VAST project - status

Fuzzy verification toolbox development

Naima Vela, Elena Oberto, Maria Stefania Tesini

September 8, 2014

Contents

COSMO PP VERSUS - Project Plan - Task 4: VAST

Overview

- Introduction of additional statistical techniques in VERSUS

COSMO PP VERSUS - Project Plan - Task 4: VAST

Overview

- Introduction of additional statistical techniques in VERSUS
- Needed for high resolution forecast and observation data (neighborhood methods)

COSMO PP VERSUS - Project Plan - Task 4: VAST

Overview

- Introduction of additional statistical techniques in VERSUS
- Needed for high resolution forecast and observation data (neighborhood methods)
- The main goal of the activity is the integration or adaptation of pre-existing packages

COSMO PP VERSUS - Project Plan - Task 4: VAST

Overview

- Introduction of additional statistical techniques in VERSUS
- Needed for high resolution forecast and observation data (neighborhood methods)
- The main goal of the activity is the integration or adaptation of pre-existing packages
  - Beth Ebert Fuzzy Verification Toolbox

COSMO PP VERSUS - Project Plan - Task 4: VAST

Overview

- Introduction of additional statistical techniques in VERSUS
- Needed for high resolution forecast and observation data (neighborhood methods)
- The main goal of the activity is the integration or adaptation of pre-existing packages
  - Beth Ebert Fuzzy Verification Toolbox
- The pre-processing operations will be performed by the LIBSIM software

COSMO PP VERSUS - Project Plan - Task 4: VAST

Overview

- Introduction of additional statistical techniques in VERSUS
- Needed for high resolution forecast and observation data (neighborhood methods)
- The main goal of the activity is the integration or adaptation of pre-existing packages
  - Beth Ebert Fuzzy Verification Toolbox
- The pre-processing operations will be performed by the LIBSIM software
  - So the system will be able to receive GRIB (1 and 2) as input both for observation and forecast

Task 4.a: Setup of a software able to reproduce Ebert package functionalities

- The main Fortran code (fuzzy_verify.f90) has been produced
  - It will be optimized according to the COSMO standards by the end of October
  - More methods and scores can be added in 2015 according to the WG5 needs

Task 4.a: Setup of a software able to reproduce Ebert package functionalities

- The main Fortran code (fuzzy_verify.f90) has been produced
  - It will be optimized according to the COSMO standards by the end of October
  - More methods and scores can be added in 2015 according to the WG5 needs
- Already included methods: upscaling, yes/no, minimum coverage, fuzzy logic, joint probability, multi-event contingency table, pragmatic approach, practically perfect hindcast

Task 4.a: Setup of a software able to reproduce Ebert package functionalities

- The main Fortran code (fuzzy_verify.f90) has been produced
  - It will be optimized according to the COSMO standards by the end of October
  - More methods and scores can be added in 2015 according to the WG5 needs
- Already included methods: upscaling, yes/no, minimum coverage, fuzzy logic, joint probability, multi-event contingency table, pragmatic approach, practically perfect hindcast
- Already included scores: BIAS, POD, FAR, POFD, Hanssen and Kuipers (HK), ETS, ETS ratio, Fractions skill score (FSS), Brier skill score (BSS), area related RMSE
- All the information needed by the code will be passed through editable namelists

Task 4.b: Setup of a graphic package using R functionalities

- The software will produce various types of graphics according to the user's choice starting from the same input data
  - All the information must be inserted into an editable namelist before the execution of the software

Task 4.b: Setup of a graphic package using R functionalities

- The software will produce various types of graphics according to the user's choice starting from the same input data
  - All the information must be inserted into an editable namelist before the execution of the software
- One R script is produced for each graphic and copied into a backup directory at the end of the process (for possible small modification without re-running all the software)

Task 4.b: Setup of a graphic package using R functionalities

- The software will produce various types of graphics according to the user's choice starting from the same input data
  - All the information must be inserted into an editable namelist before the execution of the software
- One R script is produced for each graphic and copied into a backup directory at the end of the process (for possible small modification without re-running all the software)
- One additional R script is produced containing all the commands useful to run the other scripts

Task 4.b: Setup of a graphic package using R functionalities

- The software will produce various types of graphics according to the user's choice starting from the same input data
  - All the information must be inserted into an editable namelist before the execution of the software
- One R script is produced for each graphic and copied into a backup directory at the end of the process (for possible small modification without re-running all the software)
- One additional R script is produced containing all the commands useful to run the other scripts
- The graphics are placed in the folder indicated in the namelist file

Task 4.b: Setup of a graphic package using R functionalities

- The software will produce various types of graphics according to the user's choice starting from the same input data
  - All the information must be inserted into an editable namelist before the execution of the software
- One R script is produced for each graphic and copied into a backup directory at the end of the process (for possible small modification without re-running all the software)
- One additional R script is produced containing all the commands useful to run the other scripts
- The graphics are placed in the folder indicated in the namelist file
- Types of graphics produced now: score VS scale, score VS intensity, scale-intensity (with both colored boxes and lines)

Task 4.b: Setup of a graphic package using R functionalities

- The software will produce various types of graphics according to the user's choice starting from the same input data
  - All the information must be inserted into an editable namelist before the execution of the software
- One R script is produced for each graphic and copied into a backup directory at the end of the process (for possible small modification without re-running all the software)
- One additional R script is produced containing all the commands useful to run the other scripts
- The graphics are placed in the folder indicated in the namelist file
- Types of graphics produced now: score VS scale, score VS intensity, scale-intensity (with both colored boxes and lines)
- The required R libraries are "fields" and "gplots".

