_1-025.nc  SCM_1-031.nc  SCM_1-037.nc  SCM_1-0
SCM_1-002.nc  SCM_1-008.nc  SCM_1-014.nc  SCM_1-020.nc  SCM_1-026.nc  SCM_1-032.nc  SCM_1-038.nc  SCM_1-0
SCM_1-003.nc  SCM_1-009.nc  SCM_1-015.nc  SCM_1-021.nc  SCM_1-027.nc  SCM_1-033.nc  SCM_1-039.nc  SCM_1-0
SCM_1-004.nc  SCM_1-010.nc  SCM_1-016.nc  SCM_1-022.nc  SCM_1-028.nc  SCM_1-034.nc  SCM_1-040.nc  SCM_1-0
```

Step 3 : Computing metrics (htune_ncdf2Rdata.R)

i/ Reads results from WAVE1/[cas]/*nc
ii/ Plots profiles for all the simulations with function "tout_tracer" from htune_plot
iii/ computes metrics with function get_metric from htune_metric.R
output files
Wave1_LES.Rdata : metrics computed on LES
Wave1_SCM.Rdata : metrics computed on SCM
Assume that all the files are at an hourly time frequency
Metrics already available :
targetvar="theta500" theta 500hPa
targetvar="zhneb"   average height of cloudiness (int z f dz / int f dz)
TBD : control case_name from outside
TBD : extend the "tout_tracer" to "tout"

Ayotte 24SC


File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
htune_ncdf2Rdata.R
htune_EmulatingSCM.R
Source on Save
Run
Source
# pour extraire les colonnes 2 à nparam+1
tData=cbind(wave_param_US[,1:nparam+1],metric_array)
later we should be able to put multiple metrics in this array (EOFs coeff)
ind_metric = dim(tData)[2]
names(tData)[ind_metric]=targetvar
# 2.2 add input uncertainty by generating random white noise
std = 0.05
Noise = rnorm(NRUNS,0,std)
tData=cbind(tData,Noise)
# 4. Compute info about the structure of the data
# 5. Save data to Rdata file
print("Save data")
tFILE=paste("Wave",WAVEN,"_SCM.Rdata",sep="")
save(tData,nparam,NRUNS,targetvar,file=tFILE)
119:1
(Untitled)
R Script
Console ~/ITUNE/SR/
> save(tData,nparam,NRUNS,targetvar,file=tFILE)
> head(tData)
thermals_fact_epsilon thermals_betalpha thermals_afact thermals_ed_dz cld_lc_lsc z0min
1 0.2152354 -0.7202268 0.6324145 0.1948065 0.96918861 0.7670579
2 -0.2526850 -0.7577154 -0.3164263 0.9421277 -0.24070385 -0.2179390
3 0.0910091 -0.8409875 0.2129684 -0.3563326 -0.76644575 -0.7544666
4 -0.1012039 0.2455060 -0.5295244 -0.5447321 -0.91654165 0.7175044
5 0.6648075 0.1061499 -0.1007708 -0.4831832 0.63564860 0.8752665
6 -0.6715380 -0.1696963 -0.2120739 0.2529518 -0.05301164 0.9628658
theta500 Noise
1 305.6026 -0.04023223
2 305.5690 0.02105331
3 305.7513 0.03652992
4 305.8534 0.04691571
5 305.6061 -0.06765749
6 305.7834 0.02473636
Environment History
Import Dataset
List
Global Environment
pltsu num [1:3] 305 312 2000
std 0.05
targetvar "theta500"
tFILE "Wave1_SCM.Rdata"
TimeLES 5
TimeSCM 5
tObs 305.924372558594
Files Plots Packages Help Viewer
Zoom
Export
Publish
zzz
0
500
1000
1500
2000
305
306
307
308
309
310
311
312
θ (K)
var[, timeval]

Sanduref

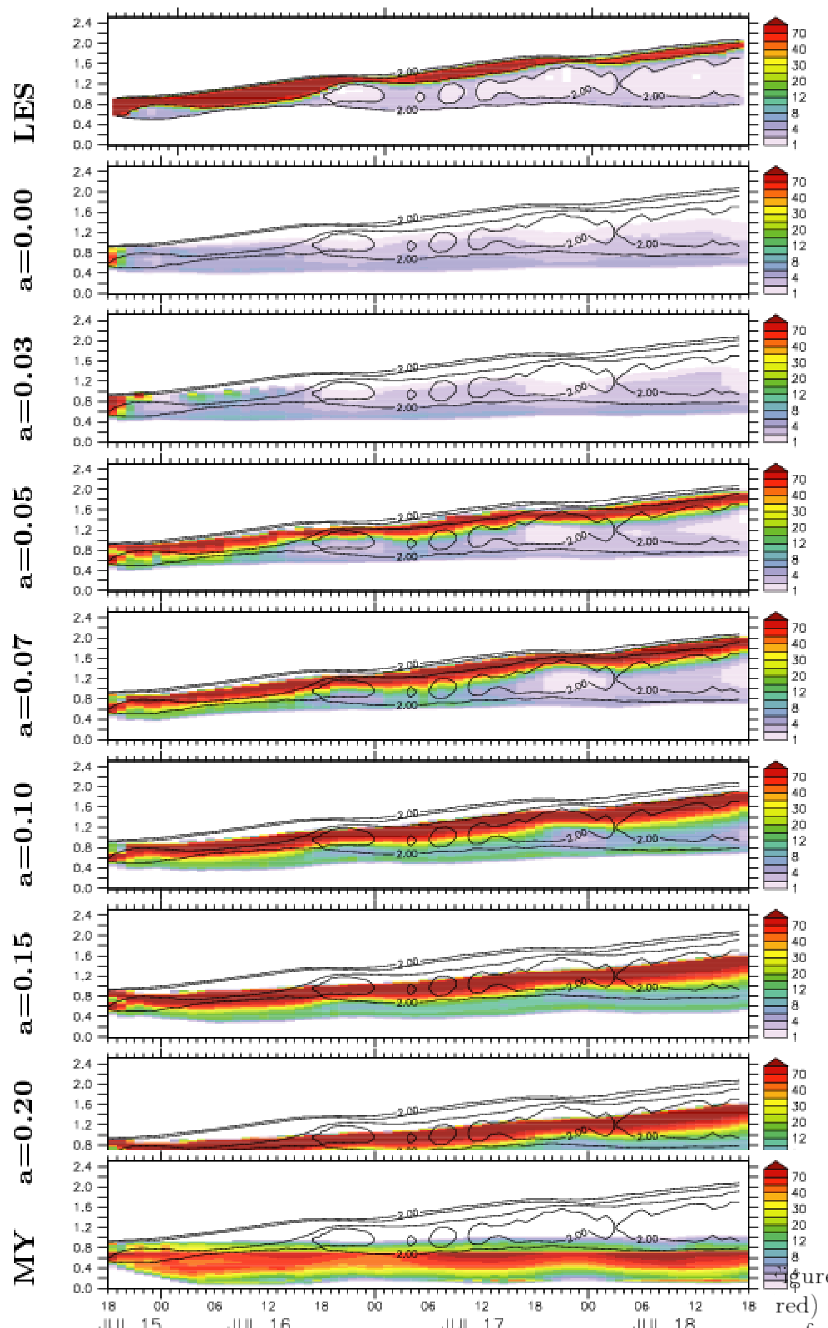
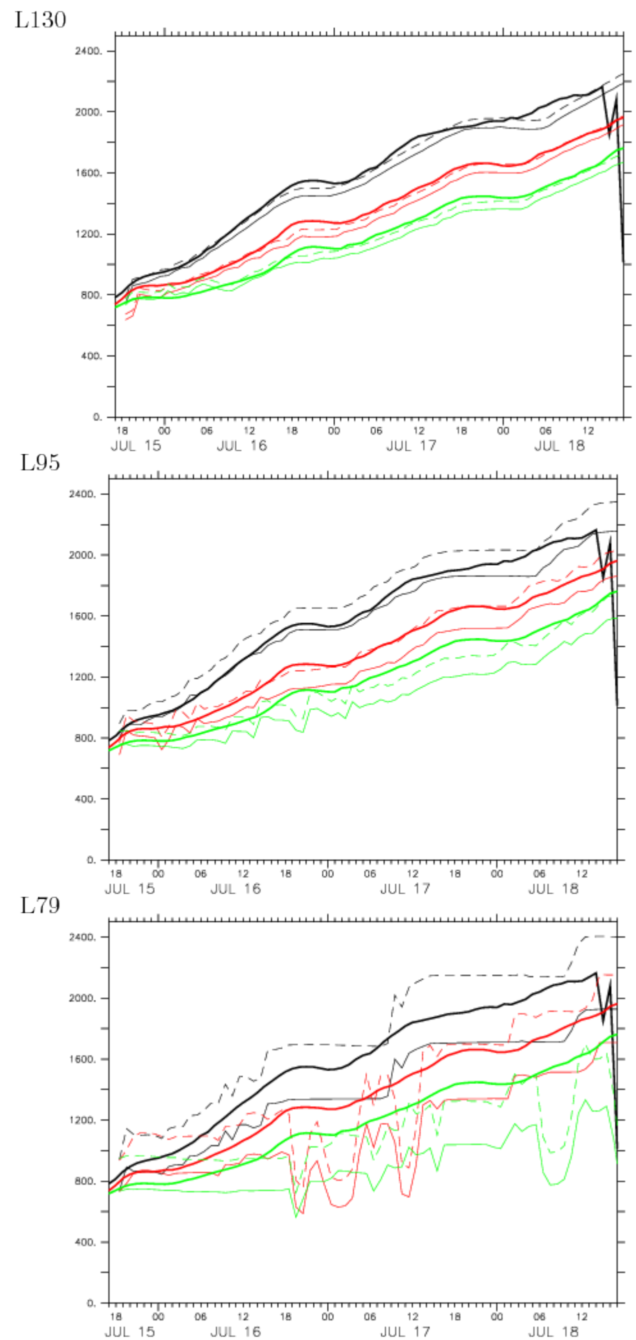


Figure 2: Case sanduref. 130 layers



igure 3: Sensitivity to vertical resolution. Case sandufast (black), sanduref red) and sanduslow (green), a=0.07. Height of maximum nebuloisty computed ıs $\int r^4 z dz / \int r^4 dz$ where r is the nebuosity for the LES (thick curves), SCM thin full lines). The dashed curves is one level above the thin curve for the SCM.