Software description

The software package contains:

Software description

The software package contains:

- `vast.sh`: Shell script containing all the commands to execute the different parts of the procedure

Software description

The software package contains:

- `vast.sh`: Shell script containing all the commands to execute the different parts of the procedure
- `input_csv.nml`: namelist, input for Fortran program `read_csv.f90`

Software description

The software package contains:

- vast.sh: Shell script containing all the commands to execute the different parts of the procedure
- input_csv.nml: namelist, input for Fortran program read_csv.f90
- read_csv.f90: first Fortran program able to process the output of LIBSIM

Software description

The software package contains:

- `vast.sh`: Shell script containing all the commands to execute the different parts of the procedure
- `input_csv.nml`: namelist, input for Fortran program `read_csv.f90`
- `read_csv.f90`: first Fortran program able to process the output of LIBSIM
- `output_csv.nml`: namelist, output of the first Fortran program, input for the main Fortran program (`fuzzy_verify.f90`).

Software description

The software package contains:

- vast.sh: Shell script containing all the commands to execute the different parts of the procedure
- input_csv.nml: namelist, input for Fortran program read_csv.f90
- read_csv.f90: first Fortran program able to process the output of LIBSIM
- output_csv.nml: namelist, output of the first Fortran program, input for the main Fortran program (fuzzy_verify.f90).
- input_fuzzy.nml: namelist, input of the main Fortran program (fuzzy_verify.f90)

Software description

The software package contains:

- `vast.sh`: Shell script containing all the commands to execute the different parts of the procedure
- `input_csv.nml`: namelist, input for Fortran program `read_csv.f90`
- `read_csv.f90`: first Fortran program able to process the output of LIBSIM
- `output_csv.nml`: namelist, output of the first Fortran program, input for the main Fortran program (`fuzzy_verify.f90`).
- `input_fuzzy.nml`: namelist, input of the main Fortran program (`fuzzy_verify.f90`)
- `fuzzy_verify.f90`: Fortran program that produces the verification.

Software description

The software package contains:

- `vast.sh`: Shell script containing all the commands to execute the different parts of the procedure
- `input_csv.nml`: namelist, input for Fortran program `read_csv.f90`
- `read_csv.f90`: first Fortran program able to process the output of LIBSIM
- `output_csv.nml`: namelist, output of the first Fortran program, input for the main Fortran program (`fuzzy_verify.f90`).
- `input_fuzzy.nml`: namelist, input of the main Fortran program (`fuzzy_verify.f90`)
- `fuzzy_verify.f90`: Fortran program that produces the verification.

The software needs:

- Input files (forecast and observed) in csv format preprocessed by LIBSIM (this will change in the future)

The procedure

- The goal of this part of the work (ending in November 2014) is to create a Shell script that can be run by the user and execute all the operations required by the verification with just one click.

The procedure

- The goal of this part of the work (ending in November 2014) is to create a Shell script that can be run by the user and execute all the operations required by the verification with just one click.
- It exists now a Shell script (`vast.sh`) that:
  - Finds the directory where the LIBSIM output are stored
  - Creates a backup of the files in the appropriate directory
  - Creates lists of the observed/forecast files
  - Compiles and runs the Fortran code which reads and rearrange the LIBSIM output, then writes new outputs (fitted for the `fuzzy_verify.f90` code) in the appropriate directory
  - Creates lists of the new observed/forecast processed files
  - Compiles and runs the `fuzzy_verify.f90` code which produces the R scripts (one for each plot) and puts them in the correct directory
  - Runs the command to create all the plots and puts them in the appropriate directory

The procedure

- The goal of this part of the work (ending in November 2014) is to create a Shell script that can be run by the user and execute all the operations required by the verification with just one click.
- It exists now a Shell script (`vast.sh`) that:
  - Finds the directory where the LIBSIM output are stored
  - Creates a backup of the files in the appropriate directory
  - Creates lists of the observed/forecast files
  - Compiles and runs the Fortran code which reads and rearrange the LIBSIM output, then writes new outputs (fitted for the `fuzzy_verify.f90` code) in the appropriate directory
  - Creates lists of the new observed/forecast processed files
  - Compiles and runs the `fuzzy_verify.f90` code which produces the R scripts (one for each plot) and puts them in the correct directory
  - Runs the command to create all the plots and puts them in the appropriate directory
- The script has to be optimized

The procedure: problem

- The IDL Ebert procedure that I followed to produce the Fortran code was not optimized for big amount of data
  - I realised this trying to produce a verification for the whole month of July, 3 hours cumulation, 1 km resolution over the North of Italy.
- The amount of RAM needed to compute the results was too large for the machines I was working on

The procedure: problem

- The IDL Ebert procedure that I followed to produce the Fortran code was not optimized for big amount of data
  - I realised this trying to produce a verification for the whole month of July, 3 hours cumulation, 1 km resolution over the North of Italy.
- The amount of RAM needed to compute the results was too large for the machines I was working on
- I will need some time to modify the code and try to improve its capabilities

Directory structure

IMPORTANT!!

The structure of the directories can be different from the one described here, but the configuration file and, at the moment, the Shell script have to be modified accordingly

Directory structure

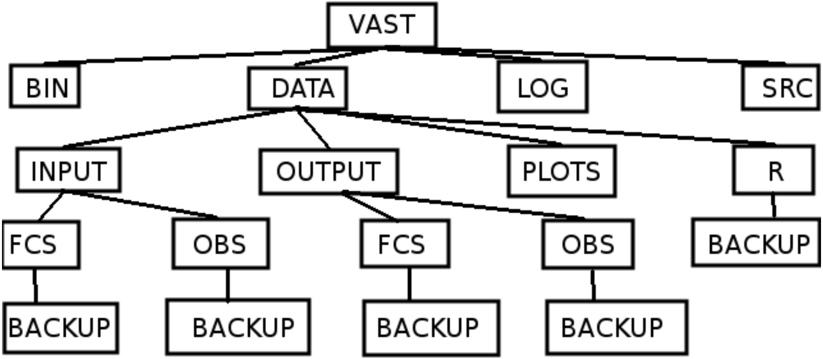


Figure: Structure

Directory structure

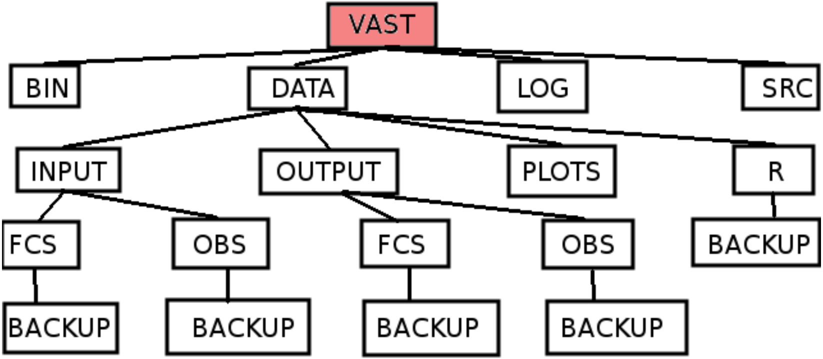


Figure: Main directory, containing all the package

Directory structure

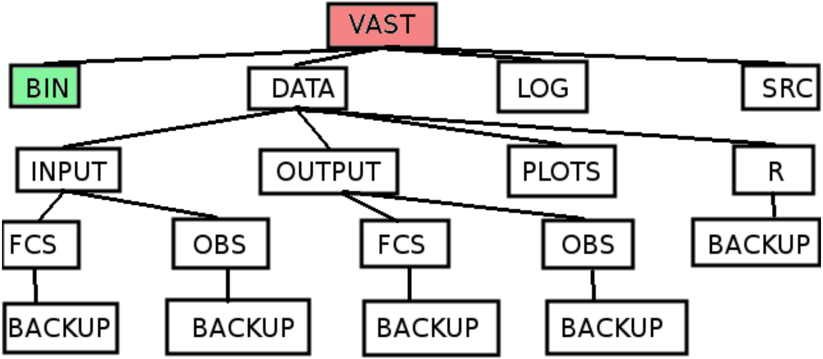


Figure: Executable files

Directory structure

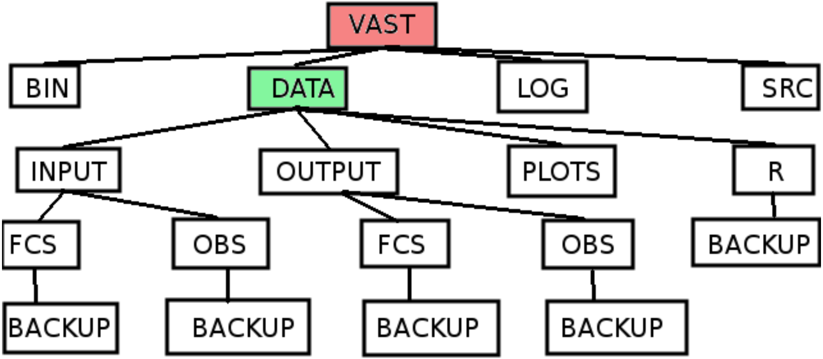


Figure: Data to be analysed

Directory structure

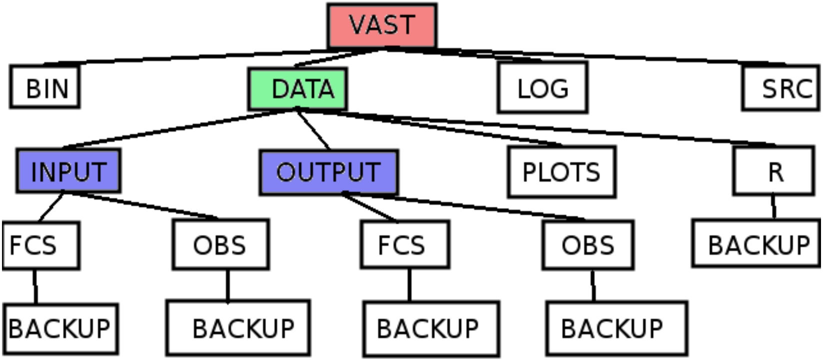