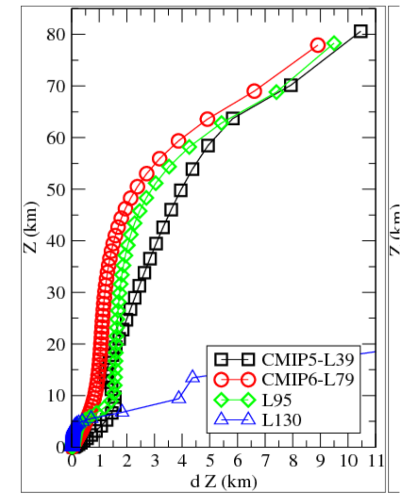
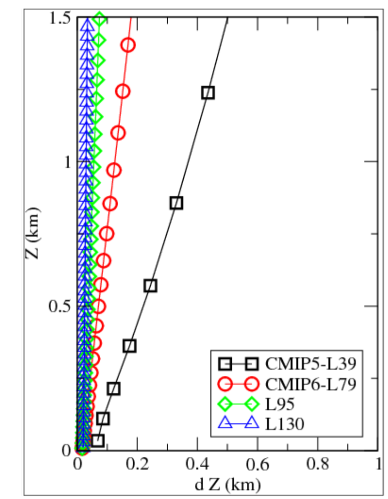


Figure 1: Vertical grids used



l for sensitivity experiments

3

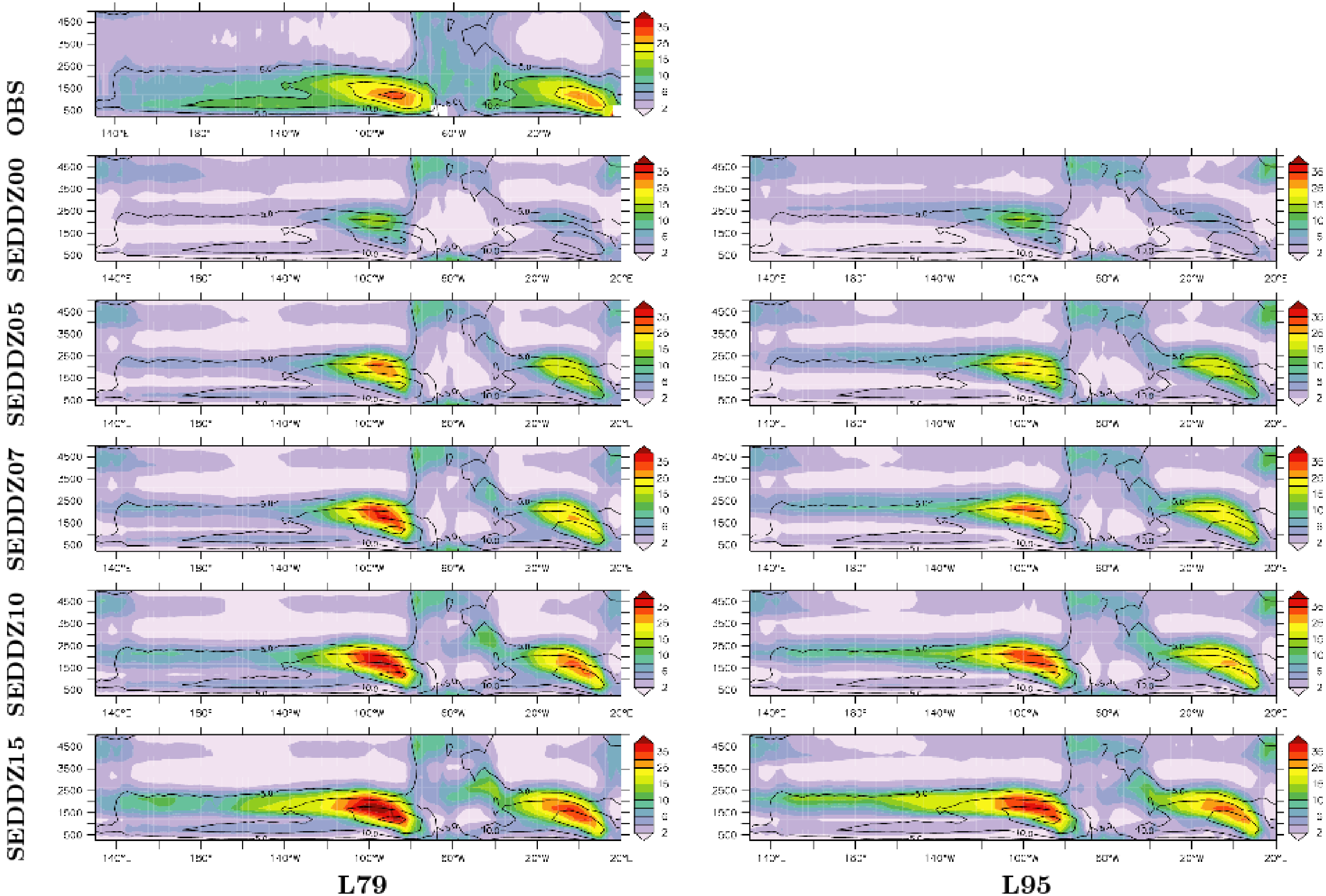
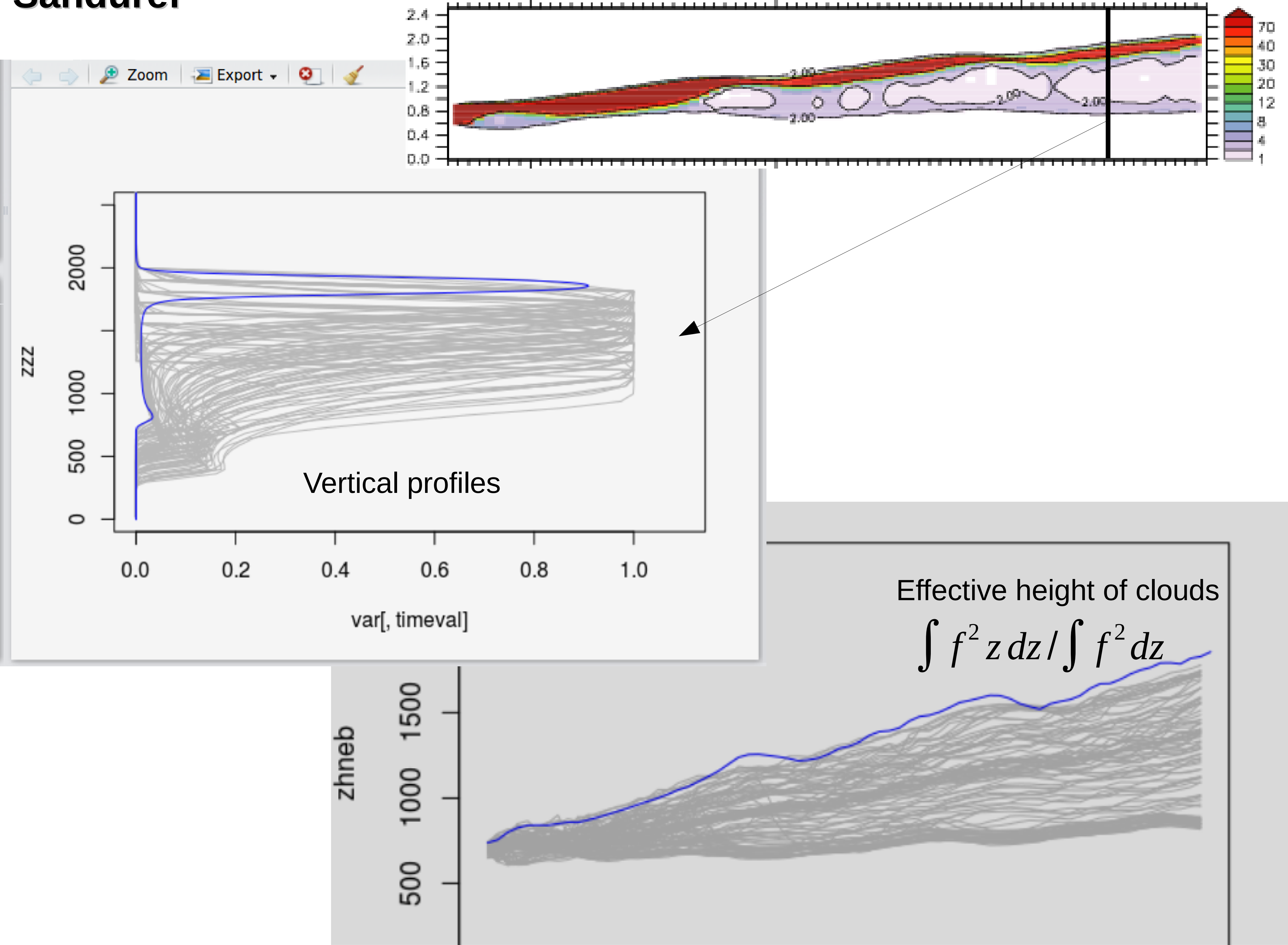


Figure 4: Nebulosity average 25S-15S, Cosp simulator, Annual mean, L79 et L95, 4 years

Sanduref



2.4
2.0
1.6
1.2
0.8
0.4
0.0
2.00
70
40
30
20
12
8
4
1
Zoom
Export
zzz
0
500
1000
2000
0.0
0.2
0.4
0.6
0.8
1.0
var[, timeval]
Vertical profiles
Effective height of clouds
$\int f^2 z\,dz / \int f^2 dz$
zhneb
500
1000
1500

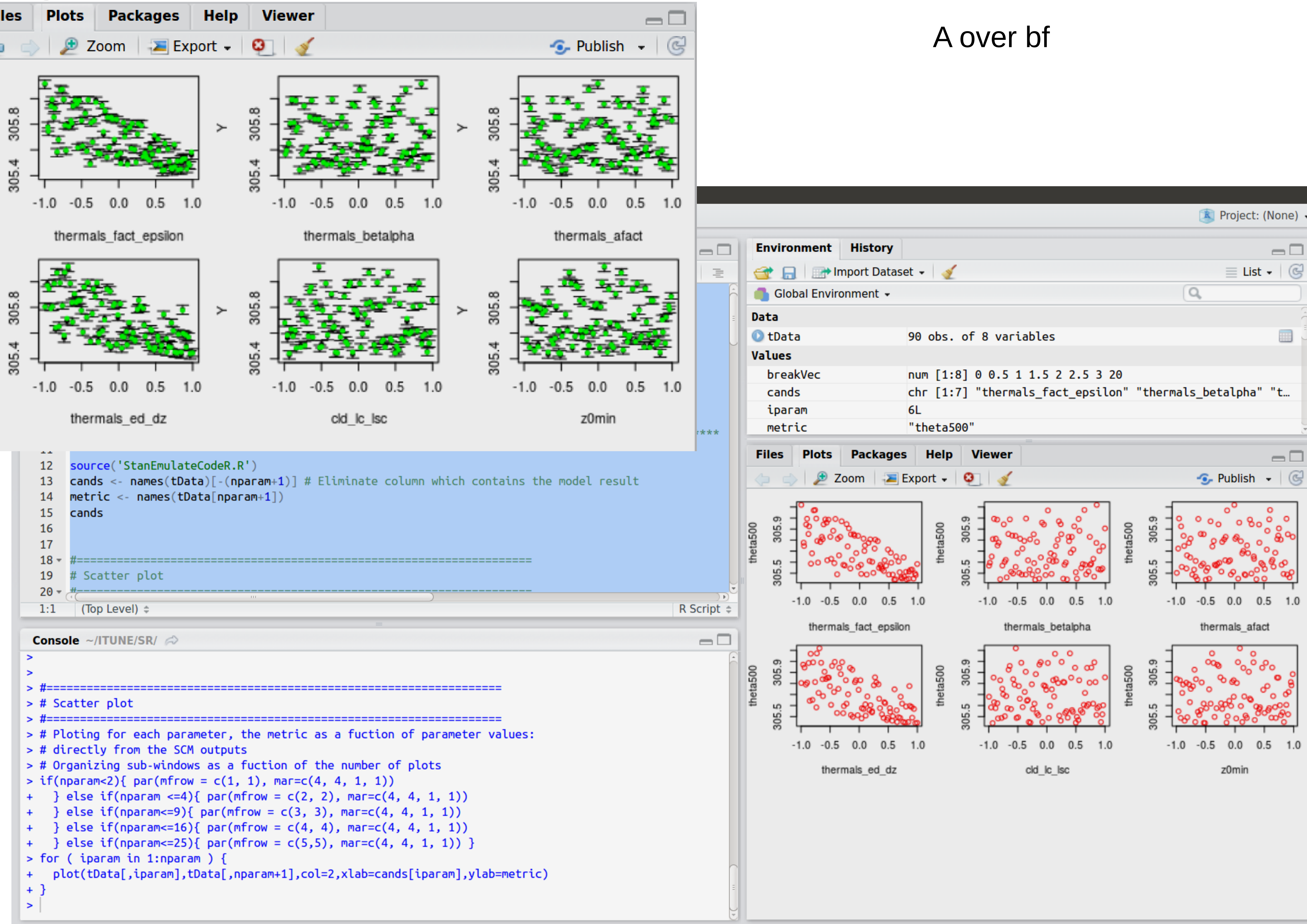
Step 4 : Building and using emulators (htune_EmulatingSCM.R)