Figure: Data to be analysed

Directory structure

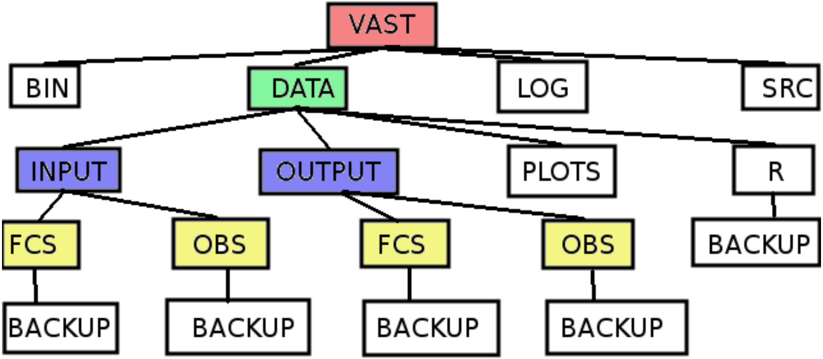


Figure: Forecast and observation subdivision

Directory structure

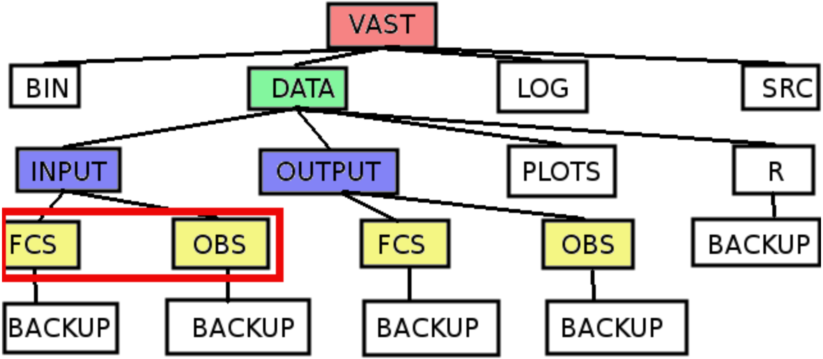


Figure: Directories to be filled with input data, preprocessed by LIBSIM

Directory structure

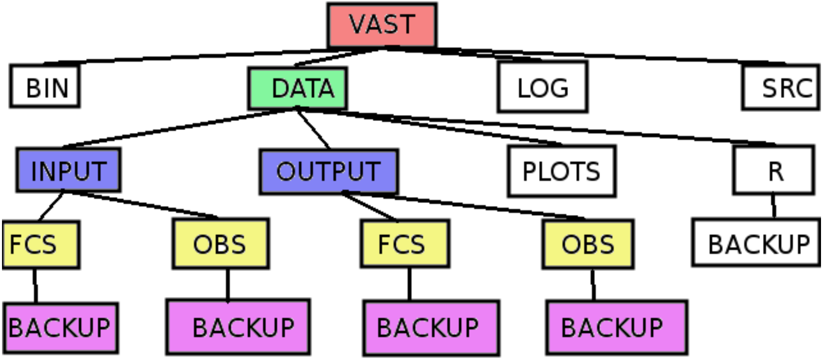


Figure: At the end of the process all the input and the semi processed data can be found in the backup directories

Directory structure

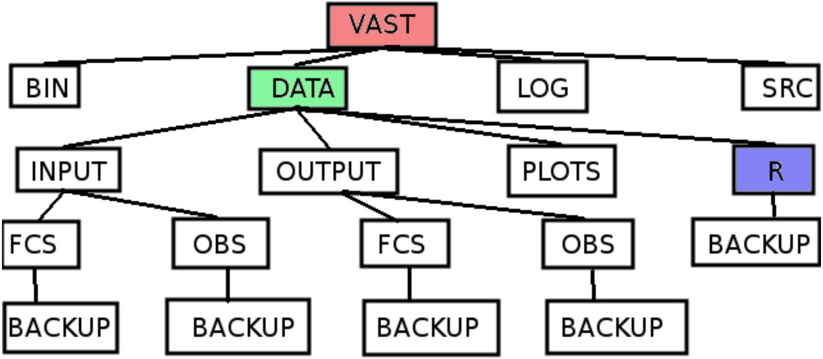


Figure: R scripts (direct output of the Fortran code `fuzzy_verify.f90`)

Directory structure

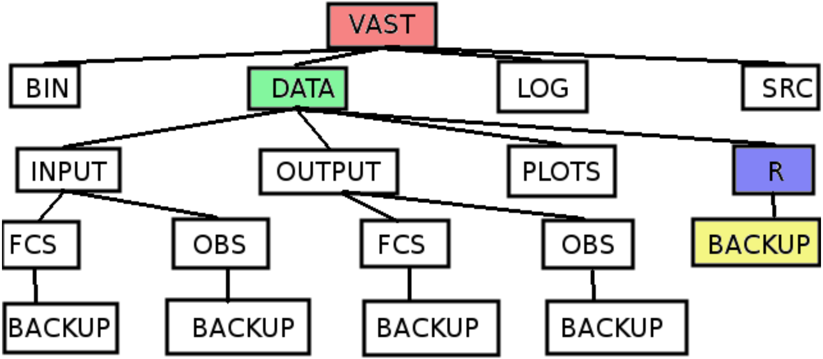


Figure: Backup of the R scripts can be found here at the end of the process

Directory structure

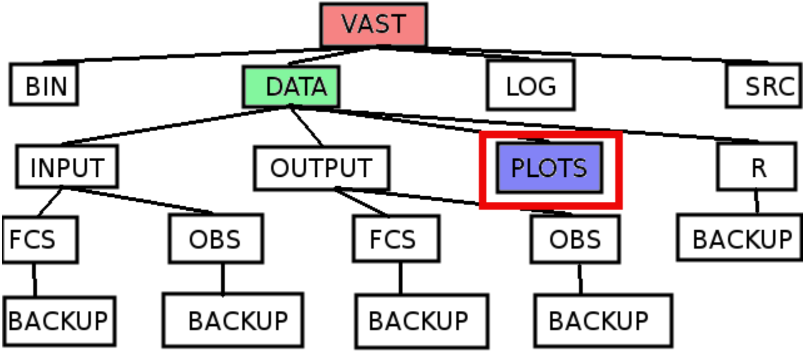


Figure: Folder containing all the produced plots. They should be manually moved from here to their final destination before re-starting the procedure, or there is a chance that they will be overwritten

Configuration files

- There are a total of three configuration files needed to run the software
  - input_csv.nml: Contains the directories needed by the Fortran code read_csv.f90, which is a link between LIBSIM output and the second (more important) Fortran code fuzzy_verify.f90.
  - output_csv.nml: Produced by the first Fortran code. Contains the dimension of the processed gridded data.
  - input_fuzzy.nml: Contains all the specification needed by the main Fortran code (fuzzy_verify.f90).
- input_csv.nml and output_csv.nml should not be edited (after a first correction of the directories' paths)
- input_fuzzy.nml must be edited in order to produce the required results

Configuration files: input_csv.nml

Information read by the read_csv.f90 Fortran code which refines the input files to be ingested by the main program

```
!Directories of input and output for csv files (should be used the default ones)
&directories
dirin_obs='/home/nvela/lavoro/vast_def/data/input/obs/'
dirin_fcs='/home/nvela/lavoro/vast_def/data/input/fcs/'
dirout_obs='/home/nvela/lavoro/vast_def/data/output/obs/'
dirout_fcs='/home/nvela/lavoro/vast_def/data/output/fcs/'
&end

!Names of the list of output files (should be used the default ones)
&filenames
filelist_obs='list_obs.dat'
filelist_fcs='list_fcs.dat'
&end
```

Figure: All the highlighted information should be left unchanged, unless the directory structure has been modified

Configuration files: output_csv.nml

Information about the grid dimension, produced by the first Fortran code (read_csv.f90), read by the second (fuzzy_verify.f90)

```
&dimensions_in
szo1=              80 ,
szo2=              40 ,
szf1=              80 ,
szf2=              40 ,
&end
```

Figure: All these information should be left unchanged if the procedure is followed from the beginning to the end

Configuration files: `input_fuzzy.nml`

Group name: `directories`

```
&directories
!Directory observation (should be used the default one)
dirin_obs='/home/nvela/lavoro/vast_def/data/output/obs/',
!Directory forecast (should be used the default ones)
dirin_fcs='/home/nvela/lavoro/vast_def/data/output/fcs/',
!Directory output R (should be used the default one)
dir_out='/home/nvela/lavoro/vast_def/data/R/',
!Directory output plots (should be used the default one)
dir_plot='/home/nvela/lavoro/vast_def/data/plots/',
!Directory log (should be used the default one)
dir_log='/home/nvela/lavoro/vast_def/log/',
&end
```

IMPORTANT!!

If these paths are modified, the Shell script should be modified accordingly (this will be fixed).

Configuration files: input_fuzzy.nml

Group name: filenames

```
&filenames
!List obs (should be used the default one)
list_obs='list_obs.dat',
!List fcs (should be used the default one)
list_fcs='list_fcs.dat',
!Model
model='COSMO-I2',
!Period
period='20140614-15'
!File list R(should be used the default one)
file_list='list.r'
&end
```

IMPORTANT!!

If these names are modified, the Shell script should be modified accordingly (this will be fixed).

Configuration files: input_fuzzy.nml

Group name: filenames

```
&filenames
!List obs (should be used the default one)
list_obs='list_obs.dat',
!List fcs (should be used the default one)
list_fcs='list_fcs.dat',
!Model
model='COSMO-I2',
!Period
period='20140614-15'
!File list R(should be used the default one)
file_list='list.r'
&end
```

These parameters should be modified to create reasonable file names and plot titles.