Emulator building reading Wave1_LES.Rdata and Wave1_SCM.Rdata

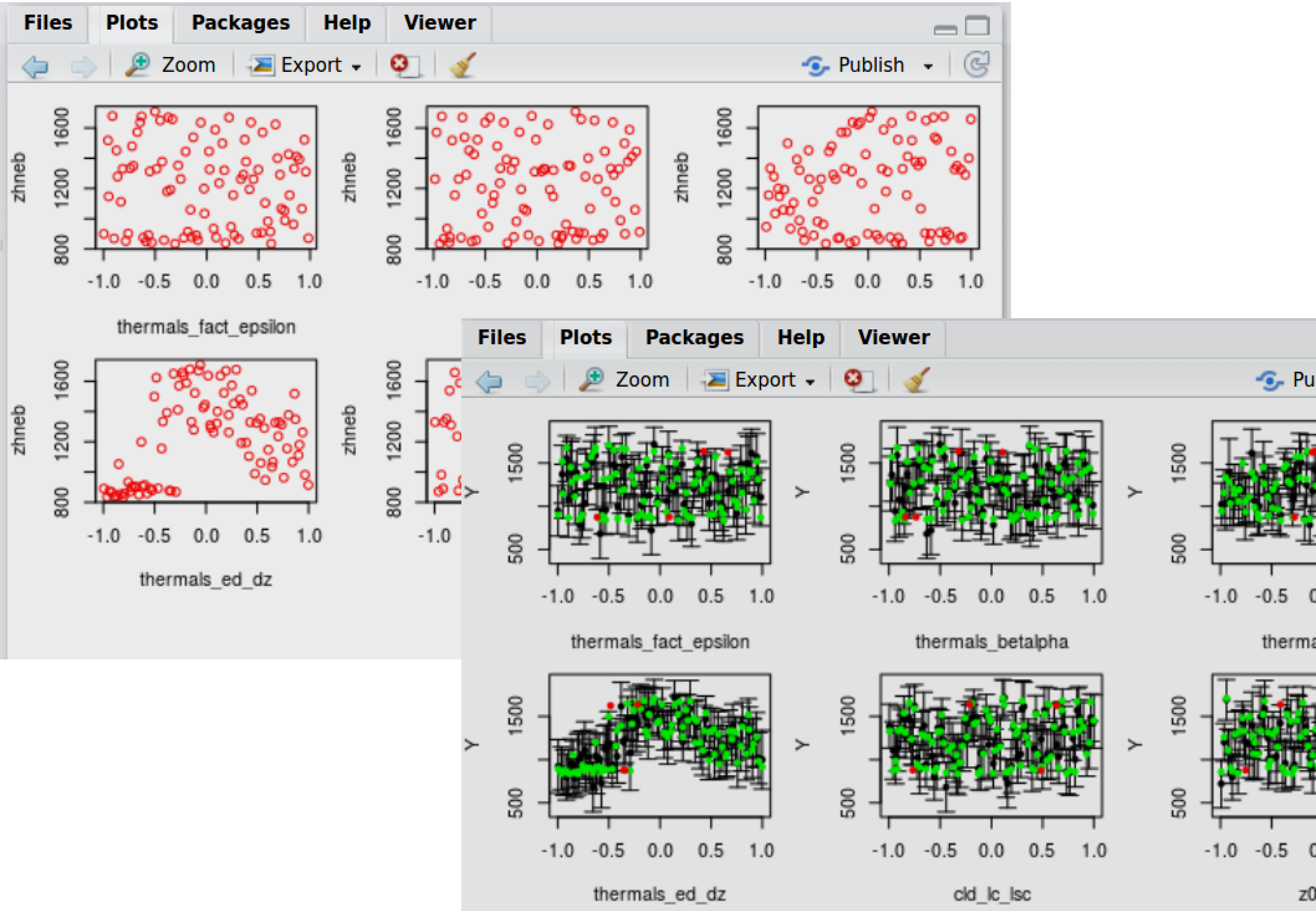
Definition and plots of ROY spaces

Ayotte 24SC

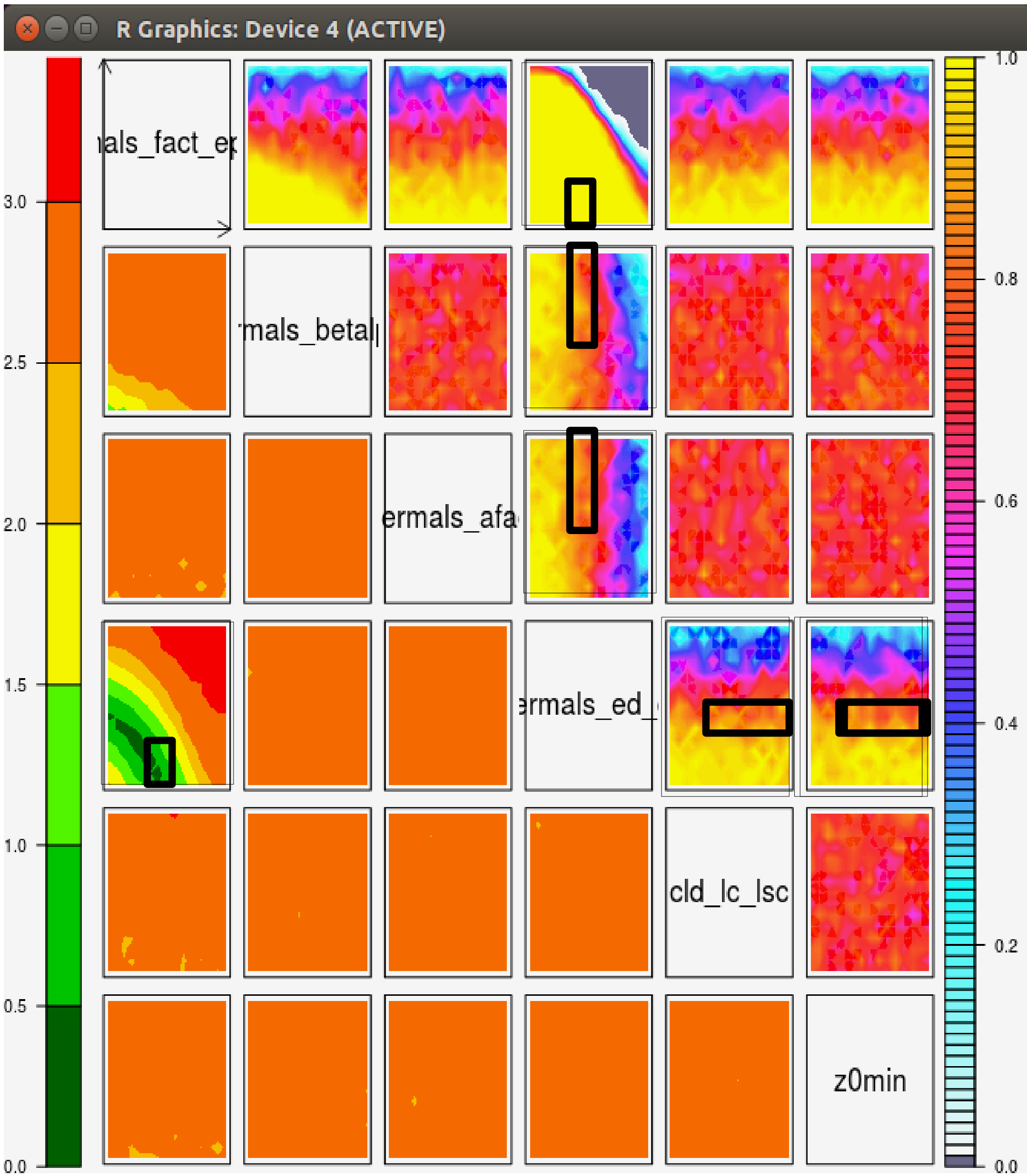
A over bf



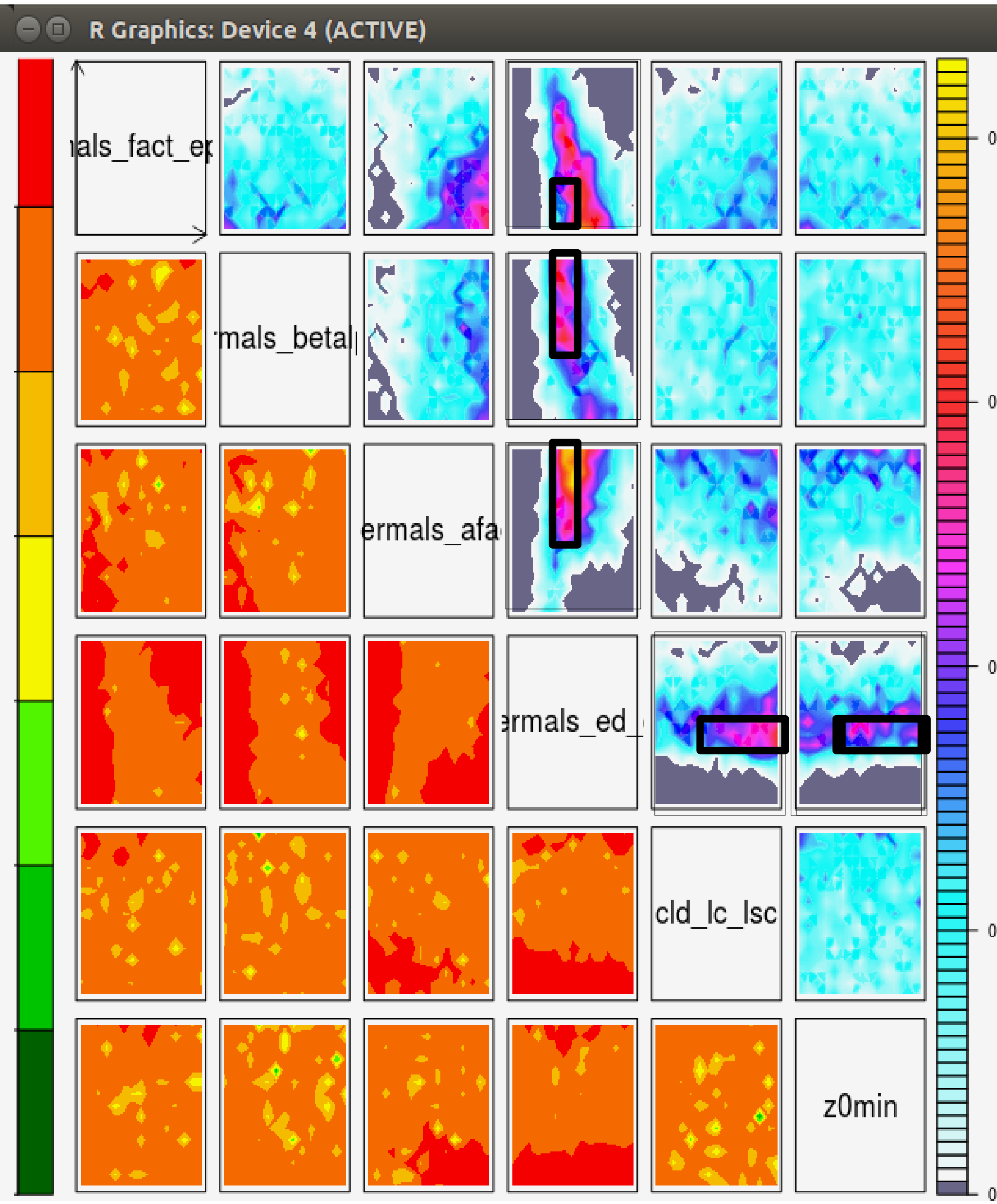
Sanduref


Files
Plots
Packages
Help
Viewer
Zoom
Export
Publish
zhneb
thermals_fact_epsilon
thermals_ed_dz
Y
thermals_betalpha
cld_lc_lsc

Ayotte 24SC, θ500

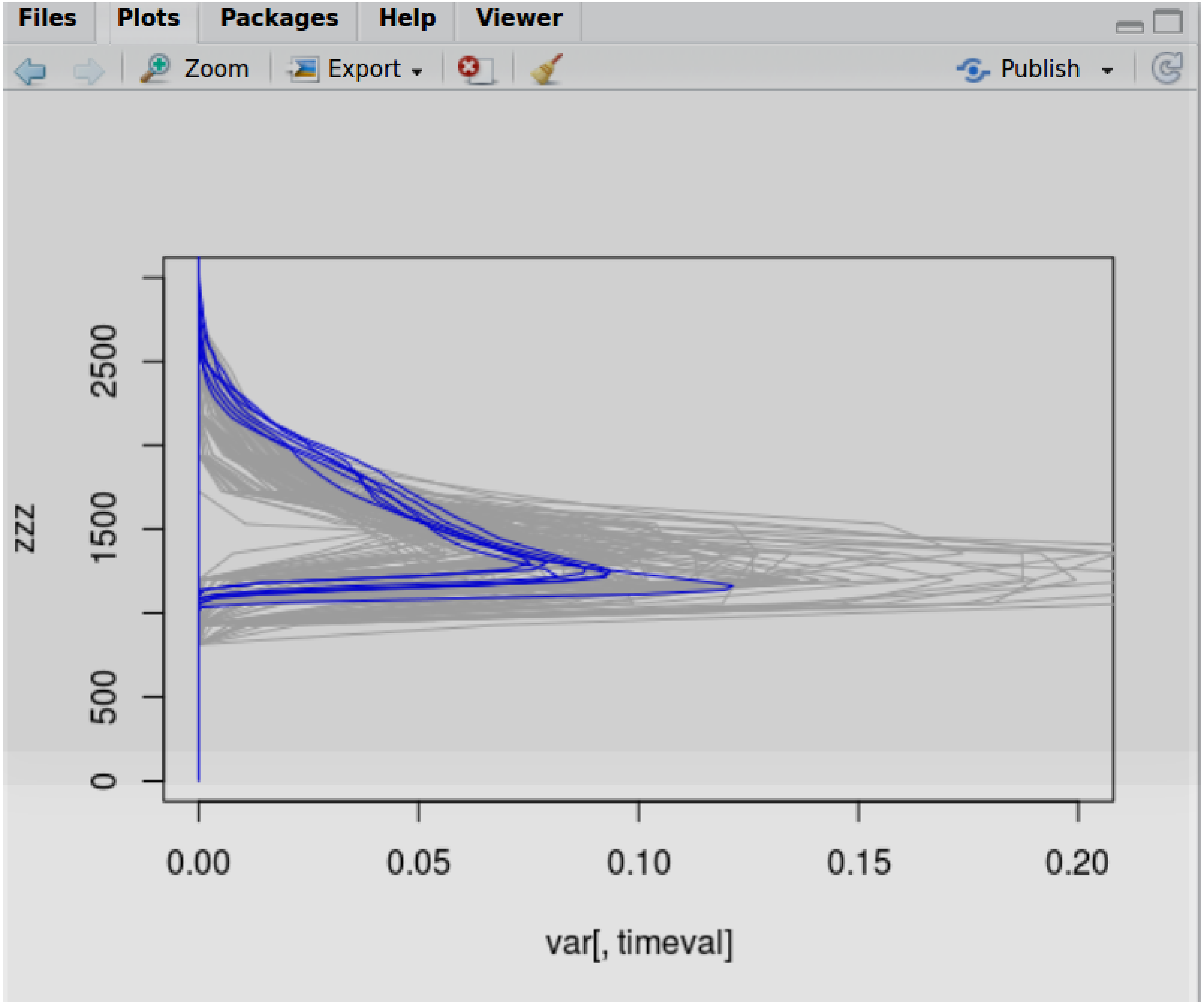


Sanduref, cloud effective height

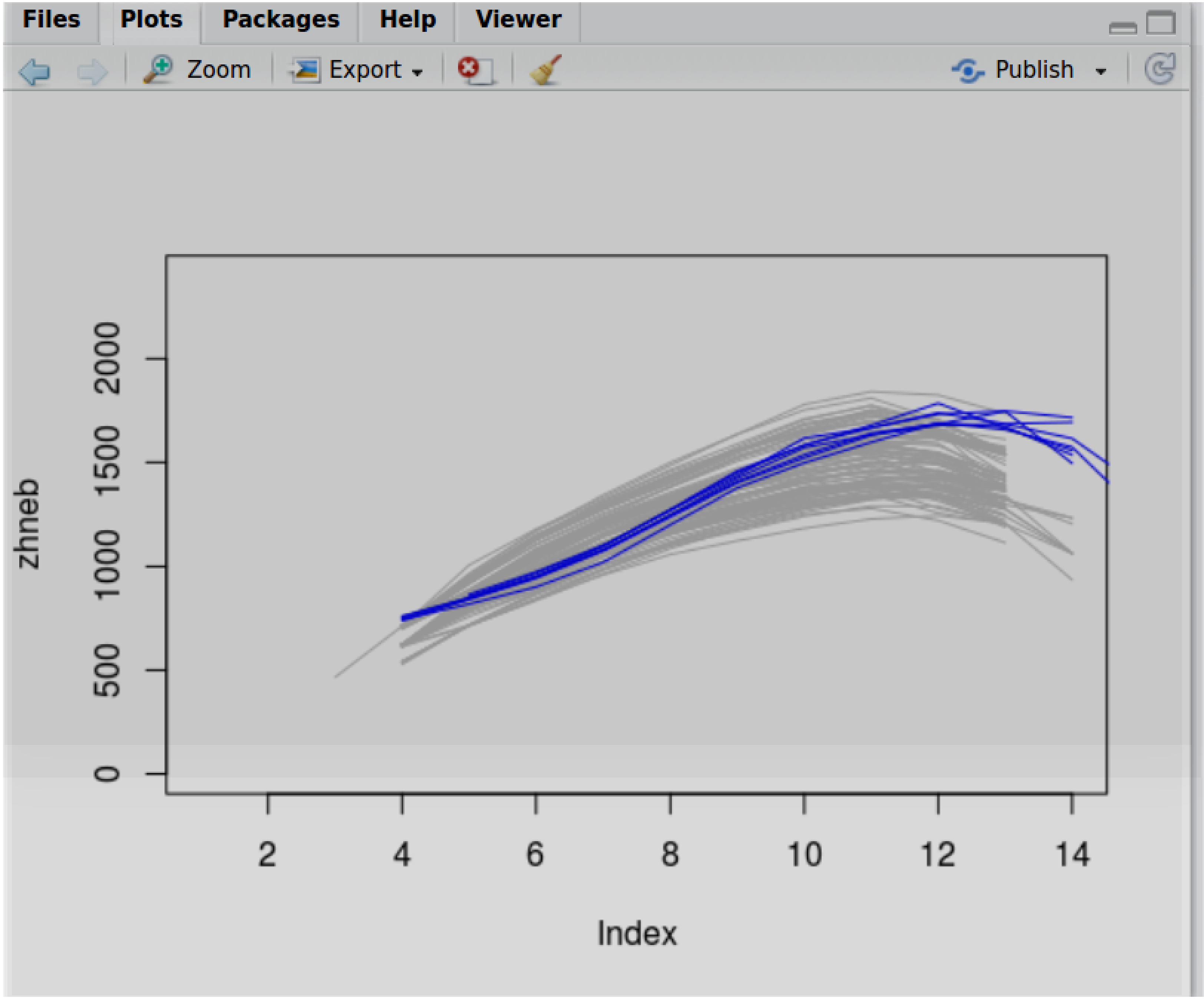


Other cases

arm_cu

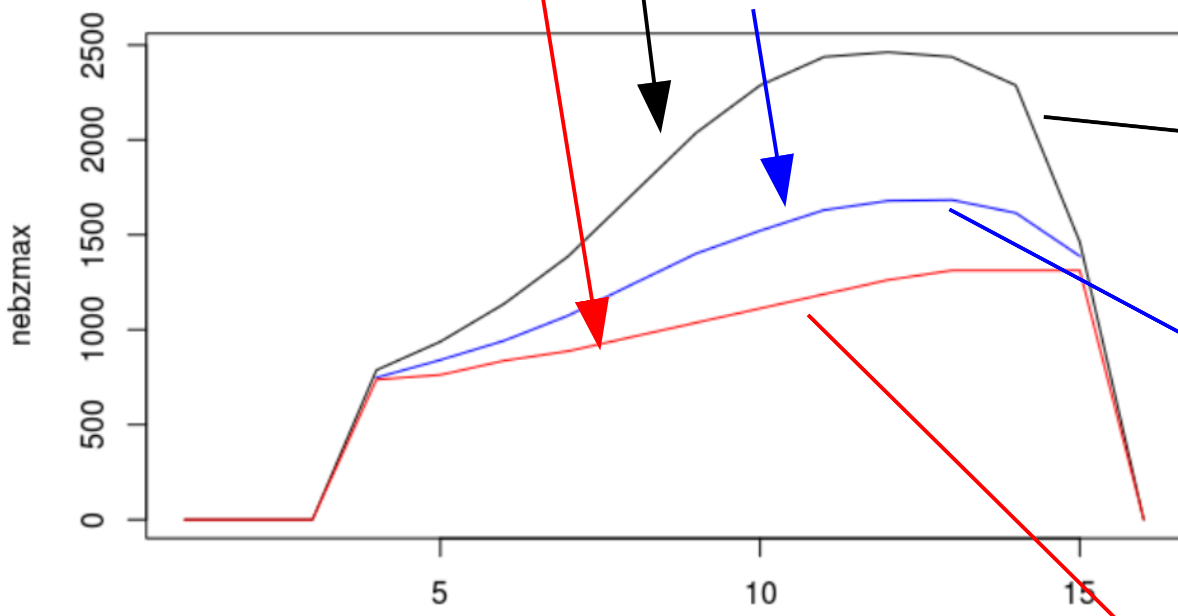

Files
Plots
Packages
Help
Viewer
Zoom
Export
Publish
zzz
0
500
1500
2500
0.00
0.05
0.10
0.15
0.20
var[, timeval]

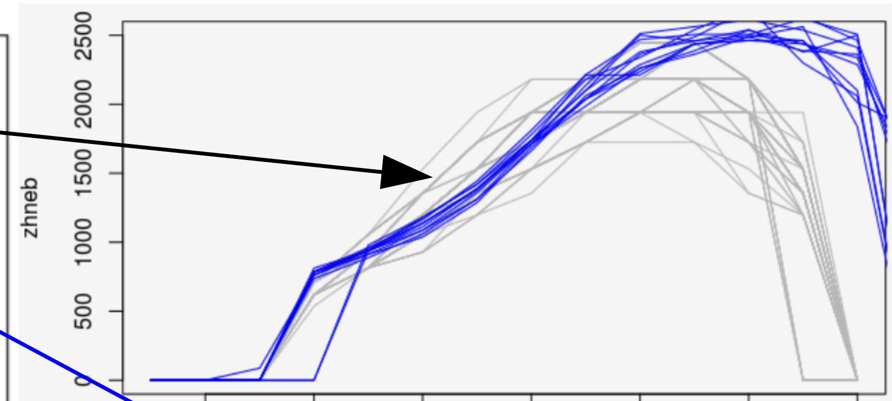
arm_cu

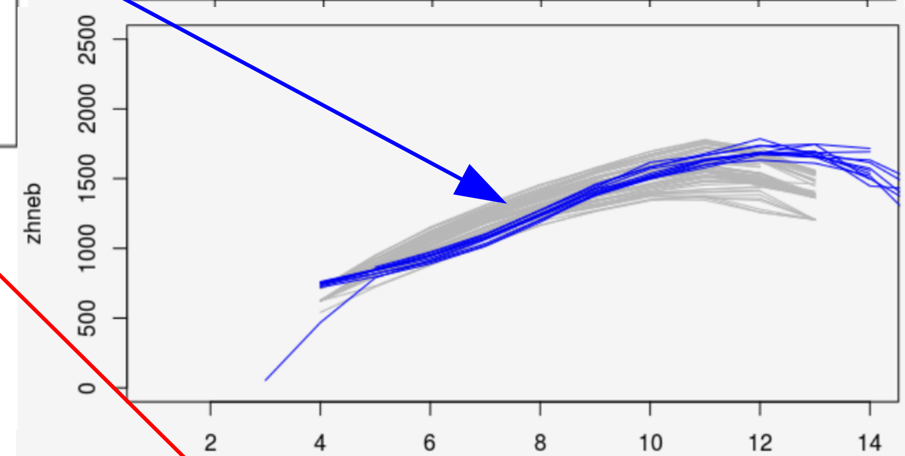

Files
Plots
Packages
Help
Viewer
Zoom
Export
Publish
zhneb
0
500
1000
1500
2000
2
4
6
8
10
12
14
Index

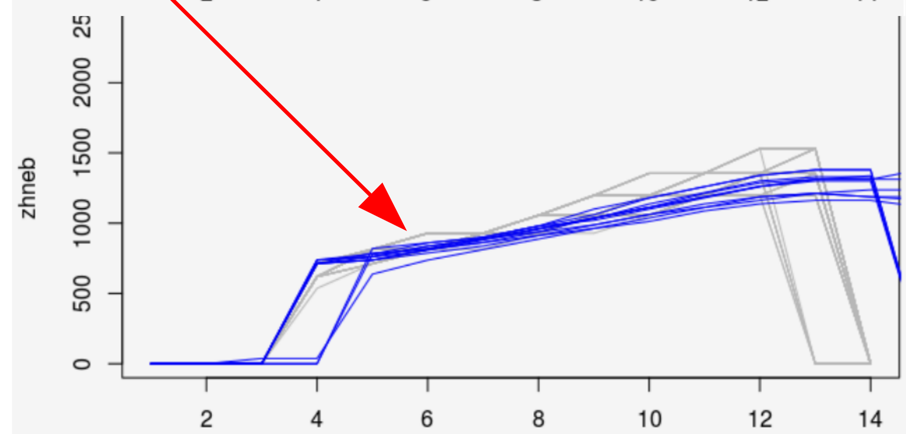
arm_cu

Height min , max (coresponding to a nebulosity > 5 % of max value
Effective height $\int f^2 z\, dz / \int f^2 dz$

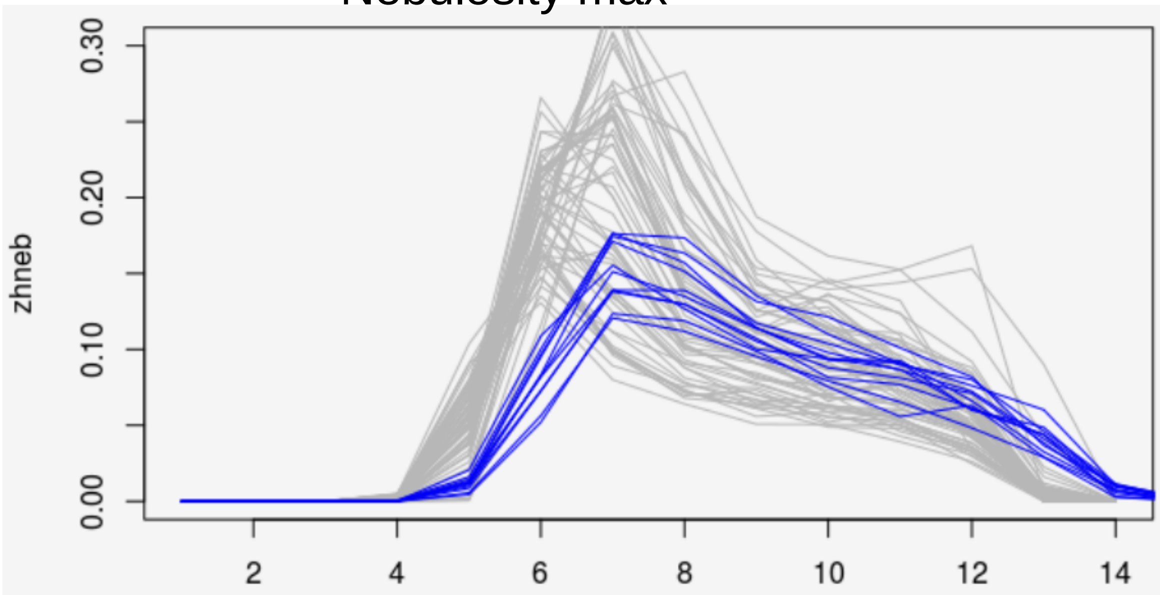






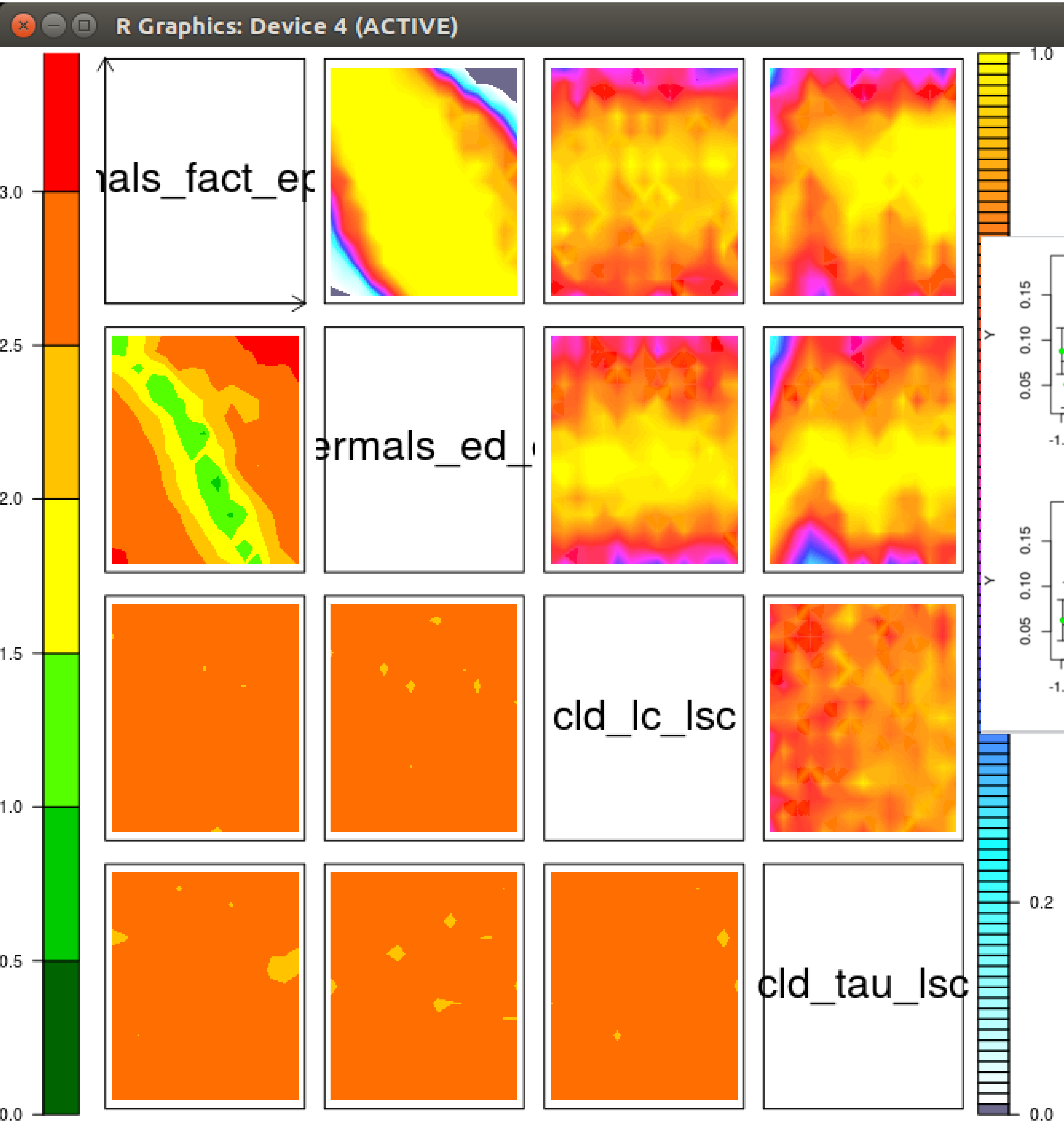


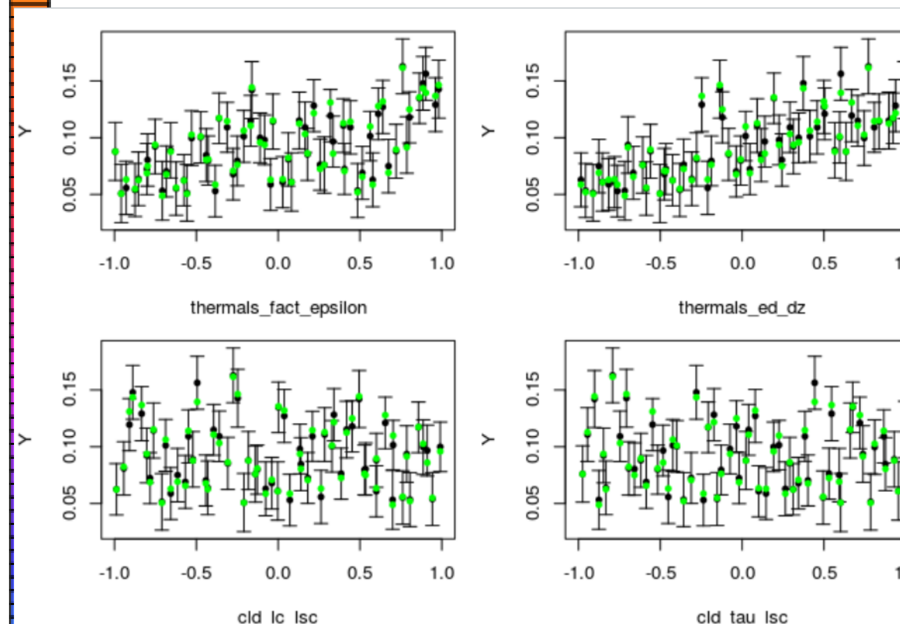
Nebulosity max



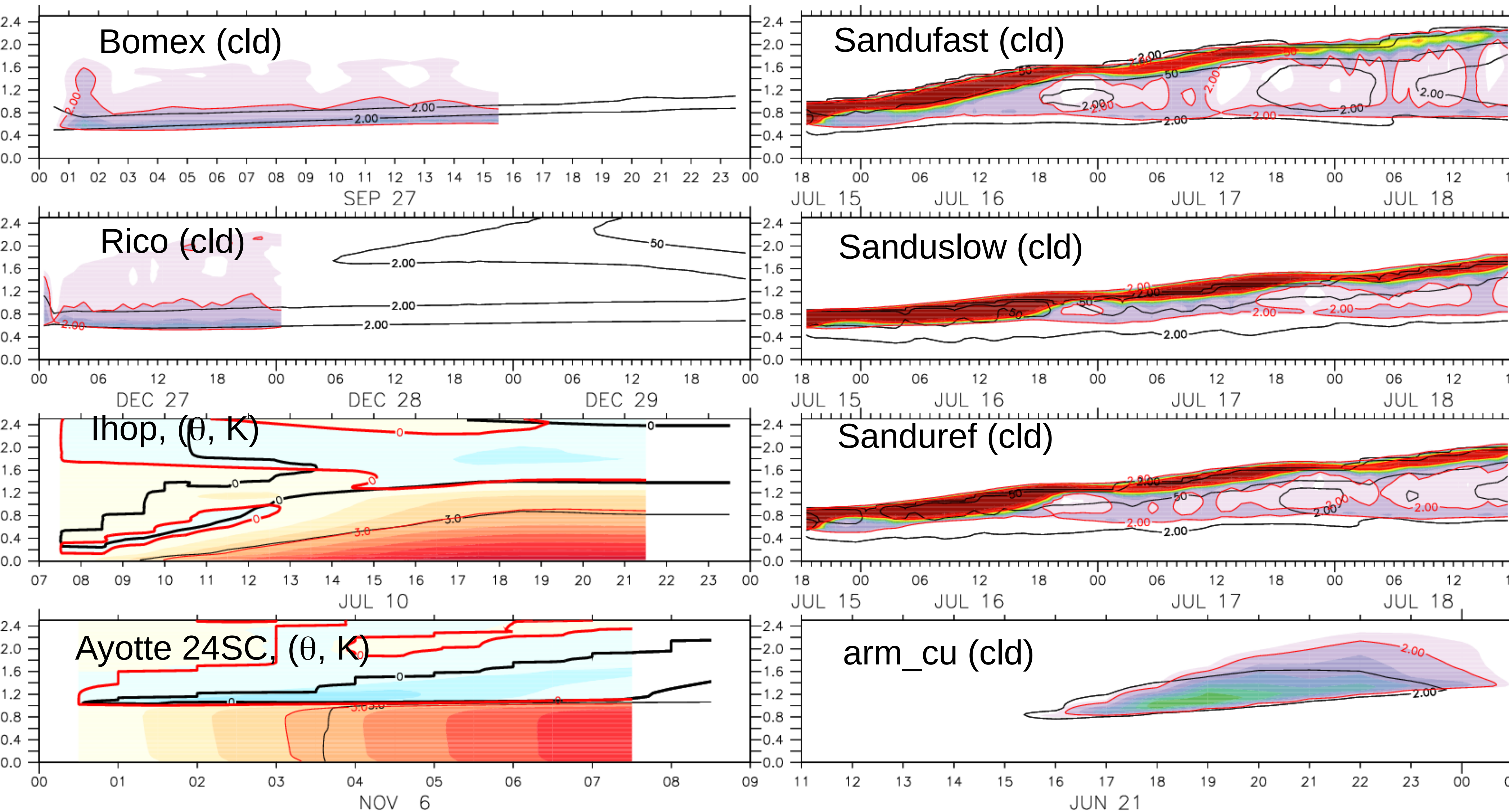
arm_cu

Nebul max arm_cu



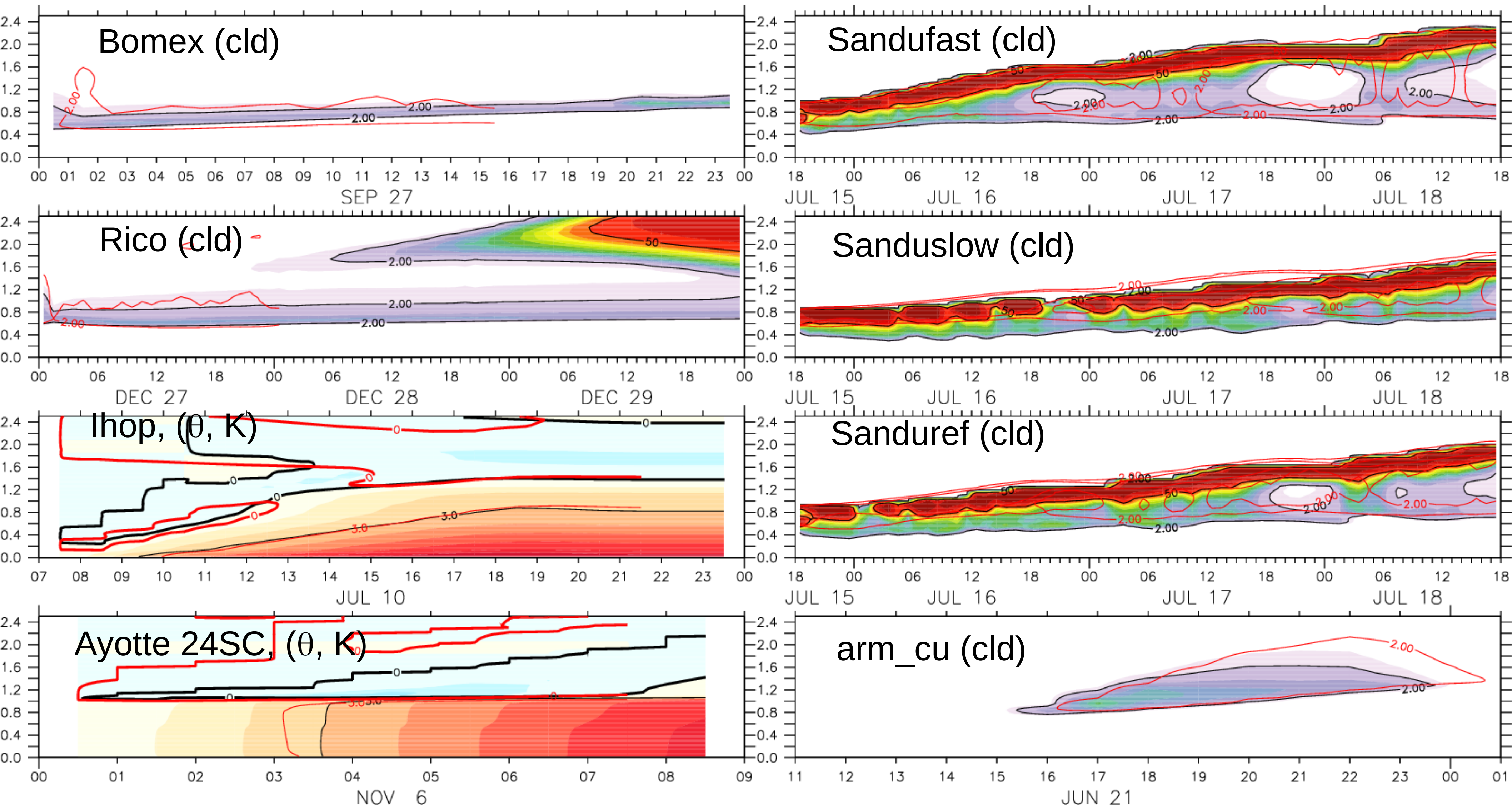