Configuration files:vast/src/input_fuzzy.nml

Group name: dimensions

```
&dimensions
!szo3->number of temporal steps in observation input file
szo3=1,
!szf3->number of temporal steps in forecast input file
szf3=1,
!Number of windows (1,3,5,9,17,33,65,...)
n_windows=5,
!Number of valid thresholds
n_thresh=10,
!Thresholds (array of size 10. Fill with -999 if needed)
thresh(1)=0.1,
thresh(2)=0.2,
thresh(3)=0.5,
thresh(4)=1.0,
thresh(5)=2.0,
thresh(6)=5.0,
thresh(7)=10.0,
thresh(8)=15.0,
thresh(9)=20.0,
thresh(10)=30.0
&end
```

Configuration files:vast/src/input_fuzzy.nml

Group name: dimensions

```
&dimensions
!szo3->number of temporal steps in observation input file
szo3=1,
!szf3->number of temporal steps in forecast input file
szf3=1,
!Number of windows (1,3,5,9,17,33,65,...)
n_windows=5,
!Number of valid thresholds
n_thresh=10,
!Thresholds (array of size 10. Fill with -999 if needed)
thresh(1)=0.1,
thresh(2)=0.2,
thresh(3)=0.5,
thresh(4)=1.0,
thresh(5)=2.0,
thresh(6)=5.0,
thresh(7)=10.0,
thresh(8)=15.0,
thresh(9)=20.0,
thresh(10)=30.0
&end
```

To be implemented

Configuration files:vast/src/input_fuzzy.nml

Group name: dimensions

```
&dimensions
!szo3->number of temporal steps in observation input file
szo3=1,
!szf3->number of temporal steps in forecast input file
szf3=1,
!Number of windows (1,3,5,9,17,33,65,...)
n_windows=5,
!Number of valid thresholds
n_thresh=10,
!Thresholds (array of size 10. Fill with -999 if needed)
thresh(1)=0.1,
thresh(2)=0.2,
thresh(3)=0.5,
thresh(4)=1.0,
thresh(5)=2.0,
thresh(6)=5.0,
thresh(7)=10.0,
thresh(8)=15.0,
thresh(9)=20.0,
thresh(10)=30.0
&end
```

Differernt box dimension: to be implemented

Configuration files: input_fuzzy.nml

Group name: methods_scores

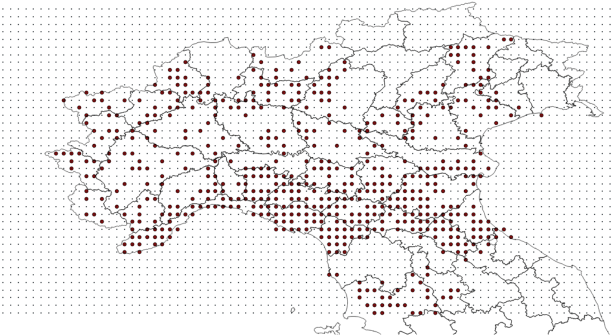
```
&methods_scores
!Number of methods of type 1 (max 6)
nm1=5,
!Methods
method1(1)='UP',
method1(2)='YN',
method1(3)='MC',
method1(4)='FZ',
method1(5)='JP',
!Scores
score1(1)='BIAS',
score1(2)='FAR',
score1(3)='FAR',
score1(4)='FAR',
score1(5)='FAR',
```

```
!Number of methods of type 2 (max 3)
nm2=2,
!Methods
method2(1)='FB',
method2(2)='PG',
!Scores
score2(1)='FSS',
score2(2)='BSS',
!Number of methods of type 3
nm3=1,
!Methods
method3(1)='RM',
!Scores
score3(1)='RMSE'
&end
```

The same score can be calculated with different methods and one method can calculate more scores

Grid and rain gauges

All the data used for this test have been preprocessed with LIBSIM by Maria Stefania Tesini (Arpa SIMC)



- Grid width: LON 06.0 - 14.0 E; LAT 43.0 - 47.0 N
- Grid resolution: 0.1°
- Precipitation cumulation: 3h
- Observation data: rain gauges
- Forecast data: COSMO-I2

Case study: June 14-15 2014

Widespread rainfalls over the North od Italy

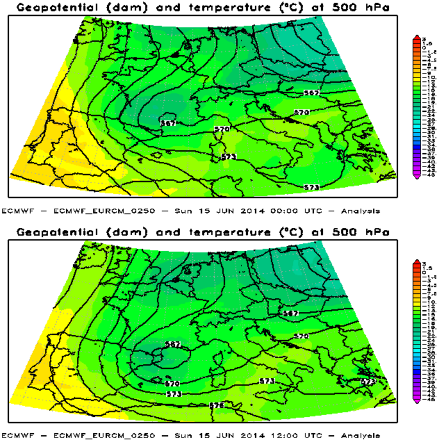


Figure: Geopotential at 500 hP over South-western Europe - June 15 at 00, 06, 12, 18 UTC- ECMWF Analysis

Case study: June 14-15 2014

Widespread rainfalls over the North od Italy

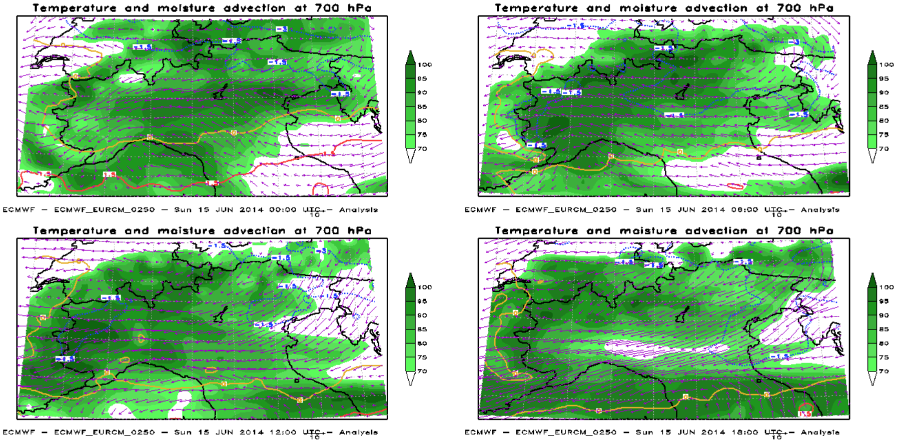


Figure: Temperature and moisture advection at 700 hP over the North of Italy - June 15 at 00, 06, 12, 18 UTC- ECMWF Analysis

Data: Template of the forecast data elaborated by LIBSIM

```
Date,Time_range,P1,P2,Longitude,Latitude,Level1,L1,Level2,L2,Report,B01192,B13011
2014-06-14 21:00:00,1,75600,10800,6.00000,43.00000,1,,,,generic,1,0.00000000
2014-06-14 21:00:00,1,75600,10800,6.10000,43.00000,1,,,,generic,2,0.00000000
2014-06-14 21:00:00,1,75600,10800,6.20000,43.00000,1,,,,generic,3,0.00000000
2014-06-14 21:00:00,1,75600,10800,6.30000,43.00000,1,,,,generic,4,0.00000000
2014-06-14 21:00:00,1,75600,10800,6.40000,43.00000,1,,,,generic,5,0.00000000
2014-06-14 21:00:00,1,75600,10800,6.50000,43.00000,1,,,,generic,6,0.00000000
2014-06-14 21:00:00,1,75600,10800,6.60000,43.00000,1,,,,generic,7,0.00000000
2014-06-14 21:00:00,1,75600,10800,6.70000,43.00000,1,,,,generic,8,0.00000000
2014-06-14 21:00:00,1,75600,10800,6.80000,43.00000,1,,,,generic,9,0.00000000
2014-06-14 21:00:00,1,75600,10800,6.90000,43.00000,1,,,,generic,10,0.00000000
2014-06-14 21:00:00,1,75600,10800,7.00000,43.00000,1,,,,generic,11,0.00000000
2014-06-14 21:00:00,1,75600,10800,7.10000,43.00000,1,,,,generic,12,0.00000000
2014-06-14 21:00:00,1,75600,10800,7.20000,43.00000,1,,,,generic,13,0.332031250E-01
2014-06-14 21:00:00,1,75600,10800,7.30000,43.00000,1,,,,generic,14,3.50585938
2014-06-14 21:00:00,1,75600,10800,7.40000,43.00000,1,,,,generic,15,8.21679688
2014-06-14 21:00:00,1,75600,10800,7.50000,43.00000,1,,,,generic,16,2.63867188
2014-06-14 21:00:00,1,75600,10800,7.60000,43.00000,1,,,,generic,17,0.429687500E-01
2014-06-14 21:00:00,1,75600,10800,7.70000,43.00000,1,,,,generic,18,0.996093750E-01
```

Forecast time steps

0024: 2014/06/14_00_21⇒2014/06/15_00_24

2448: 2014/06/14_00_27⇒2014/06/14_00_48

Data: Template of the observed data elaborated by LIBSIM

```
Date,Time range,P1,P2,Longitude,Latitude,Level1,L1,Level2,L2,Report,B01192,B13011
2014-06-14 21:00:00,1,0,10800,10.90000,43.00000,1,,,,generic,50,0.00000000
2014-06-14 21:00:00,1,0,10800,11.00000,43.00000,1,,,,generic,51,0.00000000
2014-06-14 21:00:00,1,0,10800,11.40000,43.00000,1,,,,generic,55,0.599609375
2014-06-14 21:00:00,1,0,10800,11.50000,43.00000,1,,,,generic,56,0.799804688
2014-06-14 21:00:00,1,0,10800,11.70000,43.00000,1,,,,generic,58,3.00000000
2014-06-14 21:00:00,1,0,10800,10.80000,43.10000,1,,,,generic,130,0.00000000
2014-06-14 21:00:00,1,0,10800,10.90000,43.10000,1,,,,generic,131,0.00000000
2014-06-14 21:00:00,1,0,10800,11.00000,43.10000,1,,,,generic,132,0.00000000
2014-06-14 21:00:00,1,0,10800,11.10000,43.10000,1,,,,generic,133,0.00000000
2014-06-14 21:00:00,1,0,10800,11.20000,43.10000,1,,,,generic,134,0.00000000
2014-06-14 21:00:00,1,0,10800,11.30000,43.10000,1,,,,generic,135,1.40039063
2014-06-14 21:00:00,1,0,10800,11.70000,43.10000,1,,,,generic,139,2.00000000
2014-06-14 21:00:00,1,0,10800,10.70000,43.20000,1,,,,generic,210,0.00000000
2014-06-14 21:00:00,1,0,10800,11.00000,43.20000,1,,,,generic,213,0.00000000
2014-06-14 21:00:00,1,0,10800,11.10000,43.20000,1,,,,generic,214,0.00000000
2014-06-14 21:00:00,1,0,10800,11.60000,43.20000,1,,,,generic,219,2.20019531
2014-06-14 21:00:00,1,0,10800,11.80000,43.20000,1,,,,generic,221,1.79980469
2014-06-14 21:00:00,1,0,10800,11.90000,43.20000,1,,,,generic,222,0.400390625
```