LES

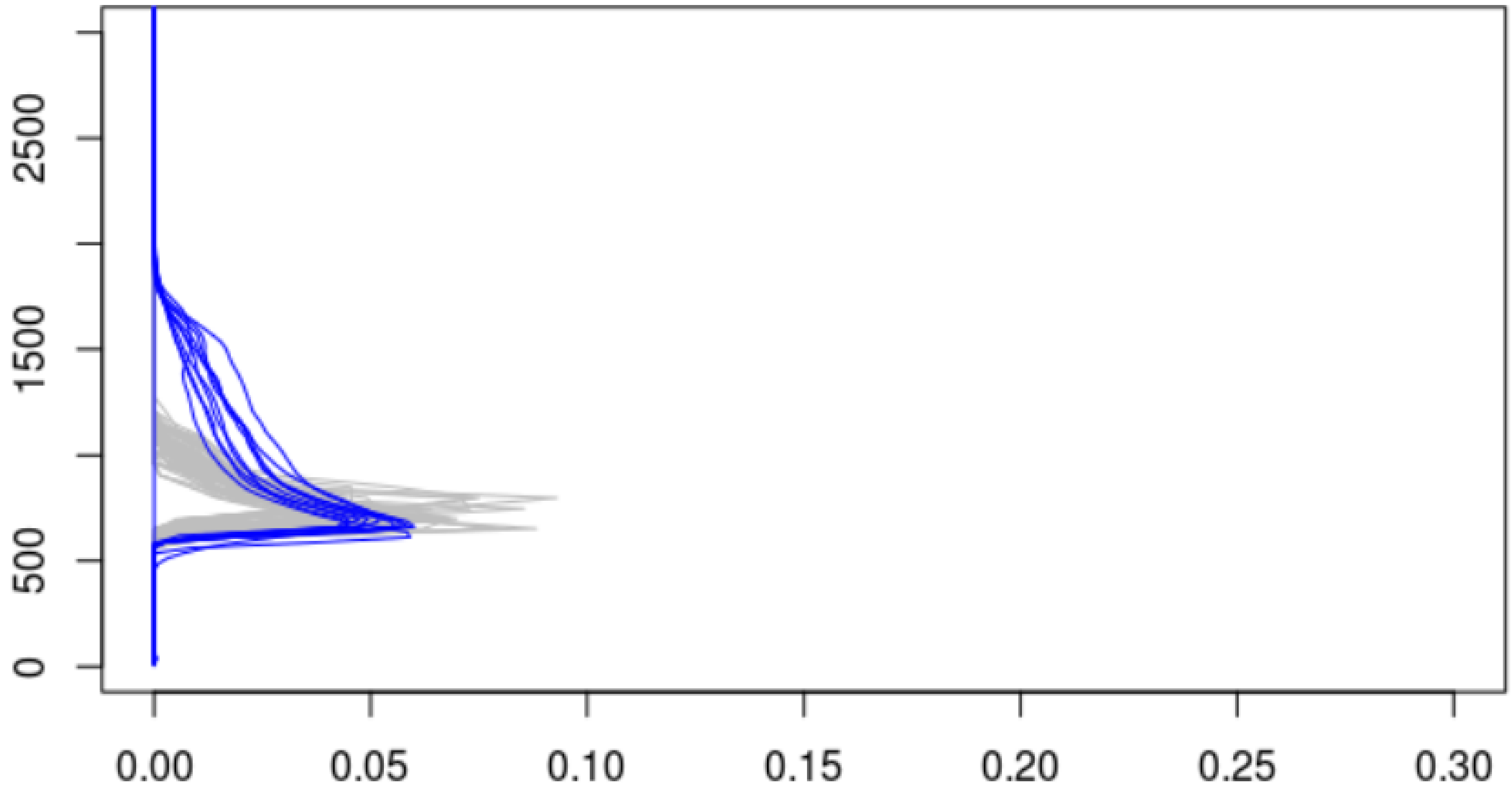


Bomex (cld)
SEP 27
Sandufast (cld)
JUL 15
JUL 16
JUL 17
JUL 18
Rico (cld)
DEC 27
DEC 28
DEC 29
Sanduslow (cld)
Ihop, (θ, K)
JUL 10
Sanduref (cld)
Ayotte 24SC, (θ, K)
NOV 6
arm_cu (cld)
JUN 21

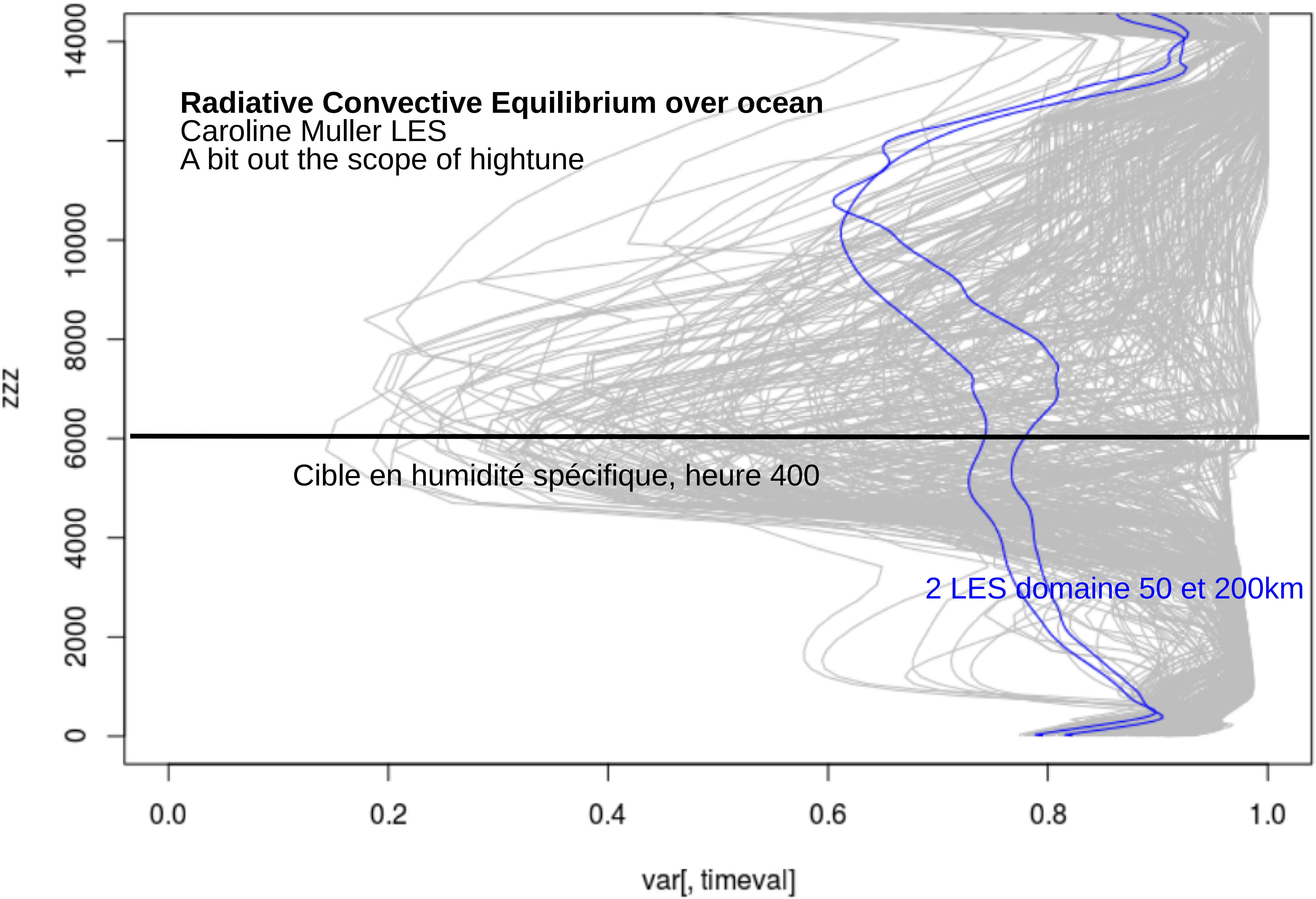
SCM

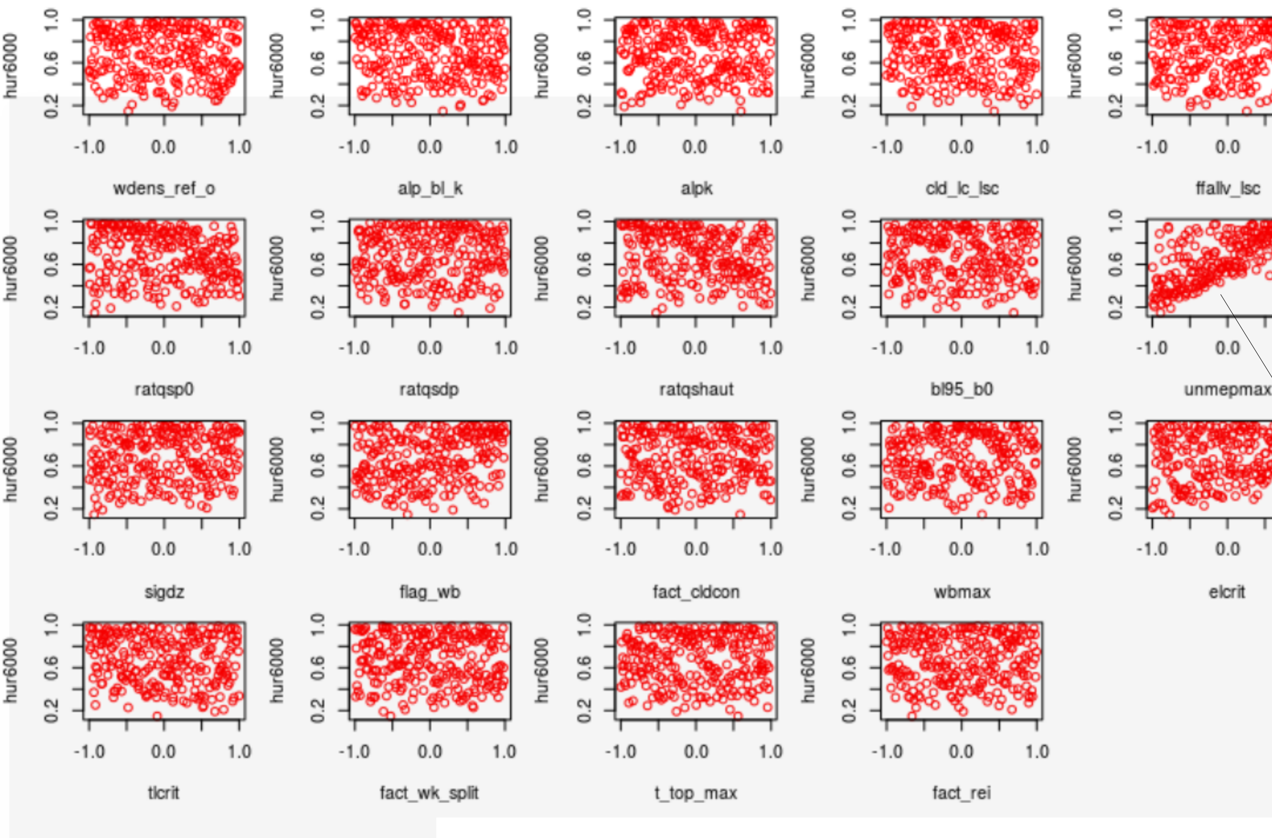

Bomex (cld)
SEP 27
Sandufast (cld)
JUL 15
JUL 16
JUL 17
JUL 18
Rico (cld)
DEC 27
DEC 28
DEC 29
Sanduslow (cld)
Ihop, (θ, K)
JUL 10
Sanduref (cld)
Ayotte 24SC, (θ, K)
NOV 6
arm_cu (cld)
JUN 21

Bomex

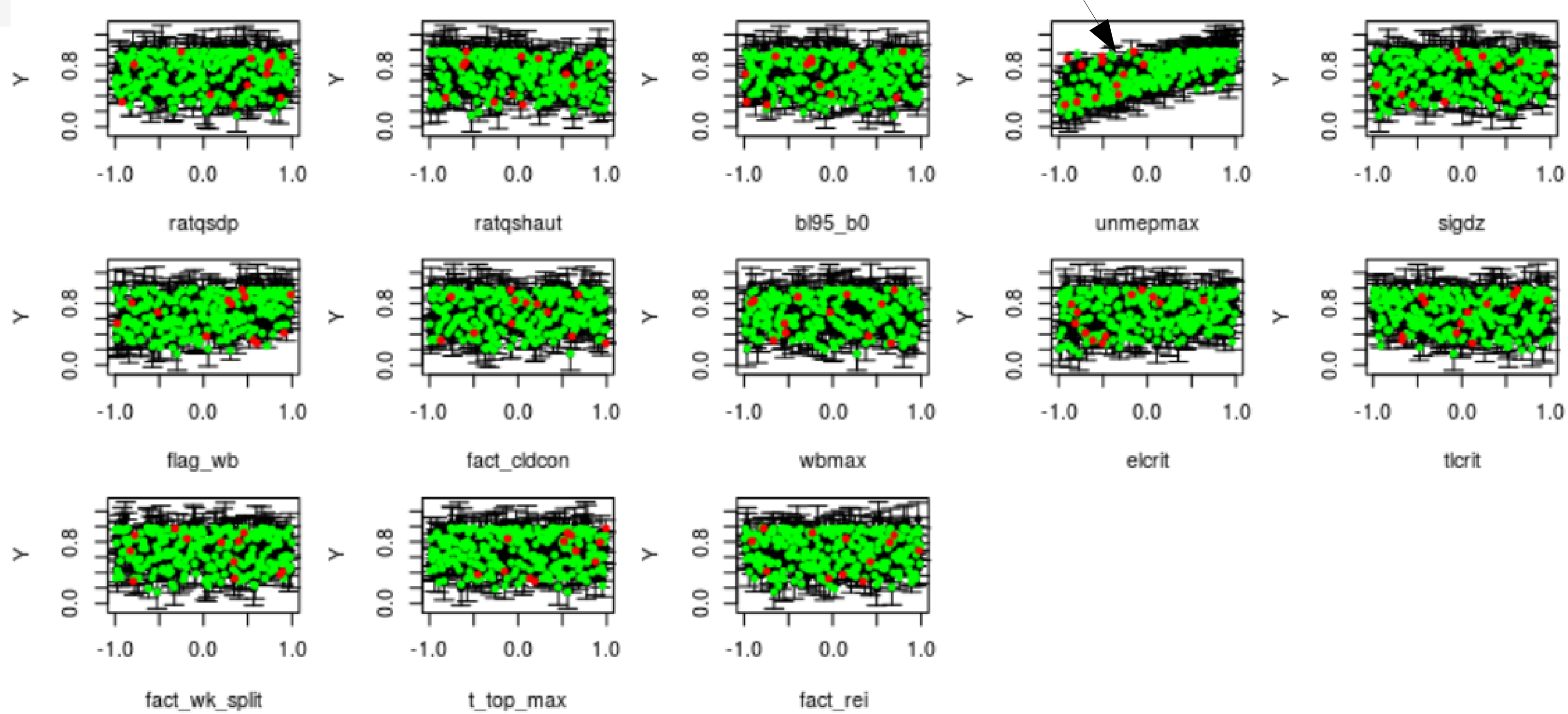


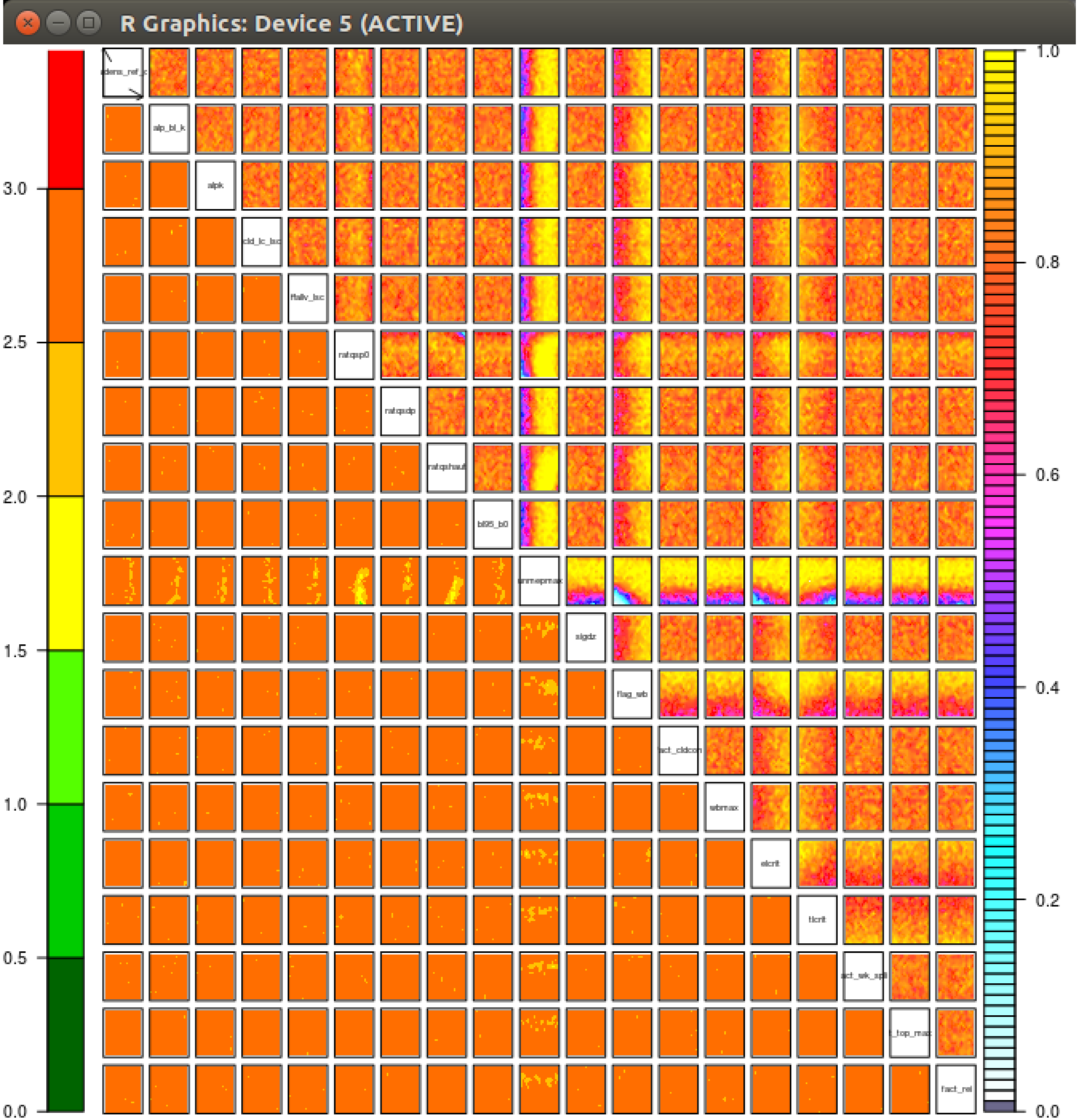
Vertical humidity profile