Observed time steps

2014/06/14_21⇒2014/06/16_00

Results: FAR

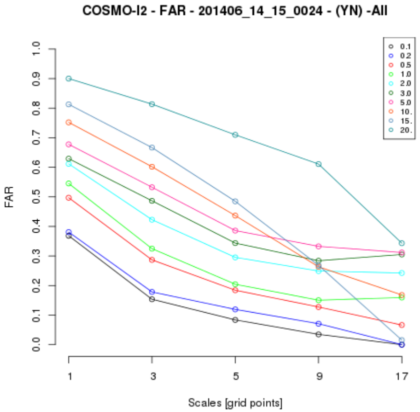


Figure: FAR, Yes/No method, first 24 hours of forecast

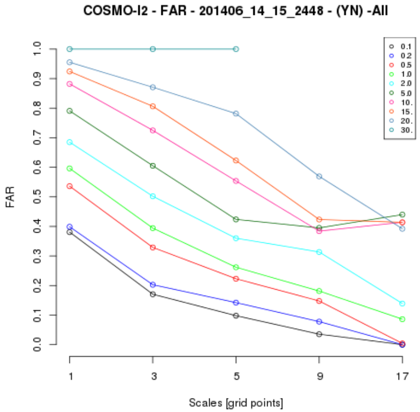


Figure: FAR, Yes/No method, second 24 hours of forecast

Results: POD

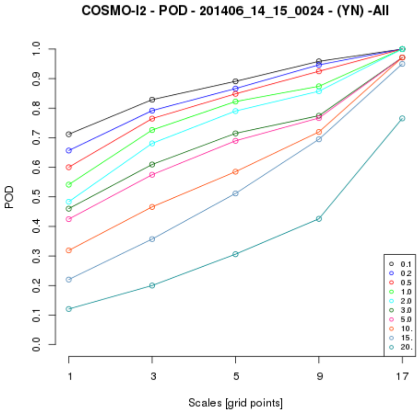


Figure: POD, Yes/No method, first 24 hours of forecast

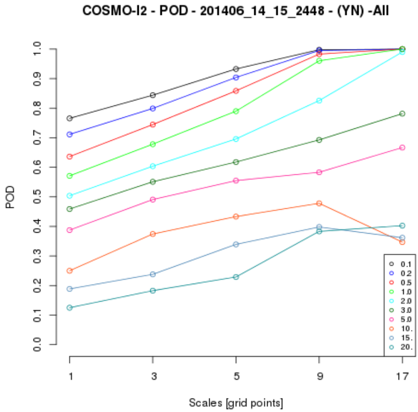


Figure: POD, Yes/No method, second 24 hours of forecast

Results: FSS (with the indication of the skilful scales)

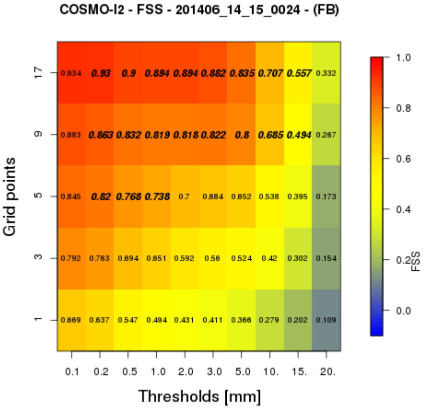


Figure: FSS, first 24 hours of forecast
.

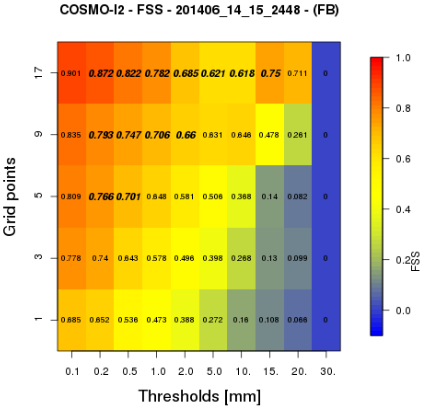


Figure: FSS, second 24 hours of forecast

Conclusions

Conclusions

- **The software works!**

Conclusions

- **The software works!**

BUT

Conclusions

- **The software works!**

BUT

- The software needs to be optimized

Conclusions

- **The software works!**

BUT

- The software needs to be optimized
  - For the ingestion of large amount of data
  - To fit with the COSMO guidelines
  - To produce more/different scores

THANK YOU FOR YOUR ATTENTION!