Maximum precipitation efficiency





Ongoing simulations with the 3D GCM (~ 40 / 250.)

Already working :

→ LES cases avaliable : bomex, arm_cu, sandu*, fire, ayote24SC, ihop, rce
→ metrics : cloud height or native variables at a given time
→ using standard format
→ R scripts to generate samples, compute metrics, compute nroy

To be done on 1D :

→ Generalize standard format for outputs and names
→ Assess LES and 1D setups carefully
→ impose a time step of 1 hour for every model and case ?
→ Additional variables : use CMIP names when possible ?
→ Share ensemble of LES 1D files, at standard format.
→ Choose case names once for all. Key words for the scripts

Objectives in terms of parameterizations

→ Multi metrics - multi case tuning (Ionela, Fleur)
→ Weight of humidity in e/d formulations
→ Comparing EDMF in Arome and LMDZ (Rachel and Marie-Pierre)
→ Competition of shallow/deep convection on RCE (Catherine, Jean-Yves, Abdoul Khadre)
→ Using tendencies versus sampled internal variables as metrics

Technical expectations

→ profiles and 2D z-t metrics (through EOFs)
→ multi-metrics , multi-case combinations
→ explore more efficiently the N-dimensional space to extract parameter values
→ benefit from new developments / optimizations
→ Emulating « atlas » of the 1D / LES comparison
→ Learn how to control the LHC sampler

About the strategy:

→ publication
→ anticipationg radiative tuning. 3D ? 3D controled by 1D ?

Practical questions on scripts :

→ R scripts run under bash ? Combined with python ?
→ Link with python common post-processing
→ distribution under svn or git, with which kind of secuirty

About the strategy:

→ htune_ncdf... could be a shell script + python

Step 1 : sample generation , shell + Rscript
Step 2 : runing SCMS, shell + fortran
Step 3 : computing metrics and graphics, shell + python + cdo/nco
Step 4 : emulator / NROY

To be developed :

Transformation of axes from [-1,1] to real parameter values on graphs, including log axis